

THE FIRST 20 HOURS

How to Learn
Anything . . .

Fast

'Lots of books promise
to change your life.
This one actually will'
SETH GODIN



JOSH KAUFMAN

Author of the Number One Bestseller

the personal
MBA



Josh Kaufman20 JAM

PERTAMA

Bagaimana Belajar Apa Saja ... Cepat



Isi



Catatan untuk Pembaca¹ Potret Penulis
sebagai Pecandu Belajar² Sepuluh Prinsip
Akuisisi Keterampilan Cepat³ Sepuluh Prinsip
Pembelajaran Eektif⁴ Yoga⁵ Pemrograman⁶
Mengetik Sentuh⁷ Go⁸ Ukulele⁹ Windsurng

Kata

AkhirCatatanUca

pan Terima Kasih

TENTANG PENULIS

Josh Kaufman membantu orang menghasilkan lebih banyak uang, menyelesaikan lebih banyak, dan bersenang-senang. Buku pertamanya, *The Personal MBA*, adalah buku terlaris internasional. Dia tinggal di Colorado.

www.personalmba.com

Untuk Lela

20 JAM PERTAMA

"Peluang besar tidak berharga tanpa keterampilan. Tidak ada lagi alasan! Kaufman membuktikan bahwa kita semua memiliki kapasitas untuk menjadi ahli! Scott Belsky, pendiri, Behance, dan penulis Making Ideas Happen! Jika Anda seperti saya, Anda akan sangat terinspirasi sehingga Anda akan berhenti membaca untuk menerapkan pendekatan ini pada proyek Anda sendiri yang ditunda-tunda. Setelah membaca tiga bab, saya mencoba tekniknya untuk mempelajari bahasa pemrograman baru, dan saya terpesona dengan betapa cepatnya saya menjadi uent! Derek Sivers, pendiri, CD Baby, sivers.org! Dalam buku kecil yang menginspirasi ini, Josh Kaufman berpendapat bahwa Anda bisa menjadi cukup baik dalam apa pun untuk menikmati diri Anda hanya dalam 20 jam. Yang ada di antara Anda dan bermain ukulele adalah waktu TV Anda selama dua minggu ke depan! Laura Vanderkam, penulis 168 Hours dan What the Most Successful People Do Before Breakfast

"Dengan banyaknya informasi dan perubahan di dunia saat ini, orang yang dapat beradaptasi dan belajar paling cepat akan menjadi yang paling sukses. Kaufman menguraikan ilmu belajar dengan cara yang berguna, menghibur, dan menarik. Jika Anda peduli untuk mempertahankan pekerjaan, bisnis Anda, atau keunggulan Anda, buku ini untuk Anda! Pamela Slim, penulis Escape from Cubicle Nation

Catatan untuk Pembaca

Lyf begitu pendek, kapal begitu panjang ke lerne.—
GEOFFREY CHAUCER, PARLEMENT OF FOULES, 1374

• • •

"Ada begitu banyak yang ingin saya lakukan ... dan sangat sedikit waktu." Kisah kehidupan modern. Luangkan waktu sejenak untuk mempertimbangkan berapa banyak hal yang ingin Anda pelajari bagaimana melakukannya. Apa yang ada di daftar Anda? Apa yang menghalangi Anda untuk memulai? Dua hal, kemungkinan besar: waktu dan keterampilan. Inilah kebenaran yang tidak nyaman: pengalaman paling berharga dalam hidup hampir selalu membutuhkan beberapa tingkat keterampilan. Keterampilan membutuhkan waktu dan untuk dikuasai—waktu yang tidak kita miliki, dan eort kita enggan untuk berkontribusi." Aku akan melakukannya suatu hari nanti, ketika aku tiba waktunya." Lebih mudah untuk duduk di depan televisi atau menjelajahi web, terus terang ... Jadi itulah yang kebanyakan dari kita lakukan, dan keinginan kita tetap menjadi mimpi. Inilah kebenaran lain yang tidak nyaman: banyak hal yang tidak menyenangkan sampai Anda pandai dalam hal itu. Setiap keterampilan memiliki apa yang saya sebut penghalang frustrasi—periode waktu di mana Anda sangat tidak terampil, dan Anda sangat menyadari fakta itu. Mengapa memulai sesuatu ketika Anda tahu Anda akan menjadi buruk dalam hal itu? Bukankah menyenangkan bisa menguasai keterampilan baru dengan lessangst? Untuk menembus penghalang frustrasi dengan cepat, sehingga Anda bisa

Masuk ke bagian yang bermanfaat? Untuk menghabiskan lebih sedikit waktu untuk bersusah payah melalui kebingungan dan keraguan, dan lebih banyak waktu untuk bersenang-senang? Apakah mungkin untuk memperoleh keterampilan baru dengan tidak terlalu menyakitkan, dengan cara yang membutuhkan waktu dan semangat yang jauh lebih sedikit? Saya berbicara dari pengalaman: ya, itu mungkin. Buku ini adalah tentang pencarian pribadi saya untuk menguji seni dan ilmu perolehan keterampilan yang cepat—bagaimana mempelajari keterampilan baru secepat mungkin. Tujuan dari buku ini adalah untuk membantu Anda memperoleh keterampilan baru dalam waktu singkat. Menurut pengalaman saya, dibutuhkan sekitar dua puluh jam latihan untuk menembus penghalang frustrasi: untuk beralih dari tidak tahu apa-apa tentang apa yang Anda coba lakukan menjadi tampil sangat baik. Buku ini adalah pendekatan sistematis untuk memperoleh keterampilan baru secepat mungkin. Metodenya bersifat universal. Tidak masalah apakah Anda ingin belajar bahasa, menulis novel, melukis potret, memulai bisnis, atau dan pesawat terbang. Jika Anda menginvestasikan hanya dua puluh jam untuk mempelajari dasar-dasar keterampilan, Anda akan terkejut betapa baiknya Anda. Keterampilan apa pun yang ingin Anda peroleh, buku ini akan membantu Anda mendapatkannya dalam waktu yang lebih singkat dan dengan lebih sedikit energi yang terbuang. Dengan sedikit fokus dan strategis, Anda akan tampil baik dengan cepat, tanpa frustrasi yang mengebarkan. Dalam buku ini, kita akan mulai dengan prinsip-prinsip perolehan keterampilan yang cepat: bagaimana memperoleh keterampilan baru secepat mungkin. Ide-ide dan praktik ini tidak rumit, jadi tidak perlu waktu lama untuk dipelajari. Kemudian, saya akan menjelaskan cara menggunakan prinsip-prinsip ini di dunia nyata dengan menunjukkan kepada Anda bagaimana saya memperoleh enam keterampilan baru berikut dalam dua puluh jam atau kurang masing-masing, dengan tidak lebih dari sembilan puluh menit latihan per hari.

Mengembangkan latihan yoga pribadi

- Menulis program komputer berbasis webBelajar kembali ke jenis sentuhanMenjelajahi permainan papan tertua dan terkompleks dalam sejarahMemainkan alat musikWindsurng

Saya harap buku ini mendorong Anda untuk membersihkan daftar "ingin Anda berdebar" lama, memeriksanya kembali, dan berkomitmen untuk mempelajari sesuatu yang baru.

Josh Kaufman
Fort
Collins, Colorado,
Amerika Serikat

Untuk pembaruan tentang materi dalam buku ini, kunjungi <http://first20hours.com/updates>.



Potret Penulis sebagai Pecandu Pembelajaran

Saya bangun setiap pagi bertekad untuk mengubah dunia dan bersenang-senang. Terkadang ini membuat perencanaan hari saya menjadi sulit.—E. B. WHITE, PENULIS ESAI DAN PENULIS CHARLOTTE'S WEB AND THE ELEMENTS OF STYLE

• • •

Hai. Nama saya Josh Kaufman, dan saya seorang pecandu belajar. Rumah dan rak-rak saya ditumpuk tinggi dengan buku, peralatan, dan peralatan yang tidak terpakai dari segala jenis, yang sebagian besar adalah debu yang perlahan-lahan menumpuk. Saya memiliki daftar "untuk dipelajari" dengan ratusan item. Keranjang belanja Amazon.com saya saat ini memiliki 241 item di dalamnya—semua buku yang ingin saya baca. Saya tidak bisa masuk ke toko buku tanpa pergi dengan tiga atau empat buku baru, untuk ditambahkan ke 852 volume yang sudah saya miliki. Setiap hari, saya mendapatkan ide untuk proyek atau eksperimen lain, yang saya tambahkan ke daftar "suatu hari nanti/mungkin" saya yang terus bertambah. Melihat semua yang ingin saya pelajari bagaimana melakukannya terasa luar biasa, jadi saya tidak terlalu sering melihat daftarnya. Saya ingin belajar bagaimana meningkatkan bisnis penerbitan saya. Saya ingin belajar cara merekam dan mengedit video. Saya ingin memproduksi program audio. Saya ingin belajar bagaimana memberikan seminar yang lebih baik dan mengajar kursus yang lebih baik.

Saya punya ide untuk produk baru, tetapi saya tidak tahu bagaimana membangunnya. Saya memiliki ide untuk program komputer baru, tetapi saya tidak tahu bagaimana membuatnya. Saya memiliki lebih banyak ide proyek penulisan potensial di kepala saya daripada waktu dan energi untuk menuliskannya. Saya ingin belajar menggambar. Saya ingin belajar cara white-waterkayak. Saya ingin belajar y shing. Saya ingin belajar panjat tebing. Saya ingin bisa bermain gitar, ukulele, piano, dan biola elektrik. Ada game yang saya minati selama bertahun-tahun, seperti Go, tetapi saya belum belajar cara memainkannya. Saya memiliki permainan yang sudah saya ketahui cara bermainnya, seperti catur, tetapi saya tidak terlalu pandai dalam hal itu, jadi tidak terlalu menyenangkan, dan saya tidak terlalu sering memainkannya. Saya suka gagasan bermain golf, tetapi setiap permainan yang saya mainkan berubah menjadi latihan yang tabah dalam tertawa atau malu. (Saya biasanya mengatakan golf maraton Iplay: pada akhir delapan belas lubang, saya telah berlari maraton.)Sepertinya setiap hari saya menambahkan beberapa keterampilan baru ke daftar hal-hal yang ingin saya lakukan, ad innum. Begitu banyak yang harus dipelajari, waktu yang sangat sedikit. Secara alami, saya adalah tipe pria yang suka melakukannya sendiri. Jika sesuatu perlu dilakukan, saya lebih suka mencobanya sendiri daripada mencari bantuan. Bahkan jika orang lain bisa melakukannya lebih cepat atau lebih baik, saya enggan merampok pengalaman belajar saya sendiri. Untuk memperumit masalah, Kelsey, istri saya, menjalankan bisnisnya sendiri, menerbitkan kursus pendidikan berkelanjutan untuk guru yoga. Bisnis baik untuk kami berdua, jadi selalu ada banyak yang harus dilakukan. Untuk membuat hidup lebih menarik, kami menyambut putri kami, Lela, ke dunia. Lela berusia sembilan bulan saat saya menulis ini. Sebelum Lela lahir, Kelsey dan saya memutuskan bahwa jika kami akan memiliki anak, kami ingin menjadikan membesarkan mereka sendiri sebagai prioritas. Salah satu alasan utama saya berhenti dari pekerjaan saya sebelumnya di perusahaan Fortune 500 adalah untuk memiliki kemampuan untuk bekerja dari

pulang, atur jadwal saya sendiri, dan habiskan waktu sebanyak mungkin bersama keluarga saya. Kelsey dan saya berbagi tanggung jawab pengasuhan secara setara. Karena kami adalah rumah tangga dua bisnis, Kelsey bekerja di pagi hari, sementara saya merawat Lela. Sore harinya, Kelsey merawat Lela, dan saya bekerja sampai waktu makan malam. Itu memberi saya sekitar dua puluh jam setiap minggu untuk bekerja, ditambah waktu apa pun yang bisa saya ambil saat Lela tidur siang. Setelah Lela lahir, saya merasa hampir tidak punya cukup waktu untuk menyelesaikan pekerjaan saya, apalagi memperoleh keterampilan baru. Bagi seorang pecandu belajar, itu membuat gila. Saya tidak ingin menyerah pada pembelajaran dan pertumbuhan sepenuhnya, bahkan dengan tanggung jawab baru saya. Saya tidak punya banyak waktu luang, tetapi saya bersedia menginvestasikan apa yang saya miliki sebijak mungkin. Itulah yang mendorong minat saya pada apa yang saya sebut akuisisi keterampilan cepat: metode mempelajari keterampilan baru dengan cepat. Saya ingin terus memperoleh keterampilan baru, tetapi saya tidak ingin prosesnya memakan waktu selamanya. Saya ingin mengambil hal-hal penting dengan cepat, sehingga saya dapat membuat kemajuan yang nyata tanpa terus-menerus merasa frustrasi. Saya yakin Anda bisa berhubungan. Berapa banyak waktu "luang" yang Anda miliki setiap hari, setelah semua pekerjaan dan kewajiban keluarga Anda selesai? Apakah Anda merasa seperti Anda membutuhkan tiga puluh enam atau empat puluh delapan jam dalam sehari untuk duduk dan belajar sesuatu yang baru? Ada klise lama: "bekerja lebih cerdas, bukan lebih keras." Ternyata, proses perolehan keterampilan tidak benar-benar tentang jam mentah yang Anda habiskan ... itulah yang Anda masukkan ke dalam jam-jam itu.

Sialan, Malcolm Gladwell

Pada tahun 2008, Malcolm Gladwell menulis sebuah buku berjudul *Outliers: The Story of Success*. Di dalamnya, dia mulai mencoba menjelaskan apa yang membuat orang tertentu lebih sukses daripada yang lain. Salah satu ide yang disebutkan Gladwell berulang kali adalah apa yang disebut "aturan 10.000 jam." Berdasarkan penelitian yang dilakukan oleh Dr.

K. Anders Ericsson dari Florida State University, kinerja tingkat ahli membutuhkan, rata-rata, sepuluh ribu jam latihan yang disengaja untuk mencapainya.¹ Sepuluh ribu jam sama dengan delapan jam latihan yang disengaja setiap hari selama sekitar tiga setengah tahun, tanpa istirahat, tanpa akhir pekan, dan tanpa liburan. Dengan asumsi standar 260 hari kerja setahun tanpa gangguan, itu adalah pekerjaan penuh waktu selama hampir bertahun-tahun, dengan asumsi Anda menghabiskan 100 persen waktu itu mengerahkan 100 persen energi dan semangat Anda. Dalam praktiknya, tingkat perhatian terfokus ini sangat melelahkan. Bahkan pemain kelas dunia di usia ultrakompetitif (seperti pertunjukan musik dan olahraga profesional) hanya dapat mengumpulkan energi selama sekitar tiga setengah jam latihan yang disengaja setiap hari. Itu berarti butuh waktu satu dekade atau lebih untuk mengembangkan keterampilan penguasaan. Intinya, jika Anda ingin menguasai keterampilan baru, penelitian Dr. Ericsson menunjukkan bahwa Anda berada dalam jangka waktu yang sangat lama. Menjadi yang terbaik di dunia dalam hal apa pun, bahkan untuk sementara waktu, membutuhkan latihan tanpa henti selama bertahun-tahun. Jika Anda tidak mau meluangkan waktu dan semangat, Anda akan dibayangi oleh mereka yang melakukannya. Outliers melesat langsung ke bagian atas daftar buku terlaris nonfiction, dan tinggal di sana selama tiga bulan. Dalam semalam, "aturan 10.000 jam" ada di mana-mana. Seolah-olah mempelajari keterampilan baru tidak cukup sulit. Anda tidak hanya harus meluangkan waktu untuk berlatih ... Tapi Anda sekarang juga harus menghabiskan ribuan jam? Sebagian besar dari kita menganggap diri kita beruntung jika kita dapat menyisihkan beberapa jam seminggu. Mengapa repot-repot sama sekali jika butuh waktu lama untuk menjadi pandai dalam sesuatu?

Lihatlah perbuatan-Ku, hai yang perkasa, dan putus asa!

Sebelum Anda melepaskan semua harapan, pertimbangkan ini.

Ada elemen penelitian Dr. Ericsson yang sangat mudah diabaikan: ini adalah studi tentang kinerja tingkat ahli. Jika Anda ingin menjadi Tiger Woods berikutnya, Anda mungkin perlu menghabiskan setidaknya sepuluh ribu jam dengan sengaja dan sistematis untuk berlatih setiap aspek golf. Hampir setiap pegolf profesional mulai bermain pada usia yang sangat muda dan telah berlatih tanpa henti setidaknya selama tujuh tahun. Mengembangkan penguasaan kelas dunia membutuhkan waktu. Di sisi lain, bagaimana jika memenangkan PGA Tour bukan tujuan Anda? Bagaimana jika Anda hanya ingin menjadi cukup baik dalam golf sehingga Anda dapat bermain dengan sopan, tidak mempermalukan diri sendiri, bersenang-senang, dan mungkin memiliki kesempatan besar untuk memenangkan turnamen klub negara lokal Anda? Itu masalah lain sama sekali. Penguasaan kelas dunia mungkin membutuhkan sepuluh ribu jam semangat terfokus, tetapi mengembangkan kapasitas untuk tampil cukup baik untuk tujuan Anda sendiri biasanya membutuhkan investasi yang jauh lebih sedikit. Itu bukan untuk mengabaikan nilai dari apa yang disebut Ericsson sebagai "praktik yang disengaja": berlatih secara sengaja dan sistematis untuk meningkatkan keterampilan. Latihan yang disengaja adalah inti dari perolehan keterampilan. Pertanyaannya adalah berapa banyak latihan yang disengaja diperlukan untuk mencapai tujuan Anda. Biasanya, itu jauh lebih sedikit dari yang Anda pikirkan.

Kualitas, Bukan Kuantitas

Merangkul gagasan ketenangan adalah kunci untuk perolehan keterampilan yang cepat. Dalam buku ini, kita akan membahas pengembangan kapasitas, bukan penguasaan kelas dunia. Kita akan mengatasi bagian curam dari kurva pembelajaran dan mendakinya secepat mungkin. Serahkan sepuluh ribu jam kepada para profesional. Kita akan mulai dengan dua puluh jam semangat yang terkonsentrasi, cerdas, dan terfokus. Kami menembak untuk hasil yang kami hargai dengan sebagian kecil dari theeort. Anda mungkin tidak akan pernah memenangkan medali emas, tetapi Anda akan menuai hadiah yang Anda pedulikan dalam waktu yang jauh lebih singkat.

Jika Anda akhirnya memutuskan untuk menguasai keterampilan tersebut, Anda akan memiliki peluang sukses yang lebih baik jika Anda memulai dengan dua puluh jam perolehan keterampilan yang cepat. Dengan mengetahui apa yang Anda hadapi, mempelajari dasar-dasarnya, berlatih secara cerdas, dan mengembangkan rutinitas latihan, Anda akan membuat kemajuan lebih cepat dan konsisten, dan Anda akan mencapai status ahli dalam waktu singkat.

Apa itu Akuisisi Keterampilan Cepat?

Akuisisi keterampilan yang cepat adalah sebuah proses—cara untuk memecah keterampilan yang Anda coba peroleh menjadi bagian-bagian kecil, mengidentifikasi bagian mana yang paling penting, kemudian dengan sengaja mempraktikkan elemen-elemen tersebut. Sesederhana itu. Akuisisi keterampilan yang cepat memiliki empat langkah utama:

Mendekonstruksi keterampilan menjadi subketerampilan sekecil mungkin; Belajar cukup tentang setiap subskill untuk dapat berlatih secara cerdas dan mengoreksi diri selama latihan; Menghilangkan hambatan fisik, mental, dan emosional yang menghalangi latihan; Melatih subskill yang paling penting setidaknya selama dua puluh jam.

Itu saja. Akuisisi keterampilan yang cepat bukanlah ilmu roket. Anda cukup memutuskan apa yang akan dilatih, mengerjakan cara terbaik untuk berlatih, meluangkan waktu untuk berlatih, lalu berlatih sampai Anda mencapai tingkat kinerja target Anda. Tidak ada keajaiban untuk itu—hanya keajaiban yang cerdas dan strategis yang diinvestasikan dalam sesuatu yang Anda pedulikan. Dengan sedikit persiapan, Anda akan memperoleh keterampilan baru dengan cepat, dengan lebih sedikit semangat. Itu tidak berarti bahwa hasilnya akan instan. Keinginan untuk kepuasan instan adalah salah satu alasan utama orang tidak memperoleh keterampilan baru dengan cepat.

Kesalahpahaman "Matriks"

Ingat adegan di The Matrix ketika Keanu Reeves membuka matanya, berkedip beberapa kali, dan berbisik "Saya tahu kung fu"? Maaf untuk mengatakannya kepada Anda: perolehan keterampilan yang cepat tidak secepat itu. Hollywood telah merugikan kami dalam hal perolehan keterampilan. Meskipun tentu akan menyenangkan untuk dapat mempelajari cara mengemudikan helikopter Bell 212 dalam waktu tiga detik dengan mengunggah perangkat lunak langsung ke otak kita, sains saat ini jauh di belakang sains. Sampai unggahan otak menjadi kenyataan, "cepat" berarti membutuhkan waktu yang jauh lebih sedikit daripada yang biasanya dibutuhkan untuk mempelajari keterampilan jika Anda melakukan prosesnya seperti yang dilakukan kebanyakan orang: secara membabi buta, serampangan, dan tidak konsisten. Salah satu keterampilan profesional pertama yang saya peroleh adalah pengembangan web: mampu membangun situs web yang berguna dan berfungsi. Dimulai dengan situs web Angelre.com dasar pada tahun 1996, saya mengajari diri sendiri cara membaca dan menulis HTML dan CSS (lingua franca web), menggunakan Adobe Photoshop untuk mengedit gambar, membuat server web, dan memelihara sistem yang menerbitkan karya saya. Saya tidak belajar bagaimana melakukan hal-hal ini di sekolah menengah atau perguruan tinggi. Meskipun saya menyelesaikan gelar sarjana saya dalam sistem informasi bisnis, tumpang tindih antara apa yang saya pelajari di kelas dan apa yang saya lakukan sehari-hari pada dasarnya nihil. Saya memperoleh keterampilan pengembangan web dengan mencoba hal-hal secara acak dan memukulkannya saat saya berjalan. Setiap kali saya menemukan teknik atau alat baru yang menjanjikan untuk meningkatkan situs web saya atau mengurangi beban kerja saya, saya bereksperimen dengannya. Selama periode waktu yang lama, keterampilan saya meningkat. Pendekatan serampangan saya untuk memperoleh keterampilan pengembangan web melayani tujuannya: Saya mendapat pekerjaan berdasarkan keterampilan itu, dan sekarang saya menerbitkan informasi di web untuk mencari nafkah. Misi tercapai, dari satu perspektif.

Di sisi lain, saya mempelajari segalanya dengan cara yang sulit. Anda pasti bisa mencapai tingkat kompetensi saya dalam keterampilan ini dalam waktu kurang dari tujuh belas tahun jika Anda mendekati topik ini dengan cara yang sistematis. Jika Anda mempraktikkan keterampilan ini dengan cerdas, Anda dapat mendekati tingkat kompetensi umum saya dalam sebulan. Itulah yang saya maksud dengan akuisisi keterampilan yang cepat. Jika Anda dapat mempelajari sebagian besar dari apa yang saya ketahui tentang desain web dalam satu bulan yang terfokus dibandingkan dua belas tahun, itu adalah peningkatan besar. Ini juga dalam ranah kemungkinan. Jumlah waktu yang Anda perlukan untuk memperoleh keterampilan baru sebagian besar adalah masalah berapa banyak waktu terkonsentrasi yang ingin Anda investasikan dalam latihan yang disengaja dan eksperimen cerdas dan seberapa baik Anda harus menjadi untuk tampil pada tingkat yang Anda inginkan. Jangan mengharapkan hasil dalam semalam. Perkirakan bahwa total waktu yang Anda investasikan akan banyak, jauh lebih sedikit daripada jika Anda terjun ke dalam proses tanpa strategi. Sebelum kita mengeksplorasi metode secara rinci, ada sesuatu yang harus Anda ketahui: perolehan keterampilan yang cepat tidak ada kesamaannya dengan bagaimana Anda "belajar cara belajar" di sekolah. Pembelajaran akademik dan kredensial hampir tidak tumpang tindih dengan perolehan keterampilan, apalagi mencapainya dengan cepat.

Akuisisi Keterampilan vs. Pembelajaran

Seperti banyak siswa sekolah menengah di Amerika Serikat, saya belajar bahasa asing. Setiap hari sekolah selama empat tahun, saya duduk di kelas bahasa Spanyol. Nilai saya tinggi: straight As. Today, selain mengatakan hola, cómo estás, dan muy bien, saya tidak dapat mengadakan percakapan dengan penutur asli bahasa Spanyol untuk menyelamatkan hidup saya. (Saya bahkan tidak tahu harus berkata apa jika saya tidak mengalami hari yang baik.) Di sisi lain spektrum, teman saya, Carlos Miceli, tumbuh besar berbicara bahasa Spanyol di Argentina. Di sekolah menengah, Carlos memutuskan dia ingin berbicara bahasa Inggris, jadi dia berusaha untuk menyerang sebagai

banyak percakapan sebisa mungkin dengan penutur asli bahasa Inggris. Dalam prosesnya, dia menemukan Skype dan membuat situs webnya sendiri, sehingga dia bisa berlatih berbicara dan menulis bahasa Inggris secara teratur. Carlos tidak pernah mengambil kelas. Dia tidak tahu aturan formal tata bahasa Inggris. Dia bahkan tidak bisa memberi tahu Anda bagaimana dia tahu bahasa Inggris. Itu tidak terlalu penting. Dia dapat berbicara dan menulis bahasa Inggris dengan baik, yang benar-benar penting. Dr. Stephen Krashen, dari University of Southern California, seorang ahli di bidang pemerolehan bahasa kedua. Salah satu wawasan utama Krashen adalah bahwa pemerolehan bahasa tidak dapat dipisahkan dari pembelajaran bahasa. Di sekolah, saya belajar banyak tentang bahasa Spanyol. Saya belajar ribuan kosakata kata, konjugasi kata kerja, dan aturan tata bahasa. Saya mempelajari semua hal ini dengan cukup baik untuk lulus tes dengan yingcolors. Namun, tes itu tidak ada hubungannya dengan kemampuan saya untuk melatih keterampilan berbicara bahasa Spanyol dengan mudah dipahami dan memahami penutur asli berbicara dengan kecepatan penuh. Jika tujuan saya adalah untuk dapat berbicara bahasa Spanyol dengan baik, beberapa minggu mencoba berbicara dengan orang-orang dalam bahasa Spanyol akan menghasilkan hasil yang lebih baik daripada empat tahun sekolah. Pada saat itu, berbicara bahasa Spanyol bukanlah tujuan saya. Saya hanya ingin lulus ujian nal . Carlos, di sisi lain, melewati kelas dan mulai berlatih. Alih-alih melakukan latihan konjugasi kata kerja, Carlos mempraktikkan apa yang benar-benar penting: berkomunikasi dengan orang lain dalam hal English. In eektif dan nilai jangka panjang, pendekatan Carlos jauh lebih unggul daripada saya. Tidak ada kontes.

Nilai Sebenarnya dari Pembelajaran

Itu tidak berarti belajar tentang keterampilan yang Anda peroleh tidak penting. Belajar bisa sangat penting, tetapi tidak menghalangi

Anda akan mengharapkan. Mempelajari konsep yang terkait dengan keterampilan membantu Anda mengoreksi diri sendiri saat Anda berlatih. Jika Anda tahu cara mengkonjugasi kata kerja dalam bahasa Spanyol, Anda lebih mampu mengoreksi ucapan Anda sendiri saat berbicara dengan penutur asli. Jika Anda mempelajari kata-kata kosakata umum, Anda lebih mampu memahami apa yang dikatakan penutur asli, serta mengingat kata atau frasa yang tepat untuk digunakan ketika Anda terjebak saat berbicara. Dr. Krashen menyebutnya hipotesis monitor. Belajar membantu Anda merencanakan, mengedit, dan memperbaiki diri Anda saat Anda berlatih. Itu sebabnya belajar berharga. Masalahnya datang ketika kita mengacaukan belajar dengan perolehan keterampilan. Jika Anda ingin memperoleh keterampilan baru, Anda harus mempraktikkannya dalam konteks. Belajar meningkatkan latihan, tetapi tidak menggantikannya. Jika kinerjanya penting, belajar saja tidak pernah cukup.

Akuisisi Keterampilan vs. Pelatihan

Ada juga perbedaan besar antara perolehan keterampilan dan pelatihan. Pelatihan, dalam konteks ini, berarti meningkatkan keterampilan yang telah Anda peroleh melalui pengulangan. Itulah yang terjadi setelah Anda memperoleh keterampilan dasar jika Anda ingin terus meningkatkan. Ambil contoh lari maraton. Sebagian besar dari kita memperoleh keterampilan berlari selama masa kanak-kanak. Selain meletakkan satu kaki di depan yang lain dan tetap berdiri sampai Anda menempuh jarak 26,2 mil, tidak banyak keterampilan baru yang bisa diperoleh. Namun, ada sejumlah besar tenaga yang diperlukan untuk memperkuat tubuh Anda dan menyesuaikan diri dengan tingkat kekerasan fisik yang diperlukan untuk menyelesaikan maraton. Proses pengerahan dan penguatan itu adalah pelatihan. Semakin banyak Anda berlatih, semakin kuat Anda, dan semakin cepat Anda menyelesaikan maraton. Ada juga elemen pembelajaran yang terlibat saat berlari maraton: bagaimana mendaftar untuk berpartisipasi dalam perlombaan, bagaimana memenuhi syarat untuk

acara besar seperti Boston Marathon, mengetahui apa yang diharapkan saat Anda berlari, mondar-mandir, peralatan yang berguna, dan sebagainya. Misalnya, masalah kecil seperti gesekan antara kemeja dan kulit Anda bukanlah masalah besar jika Anda berlari 5K, jadi sebagian besar pelari tidak memikirkannya. Gesekan yang tidak perlu menjadi masalah besar saat Anda berlari sejauh 26.2 mil. Gagal mempersiapkan sebelumnya dan Anda mungkin akan mengalami masalah "puting berdarah" yang terkenal. Menyakitkan, memalukan ... dan sepenuhnya dapat dicegah. (Tidak percaya padaku? Google.)Pelatihan dan pembelajaran tentu akan membuatnya lebih mudah untuk mengalahkan balapan, tetapi itu bukan perolehan keterampilan. Tanpa jumlah tertentu akuisisi keterampilan, pelatihan tidak mungkin atau berguna. Persiapan dan pengkondisian dapat membuat beberapa bentuk perolehan keterampilan lebih mudah, tetapi mereka tidak pernah dapat menggantikan latihan. Mempelajari kembali cara berlari di tingkat dasar, bagaimanapun, adalah perolehan keterampilan. Teknik seperti ChiRunning2 membantu pelari memperoleh keterampilan bergerak dengan cara yang meminimalkan semangat dan hilangnya momentum maju di antara langkah. Dengan sedikit latihan, pelari dapat memperoleh kembali keterampilan inti berlari, yang kemudian dapat diperkuat dalam pelatihan berikutnya.

Akuisisi Keterampilan vs. Pendidikan dan Kredensial

Terlepas dari keinginan guru dan profesor yang berpikiran tinggi di seluruh dunia, metode pendidikan dan kredensial modern hampir tidak ada hubungannya dengan perolehan keterampilan. Perolehan keterampilan membutuhkan latihan keterampilan yang dimaksud. Ini membutuhkan periode konsentrasi yang berkelanjutan dan terfokus. Ini membutuhkan kreativitas, kemampuan, dan kebebasan untuk menetapkan standar kesuksesan Anda sendiri. Sayangnya, sebagian besar metode pendidikan dan kredensial modern membutuhkan kepatuhan sederhana. Tujuan utama (tetapi tidak dinyatakan) bukanlah untuk memperoleh keterampilan yang berguna, tetapi untuk mensertifikasi penyelesaian

sebagian besar seperangkat kriteria sewenang-wenang, ditetapkan oleh komite standar jauh dari siswa, untuk tujuan memvalidasi kualitas tertentu yang tampaknya dipedulikan oleh pihak ketiga. Kreativitas, kemampuan, dan kebebasan untuk bereksperimen—elemen penting dari perolehan keterampilan yang cepat—bertentangan dengan proses kredensial. Jika standarnya terlalu jelas, itu bukan standar, bukan? Sayangnya, pendidikan dan kredensial yang ketat dapat secara aktif mencegah perolehan keterampilan. Masalah utama adalah biaya peluang: jika persyaratan untuk mendapatkan kredensial sangat intens sehingga mengganggu kemampuan Anda untuk menghabiskan waktu mempraktikkan keterampilan yang dimaksud, program kredensial dapat lebih merugikan daripada kebaikan. Ambil individu yang cerdas dan termotivasi yang tertarik untuk memulai perusahaan perangkat lunak. Menyelesaikan gelar sarjana dalam ilmu komputer di universitas bergengsi biasanya memakan waktu setidaknya empat tahun.³ Pada akhir empat tahun itu, lulusan kami yang baru dicetak telah menghabiskan ribuan jam mempelajari algoritme dan menganalisis kompilasi dengan cukup baik untuk lulus lusinan ujian, tetapi dia tidak lebih dekat untuk mendirikan perusahaan perangkat lunak daripada ketika dia memasuki universitas. Siswa kami yang malang telah menghafal banyak hal tentang pemrograman komputer, setidaknya untuk sementara, tetapi dia masih tidak tahu bagaimana membuat program komputer yang cukup berguna untuk dibeli oleh orang. Memulai perusahaan perangkat lunak membutuhkan perolehan keterampilan baru: mempelajari bahasa pemrograman, menyiapkan dan memelihara sistem komputer, meneliti alat dan program yang tersedia, membuat prototipe, mencari pengguna awal, mendapatkan pendanaan atau pendanaan yang diperlukan, dan menangani tugas administrasi bisnis umum. Apakah ada tumpang tindih antara memulai startup dan mendapatkan kredensial pendidikan? Yakin. Tapi perhatikan penekanannya: sebagian besar

Kemampuan mendapatkan kredensial dikhususkan untuk proses memenuhi persyaratan. Apakah persyaratan tersebut benar-benar membantu Anda memperoleh keterampilan yang Anda butuhkan untuk tampil di dunia nyata adalah masalah tersier. Dalam buku pertama saya, *The Personal MBA: Master the Art of Business* (2010), saya menjelaskan mengapa saya memutuskan untuk melewati pendidikan bisnis tingkat pascasarjana demi mengajari diri saya sendiri prinsip-prinsip praktik bisnis modern dan memulai perusahaan saya sendiri. Dengan menghindari sekolah bisnis, dan menghabiskan waktu saya untuk membangun bisnis, saya belajar banyak, dan menghemat lebih dari \$ 150.000 dalam prosesnya. Mengingat apa yang ingin saya capai, mendedikasikan waktu untuk akuisisi keterampilan bisnis sendiri lebih baik daripada sekolah bisnis dalam segala hal. Jika Anda ingin menjadi ahli dalam apa pun di mana kinerja kehidupan nyata penting, Anda harus benar-benar melatih keterampilan itu dalam konteks. Belajar, dengan sendirinya, tidak pernah cukup.

Neurofisiologi Keterampilan: Plastisitas Otak dan Memori Otak

Satu hal terakhir sebelum kita melompat ke mur dan baut perolehan keterampilan yang cepat: Anda harus sepenuhnya menghargai fakta bahwa Anda mampu memperoleh keterampilan baru. Itu tampak seperti hal yang aneh untuk dikatakan, tetapi mudah untuk percaya bahwa keterampilan Anda xed—bahwa Anda baik atau berbakat atau berbakat dalam sesuatu ... atau Anda tidak. Dalam *Mindset: The New Psychology of Success* (2007), psikolog Carol Dweck mengutip berbagai penelitian yang menunjukkan bahwa individu umumnya memiliki salah satu dari dua pandangan tentang bagaimana pikiran mereka bekerja. Menurut Dr. Dweck, orang-orang dengan pola pikir "xed" berasumsi bahwa keterampilan dan bakat adalah bawaan, bahwa Anda dilahirkan dengan kemampuan tertentu yang adanya. Jika seseorang dengan pola pikir xed "tidak pandai matematika", maka berlatih matematika lebih cepat adalah. Mengapa repot-repot jika Anda tidak akan pernah pandai dalam hal itu?

Orang dengan pola pikir "pertumbuhan", di sisi lain, berasumsi bahwa keterampilan dan kemampuan tumbuh dengan latihan dan ketekunan. Jika seseorang dengan pola pikir pertumbuhan salah dalam beberapa soal matematika, itu bukan karena mereka tidak diberkati dengan matematika yang baik; Itu karena mereka belum cukup berlatih. Dengan ketekunan dan latihan, hanya masalah waktu sebelum mereka menguasai teknik tersebut. Inilah kabar baiknya jika Anda jatuh ke dalam jebakan xedmindset: penelitian yang luas (dan terus berkembang) menunjukkan bahwa semua otak mampu meningkatkan keterampilan dan kemampuan dengan latihan. Kecenderungan genetik ada, tetapi mereka sangat kecil dibandingkan dengan kekuatan praktik yang terfokus dan cerdas. Anda dapat meningkatkan keterampilan apa pun, asalkan Anda bersedia berlatih. Otak manusia adalah plastik—istilah yang digunakan oleh ilmuwan saraf untuk menunjukkan bahwa otak Anda berubah secara fisik sebagai respons terhadap lingkungan Anda, tindakan Anda, dan konsekuensi dari tindakan tersebut. Saat Anda mempelajari keterampilan baru, fisik atau mental, kabel neurologis otak Anda berubah saat Anda mempraktikkannya. Dalam kata-kata Dr. Jon Medina (Brain Rules, 2009) "neuron yang menyatukan penyambungan," membentuk pola baru yang unik dalam sirkuit fisik otak Anda. Seiring waktu, neuron Anda mulai robek dalam pola yang lebih kuno sebagai respons terhadap umpan balik yang Anda terima dari lingkungan Anda saat Anda berlatih. Jika Anda melatih keterampilan motorik (yaitu, keterampilan yang melibatkan gerakan fisik), Anda selalu relatif canggung dan lambat. Anda harus memikirkan semua yang Anda lakukan, dan Anda sering membuat kesalahan yang membuat frustrasi. Mempelajari dasar-dasarnya adalah perjuangan yang terus-menerus. Saat Anda berlatih, koordinasi otot Anda menjadi lebih otomatis dan disinkronkan dengan proses mental Anda. Anda memperoleh kemampuan untuk lebih memperhatikan elemen halus dari apa yang Anda lakukan, dan Anda belajar menyesuaikan pendekatan Anda dengan umpan balik yang Anda dapatkan dari lingkungan.

Anda mulai melakukan lebih banyak dari apa yang berhasil, dan lebih sedikit dari apa yang tidak. Pada akhirnya, Anda dapat tampil tanpa memperhatikan setiap detail. Dalam literatur akademis, proses umum ini disebut "model tiga tahap" perolehan keterampilan,⁴ dan berlaku untuk keterampilan fisik dan mental. Ketiga tahap tersebut adalah

1. Tahap Kognitif (Awal)—memahami apa yang Anda coba lakukan, meneliti, memikirkan prosesnya, dan memecah keterampilan menjadi bagian-bagian yang dapat dikelola.
2. Tahap Asosiatif (Menengah)—mempraktikkan tugas, memperhatikan umpan balik lingkungan, dan menyesuaikan pendekatan Anda berdasarkan umpan balik tersebut.
3. Tahap Otonom (Akhir)—melakukan keterampilan secara tepat dan kuno tanpa memikirkannya atau memperhatikan proses yang tidak perlu.

Proses perolehan keterampilan neurofisiologis ini terjadi sepanjang waktu, bahkan saat Anda membaca kalimat ini. Tidak ada yang namanya pikiran dalam stasis. Otak Anda belajar, mengkodekan, dan mengkonsolidasikan keterampilan baru sepanjang waktu. Seperti yang dikatakan Dr. Dweck dalam *Mindset*: "Pikiran Anda seperti otot: semakin Anda menggunakannya, semakin tumbuh." Semakin banyak Anda berlatih, semakin kuno, efektif, dan otomatis keterampilannya. Itu kabar bagus dalam hal akuisisi keterampilan yang cepat. Jika pikiran dan tubuh Anda mampu belajar untuk tampil dengan cara yang baru dan lebih baik, kita dapat menemukan cara membuat proses itu lebih cepat.



2

Sepuluh Prinsip Akuisisi Keterampilan Cepat

Saya menyadari bahwa menjadi ahli karate bukan tentang mempelajari 4.000 gerakan, tetapi hanya melakukan beberapa gerakan 4.000 kali.—CHET HOLMES, PENULIS THE ULTIMATE SALES MACHINE

■ ■ ■

Sekarang setelah kita jelas tentang apa sebenarnya arti akuisisi keterampilan, mari kita periksa cara melakukannya dengan cepat. Tujuan dari bab ini adalah untuk memberi Anda daftar periksa praktis untuk memperoleh keterampilan baru apa pun. Saya merasa berguna untuk memikirkan prinsip-prinsip ini sebagai cara untuk menumbuhkan "obsesi sementara." Akuisisi keterampilan yang cepat terjadi secara alami ketika Anda menjadi sangat ingin tahu dan tertarik pada sesuatu sehingga masalah lain hilang, setidaknya untuk sementara. Pikirkan prinsip-prinsip ini sebagai cara untuk mengidentifikasi keterampilan yang layak untuk obsesi sementara, fokus padanya, dan menghilangkan gangguan atau penghalang yang mengalihkan perhatian Anda dari latihan efektif. Berikut adalah sepuluh prinsip utama perolehan keterampilan yang cepat:

1. Pilih proyek yang menyenangkan.
2. Fokuskan energi Anda pada satu keterampilan pada satu waktu.
3. Tentukan tingkat kinerja target Anda.
4. Dekonstruksi keterampilan menjadi subketerampilan.
5. Dapatkan alat penting.
6. Hilangkan hambatan untuk berlatih.
7. Luangkan waktu khusus untuk latihan.

8. Buat loop umpan balik yang cepat.
9. Berlatihlah dengan jam dalam semburan singkat.
10. Tekankan kuantitas dan kecepatan.

Banyak dari prinsip-prinsip ini mungkin membuat Anda masuk akal, dan tidak apa-apa. Ingat: hanya mengetahui prinsip-prinsip ini tidak cukup. Anda harus benar-benar menggunakannya untuk menuai hasilnya.

1. Pilih proyek yang menyenangkan. Karl Popper adalah salah satu filsuf terbesar abad kedua puluh. Dia adalah orang yang mempopulerkan gagasan pemalsuan ilmiah. Dalam istilah awam, jika Anda tidak berpotensi membuktikan sesuatu yang salah melalui pengamatan atau eksperimen, itu sebenarnya bukan sains. Popper mengatakan banyak hal bijaksana, tetapi saya pikir komentar berikut adalah salah satu yang paling bijaksana: "Hal terbaik yang dapat terjadi pada manusia adalah menemukan masalah, jatuh cinta dengan masalah itu, dan hidup mencoba memecahkan masalah itu, kecuali masalah lain yang bahkan lebih dicintai muncul." Jika Anda menginginkan formula untuk menjalani kehidupan yang memuaskan dan produktif, Anda tidak bisa salah dengan yang itu. Akuisisi keterampilan yang cepat membutuhkan pemilihan masalah atau proyek yang menyenangkan. Semakin bersemangat Anda tentang keterampilan yang ingin Anda peroleh, semakin cepat Anda akan memperolehnya. Dalam praktiknya, membuat proyek yang menyenangkan adalah masalah yang sangat individual. Misalnya, belajar berbicara dan menulis bahasa Mandarin tidak ada dalam daftar keterampilan saya saat ini untuk diperoleh karena saya tidak memiliki kebutuhan mendesak untuk mempelajarinya saat ini, dan saya memiliki banyak proyek lain yang lebih tertarik untuk saya tangani. Jika saya memutuskan untuk pindah ke bagian Tiongkok yang berbahasa Mandarin di masa depan, itu mungkin menjadi menyenangkan, tetapi saya belum sampai di sana. Di sisi lain, saya sangat tertarik untuk mempelajari cara memainkan Go, permainan papan strategis tertua di dunia, yang berasal dari

di Cina lebih dari tiga ribu tahun yang lalu. Ini adalah permainan yang indah, dan saya ingin belajar cara bermain sejak saya menemukannya bertahun-tahun yang lalu. Belajar bermain Go membutuhkan studi. Aturannya sederhana, tetapi secara akurat membaca pola yang berkembang dari batu hitam dan putih bergantian di papan adalah sebuah tantangan. Komputer telah mendominasi catur selama bertahun-tahun sekarang, tetapi bahkan komputer terbaik pun memiliki waktu tambahan untuk menantang pemain Go manusia yang berpengalaman. Anda secara alami mempelajari hal-hal yang Anda pedulikan lebih cepat daripada hal-hal yang tidak Anda sayangi. Saat ini saya lebih tertarik untuk mempelajari cara bermain Go, jadi saya akan belajar Go rst, dan menyimpan bahasa Mandarin untuk nanti. Jika Anda fokus untuk memperoleh keterampilan utama Anda (yaitu, proyek Anda yang paling disukai) sebelum hal lain, Anda akan mendapatkannya dalam waktu yang jauh lebih singkat.

2. Fokuskan energi Anda pada satu keterampilan pada satu waktu. Salah satu kesalahan termudah yang dilakukan saat memperoleh keterampilan baru adalah mencoba memperoleh terlalu banyak keterampilan pada saat yang bersamaan. Ini masalah matematika sederhana: memperoleh keterampilan baru membutuhkan massa kritis dari waktu terkonsentrasi dan perhatian terfokus. Jika Anda hanya memiliki satu atau dua jam setiap hari untuk dicurahkan untuk berlatih dan belajar, dan Anda menyebarkan waktu dan energi itu ke dua puluh keterampilan yang berbeda, tidak ada keterampilan individu yang akan menerima cukup waktu dan energi untuk menghasilkan peningkatan yang nyata. Menginternalisasi prinsip ini lebih sulit bagi sebagian orang daripada yang lain. Secara pribadi, saya selalu memiliki temperamen "pria Renaisans": ada ratusan hal yang ingin saya pelajari pada saat tertentu, di ratusan daerah yang berbeda. Secara emosional, sulit bagi saya untuk memutuskan untuk menunda mempelajari hal-hal baru yang saya temukan atau dengar.¹ Namun, ketika saya mencoba mempelajari semuanya sekaligus, saya tidak benar-benar belajar apa pun. Alih-alih membuat kemajuan, saya menghabiskan terlalu banyak waktu

beralih di antara keterampilan yang berbeda, frustrasi, dan beralih ke hal lain. Itu resep untuk perolehan keterampilan yang sangat lambat. Pilih satu, dan hanya satu, keterampilan baru yang ingin Anda peroleh. Tempatkan semua fokus dan energi cadangan Anda untuk memperoleh keterampilan itu, dan tunda keterampilan lainnya. David Allen, penulis *Getting Things Done* (2002), merekomendasikan untuk membuat apa yang dia sebut daftar "suatu hari nanti/mungkin": daftar hal-hal yang mungkin ingin Anda jelajahi di masa depan, tetapi itu tidak cukup penting untuk difokuskan saat ini. Dengan menambahkan item ke daftar, Anda untuk sementara membebaskan diri dari tanggung jawab untuk bertindak atau memikirkan ide tersebut sampai Anda memutuskan untuk mempromosikannya ke status aktif. Saya tidak bisa cukup menekankan hal ini. Berfokus pada satu keterampilan utama pada satu waktu mutlak diperlukan untuk perolehan keterampilan yang cepat. Anda tidak menyerah pada keterampilan lain secara permanen, Anda hanya menyimpannya untuk nanti.

3. Berikan target tingkat kinerja Anda. Tingkat kinerja target adalah kalimat sederhana yang terlihat seperti "Innog yang bagus" Dennis. Seberapa baik Anda ingin dapat melakukan keterampilan yang Anda peroleh? Tingkat Kinerja Target Anda adalah Pernyataan Singkat TENTANG Seperti Apa Tingkat Keterampilan YourSided. Pikirkan itu SA kalimat tunggal tentang apa yang ingin Anda capai, dan apa yang dapat Anda lakukan setelah selesai. Semakin spesifik tingkat kinerja target Anda, semakin baik. Menentukan tingkat kinerja target Anda membantu Anda membayangkan bagaimana rasanya tampil dengan cara tertentu. O.N.C.E. Anda menentukan dengan tepat baik yang Anda inginkan atau perlu, mudah untuk keluar dari sini. Dalam Kata-kata Charles Kettering, Penemu Sistem Pengapian Mobil Listrik: "Masalah yang dinyatakan dengan baik adalah masalah yang setengah terpecahkan."

Bagaimana Anda menentukan tingkat kinerja target Anda tergantung pada mengapa Anda memilih untuk memperoleh keterampilan di tempat pertama. Jika niat Anda adalah untuk bersenang-senang, target Anda adalah titik di mana Anda berhenti merasa frustrasi dan mulai menikmati latihan itu sendiri. Jika niat Anda adalah untuk berkinerja, berapa tingkat kinerja minimum yang ingin Anda terima di rst? Setelah Anda mencapai tingkat kinerja target awal Anda, Anda selalu dapat memilih untuk terus maju jika Anda mau. Tingkat kinerja target terbaik tampaknya berada di luar jangkauan, bukan di luar kemungkinan. Sebagai aturan, semakin santai tingkat kinerja target Anda, semakin cepat Anda dapat memperoleh keterampilan terkait. Jika Anda beroperasi kembali di bawah pola pikir penguasaan kelas dunia, ini mungkin terasa seperti curang: Anda hanya menurunkan standar sehingga Anda bisa "menang" lebih cepat, bukan? Itulah yang kami lakukan, dan itu bukan kecurangan. Ingat, penguasaan kelas dunia bukanlah titik akhir dari perolehan keterampilan yang cepat. Kami memotret untuk kapasitas dan kesucian pada kecepatan maksimum, bukan kesempurnaan. Penting untuk dicatat bahwa beberapa keterampilan memiliki pertimbangan keselamatan, yang harus selalu Anda sertakan dalam tingkat kinerja target Anda. Terluka (atau terbunuh) memperoleh keterampilan baru mengalahkan tujuannya.

4. Dekonstruksi keterampilan menjadi subskill. Sebagian besar hal yang kita anggap sebagai keterampilan sebenarnya adalah bundel subskill yang lebih kecil. Setelah Anda mengidentifikasi keterampilan untuk difokuskan, langkah selanjutnya adalah mendekonstruksinya—memecahnya menjadi bagian-bagian terkecil. Misalnya, bermain golf adalah keterampilan yang memiliki banyak subkomponen: memilih tongkat yang benar, mengemudi o tee, memukul keluar dari bunker, putting, dan sebagainya. Setelah keterampilan didekonstruksi secara suicient, jauh lebih mudah untuk mengidentifikasi subskill mana yang tampaknya paling penting. Dengan berfokus pada subskill kritis rst, Anda akan membuat lebih banyak kemajuan dengan lebih sedikit eort.

Mendekonstruksi keterampilan juga membuatnya lebih mudah untuk menghindari perasaan kewalahan. Anda tidak harus melatih semua bagian keterampilan secara bersamaan. Sebaliknya, lebih efektif untuk fokus pada subskill yang menjanjikan pengembalian keseluruhan yang paling dramatis. Mendekonstruksi keterampilan sebelum Anda memulai juga memungkinkan Anda untuk mengidentifikasi bagian-bagian keterampilan yang tidak penting bagi praktisi pemula. Dengan menghilangkan subskill atau teknik nonkritis di awal proses, Anda akan dapat menginvestasikan lebih banyak waktu dan energi Anda untuk menguasai subskill kritis.

5. Dapatkan alat penting. Sebagian besar keterampilan memiliki prasyarat untuk berlatih dan kinerja. Sangat sulit untuk bermain tenis jika Anda tidak memiliki raket tenis, atau belajar cara mengemudikan helikopter jika Anda tidak memiliki akses ke sana. Alat, komponen, dan lingkungan apa yang perlu Anda akses sebelum Anda dapat berlatih secara kuno? Bagaimana Anda bisa mendapatkan alat terbaik yang Anda bisa dan aord? Meluangkan waktu sejenak untuk mengidentifikasi alat penting sebelum Anda mulai berlatih menghemat waktu yang berharga. Dengan memastikan Anda memiliki sumber daya yang Anda butuhkan sebelum memulai, Anda memaksimalkan waktu latihan Anda.

6. Hilangkan hambatan untuk berlatih. Ada banyak hal yang bisa menghalangi latihan, yang membuatnya jauh lebih sulit untuk memperoleh keterampilan apa pun. Hambatan ini bisa apa saja dari

Upaya pralatihan yang signifikan. Seperti salah menempatkan alat Anda, tidak mendapatkannya alat yang benar sebelum berlatih, atau melewatkan persyaratan penyiapan. Ketersediaan sumber daya terputus-putus. Seperti menggunakan peralatan pinjaman atau mengandalkan sumber daya yang memiliki jam operasional terbatas. Gangguan lingkungan. Seperti televisi, telepon berdering, dan email masuk. Blok emosional. Seperti ketakutan, keraguan, dan rasa malu.

Setiap elemen ini membuatnya lebih sulit untuk mulai berlatih, dan karenanya menurunkan kecepatan akuisisi Anda. Mengandalkan kemauan untuk secara konsisten mengatasi hambatan ini adalah strategi yang merugikan. Kita hanya memiliki begitu banyak kemauan yang kita miliki setiap hari, dan yang terbaik adalah menggunakan kemauan itu dengan bijak. Cara terbaik untuk menginvestasikan kemauan dalam mendukung perolehan keterampilan adalah menggunakannya untuk menghilangkan hambatan lunak untuk berlatih ini. Dengan mengatur ulang lingkungan Anda untuk membuatnya semudah mungkin untuk mulai berlatih, Anda akan memperoleh keterampilan dalam waktu yang jauh lebih singkat.

7. Luangkan waktu khusus untuk latihan. Waktu yang Anda habiskan untuk memperoleh keterampilan baru harus berasal dari suatu tempat. Sayangnya, kita cenderung ingin memperoleh keterampilan baru dan terus melakukan banyak aktivitas lain yang kita sukai, seperti menonton TV, bermain video game, dan sebagainya. Saya akan melakukannya, ketika saya tiba waktunya, kami berkata pada diri kami sendiri. Inilah kebenarannya: waktu "nding" adalah mitos. Tidak ada yang pernah "nd" waktu untuk apa pun, dalam arti secara ajaib menemukan beberapa bank waktu ekstra, seperti mencabut uang dua puluh dolar yang secara tidak sengaja Anda tinggalkan di saku mantel Anda. Jika Anda mengandalkan waktu untuk melakukan sesuatu, itu tidak akan pernah dilakukan. Jika Anda ingin mendapatkan waktu, Anda harus meluangkan waktu. Anda memiliki 24 jam untuk berinvestasi setiap hari: 1.440 menit, tidak lebih atau kurang. Anda tidak akan pernah punya waktu lagi. Jika Anda tidur sekitar 8 jam sehari, Anda memiliki 16 jam yang Anda inginkan. Beberapa jam itu akan digunakan untuk merawat diri sendiri dan orang yang Anda cintai. Lainnya akan digunakan untuk bekerja. Apa pun yang tersisa adalah waktu yang Anda miliki untuk memperoleh keterampilan. Jika Anda ingin meningkatkan keterampilan Anda secepat mungkin, semakin besar blok waktu khusus yang dapat Anda sisihkan, semakin baik. Pendekatan terbaik untuk meluangkan waktu untuk akuisisi keterampilan adalah mengidentifikasi penggunaan waktu yang bernilai rendah, lalu memilih untuk menghilangkannya. Sebagai

percobaan, saya sarankan untuk menyimpan catatan sederhana tentang bagaimana Anda menghabiskan waktu Anda selama beberapa hari. Yang Anda butuhkan hanyalah buku catatan. Hasil dari catatan waktu ini akan mengejutkan Anda: jika Anda membuat beberapa pilihan sulit untuk memotong penggunaan waktu yang bernilai rendah, Anda akan memiliki lebih banyak waktu untuk memperoleh keterampilan. Semakin banyak waktu yang harus Anda curahkan setiap hari, semakin sedikit total waktu yang dibutuhkan untuk memperoleh keterampilan baru. Saya merekomendasikan meluangkan waktu untuk setidaknya sembilan puluh menit latihan setiap hari dengan memotong aktivitas bernilai rendah sebanyak mungkin. Saya juga merekomendasikan untuk berkomitmen untuk menyelesaikan setidaknya dua puluh jam latihan. Setelah Anda mulai, Anda harus terus berlatih sampai Anda mencapai tanda dua puluh jam. Jika Anda terjebak, teruslah mendorong: Anda tidak dapat berhenti sampai Anda mencapai tingkat kinerja target Anda atau menginvestasikan dua puluh jam. Jika Anda tidak bersedia menginvestasikan setidaknya dua puluh jam di muka, pilih keterampilan lain untuk diperoleh. Alasannya sederhana: bagian awal dari proses perolehan keterampilan biasanya terasa lebih sulit daripada yang sebenarnya. Anda sering bingung, dan Anda akan menghadapi masalah dan hambatan yang tidak terduga. Alih-alih menyerah ketika Anda mengalami sedikit pun kesulitan, berkomitmen sebelumnya untuk dua puluh jam membuatnya lebih mudah untuk bertahan. Pikirkan pendekatan ini sebagai latihan ketabahan: Anda tidak akan membiarkan beberapa masalah kecil yang konyol menghentikan Anda dari melakukan apa yang telah Anda putuskan benar-benar ingin Anda lakukan. Anda akan memecahkan masalah, atau melakukan yang terbaik sampai Anda mencapai tanda dua puluh jam. Pada saat itu, Anda akan berada dalam posisi yang lebih baik untuk memutuskan bagaimana melanjutkan.

8. Buat loop umpan balik yang cepat." Umpan balik cepat" berarti mendapatkan informasi yang akurat tentang seberapa baik kinerja Anda secepat mungkin. Semakin lama waktu yang dibutuhkan untuk mendapatkan umpan balik yang akurat, semakin lama waktu yang dibutuhkan untuk memperoleh keterampilan. Ambil seni membuat keju, misalnya. Proses kimia halus yang membuat keju ne seringkali membutuhkan waktu berbulan-bulan atau bertahun-tahun untuk diselesaikan, dan tidak ada cara untuk terburu-buru prosesnya tanpa merusak

hasilnya. Jika dibutuhkan enam bulan untuk menentukan apakah keju Anda bagus atau tidak, keterlambatan umpan balik membuatnya sulit untuk memperoleh keterampilan dengan cepat. Umpan balik yang cepat secara alami mengarah pada akuisisi keterampilan yang cepat. Jika umpan balik segera tiba, atau dengan penundaan yang sangat singkat, jauh lebih mudah untuk menghubungkan informasi tersebut dengan tindakan Anda dan membuat penyesuaian yang sesuai. Bentuk umpan balik terbaik hampir seketika. Itulah sebabnya keterampilan seperti pemrograman bisa menjadi sedikit membuat ketagihan: Anda membuat perubahan, dan beberapa milidetik kemudian komputer memberi tahu Anda apakah itu berhasil atau tidak. Jika Anda tidak menyukai umpan balik ("program saya mogok!"), buat perubahan lain dan coba lagi. Ada banyak sumber potensial umpan balik yang berguna. Seperti yang dijelaskan oleh Atul Gawande, ahli bedah veteran dan pemain tenis amatir, dalam artikel di *The New Yorker*,² pelatih dan mentor berpengalaman dapat memberi Anda umpan balik langsung tentang kinerja Anda dan merekomendasikan penyesuaian yang diperlukan. Pelatih bukan satu-satunya sumber umpan balik yang cepat. Perangkat tangkap, seperti kamera video, dapat membantu Anda menonton diri sendiri saat Anda tampil. Alat seperti program komputer, asisten pelatihan, dan perangkat lain dapat segera menunjukkan ketika Anda membuat kesalahan atau sesuatu yang salah. Semakin banyak sumber umpan balik cepat yang Anda integrasikan ke dalam latihan Anda, semakin cepat Anda akan memperoleh keterampilan.

9. Berlatihlah dengan jam dalam semburan singkat. Pikiran kita dibangun untuk belajar—untuk memperhatikan pola, mensimulasikan tindakan potensial, dan mengungkapkan apa yang mungkin akan terjadi selanjutnya. Mereka tidak dibuat untuk memperkirakan waktu secara akurat—berapa lama waktu yang dibutuhkan sesuatu, atau berapa banyak waktu yang Anda habiskan untuk melakukan sesuatu.

Pada fase awal berlatih keterampilan baru, sangat mudah untuk melebih-lebihkan berapa banyak waktu yang Anda habiskan untuk berlatih. Ketika Anda tidak baik (dan Anda tahu itu), waktu tampaknya melambat untuk merangkak, dan rasanya seperti Anda telah berlatih untuk jangka waktu yang lebih lama daripada yang sebenarnya. Solusinya adalah berlatih dengan jam. Beli penghitung waktu mundur yang layak³ dan atur selama dua puluh menit. Hanya ada satu aturan: setelah Anda memulai pengatur waktu, Anda harus berlatih sampai habis. Teknik sederhana ini akan memudahkan untuk menyelesaikan latihan berkelanjutan yang lebih lama, bahkan ketika Anda lelah atau frustrasi. Semakin banyak periode latihan berkelanjutan yang Anda selesaikan, semakin cepat perolehan keterampilan Anda. Sisihkan waktu untuk tiga sesi latihan sehari, dan Anda akan melihat kemajuan besar dalam waktu yang sangat singkat.

10. Tekankan kuantitas dan kecepatan. Ketika Anda mulai memperoleh keterampilan baru, tergoda untuk fokus pada latihan dengan sempurna—resep untuk frustrasi. Penampilan Anda, tentu saja, tidak akan mendekati kesempurnaan. Alih-alih mencoba menjadi sempurna, fokuslah untuk berlatih sebanyak yang Anda bisa secepat mungkin, sambil mempertahankan bentuk yang "cukup baik". Dalam *Art & Fear* (2001), penulis David Bayles dan Ted Orland berbagi anekdot yang sangat menarik tentang nilai volume:

Guru keramik mengumumkan pada hari pembukaan bahwa dia membagi kelas menjadi dua kelompok. Semua yang berada di sisi kiri studio, katanya, akan dinilai hanya pada jumlah karya yang mereka hasilkan, semua yang berada di sebelah kanan hanya pada kualitasnya. Prosedurnya sederhana: pada hari pertama kelas, dia akan membawa timbangan kamar mandinya dan menimbang pekerjaan kelompok "kuantitas": fty pound pot diberi peringkat A, empat puluh pound B, dan seterusnya. Namun, mereka yang dinilai berdasarkan "kualitas", hanya perlu menghasilkan satu pot—meskipun yang sempurna—untuk mendapatkan nilai A. Nah, pada saat penilaian, sebuah fakta aneh muncul: karya-karya dengan kualitas tertinggi semuanya diproduksi oleh kelompok yang dinilai berdasarkan kuantitas. Tampaknya sementara kelompok "kuantitas" sibuk menghasilkan tumpukan pekerjaan dan belajar dari kesalahan mereka, kelompok "kualitas" telah duduk berteori tentang kesempurnaan, dan di

Akhir tidak memiliki lebih banyak hal untuk ditunjukkan untuk eorts mereka daripada teori-teori megah dan apile of dead clay.

Keterampilan adalah hasil dari latihan yang disengaja dan konsisten, dan dalam latihan tahap awal, kuantitas dan kecepatan mengalahkan kualitas mutlak. Semakin cepat dan lebih sering Anda berlatih, semakin cepat Anda akan memperolehnya. Itu tidak berarti bahwa Anda harus mengabaikan bentuk yang baik saat berlatih. Beberapa keterampilan, terutama keterampilan yang membutuhkan tindakan fisik atau gerakan, membutuhkan kualitas bentuk tertentu untuk tampil baik. Jika Anda mempraktikkan teknik melukis Anda, menggunakan Jackson Pollackon seratus kanvas dalam sehari tidak akan membantu Anda jika Anda melukis potret yang hidup. Teknik itu penting. Pertama, pastikan Anda berlatih menggunakan bentuk yang cukup baik untuk memenuhi tingkat kinerja target Anda. Setelah Anda berlatih dalam bentuk yang tidak baik setidaknya 80 hingga 90 persen dari waktu, tingkatkan kecepatan untuk memperoleh keterampilan yang lebih cepat. Itu saja: sepuluh prinsip sederhana yang akan memastikan Anda berlatih keterampilan utama Anda dengan cara yang paling kuno dan efektif.

Jadi apakah itu berhasil?

Akankah metode ini benar-benar membantu Anda memperoleh keterampilan lebih cepat? Penelitian mengatakan tentu saja. Dalam studi akademis tentang akuisisi keterampilan kognitif dan motorik, para peneliti telah memperhatikan pola umum: ketika peserta penelitian mulai mempraktikkan keterampilan baru, kinerja mereka selalu meningkat secara dramatis dalam waktu yang sangat singkat. Tidak perlu banyak latihan sama sekali untuk beralih dari "sangat lambat dan sangat tidak kompeten" menjadi "cukup cepat dan terasa kompeten." Dalam literatur, ini disebut sebagai "hukum kekuasaan praktik", dan muncul berulang kali. Eect telah dikenal luas di kalangan peneliti akuisisi keterampilan setidaknya sejak tahun 1926,4 dan

itu telah direplikasi berkali-kali sejak dalam studi keterampilan fisik dan mental.⁵ Satu studi bahkan melangkah lebih jauh dengan mengatakan "setiap teori perolehan keterampilan yang tidak mengakomodasi fungsi hukum kekuasaan untuk belajar dapat segera ditolak."⁶ Studi akademis menggambar kurva "hukum praktek kekuatan" seperti ini, dengan waktu kinerja pada sumbu y dan peristiwa latihan pada sumbu x:



Karena waktu adalah kuantitas yang meningkat, kurva miring ke bawah. Dengan latihan, dibutuhkan lebih sedikit waktu untuk menyelesaikan tugas tertentu. Menarik untuk dicatat bahwa jika Anda melabeli ulang sumbu y sebagai "seberapa baik Anda" (yaitu, Anda menolak kinerja dalam istilah yang lebih umum versus satuan waktu), Anda mendapatkan kurva pembelajaran yang terkenal dan dikenal luas:



Pola umum kurva belajar terlihat seperti ini: Ketika Anda memulai, Anda mengerikan, tetapi Anda meningkat dengan sangat cepat saat Anda mempelajari bagian terpenting dari keterampilan tersebut. Setelah mencapai tingkat keterampilan tertentu dengan sangat cepat, tingkat peningkatan Anda menurun, dan peningkatan selanjutnya menjadi jauh lebih lambat. Bertentangan dengan penggunaan populer, "kurva belajar yang curam" itu baik, tidak buruk. Grafik memperjelas alasannya: Kurva belajar yang curam menunjukkan tingkat perolehan keterampilan yang sangat cepat. Semakin curam kurvanya, semakin baik Anda mendapatkan per satuan waktu. Anda dapat menganggap daftar periksa yang baru saja saya uraikan sebagai cara untuk dengan sengaja membuat kurva belajar pribadi Anda lebih curam. Prinsip-prinsip itu sendiri adalah teknik sederhana yang membuat dua tahap teoretis pertama dari proses perolehan keterampilan (kognisi dan asosiasi) lebih mudah dilakukan dalam praktik. Begitu Anda mulai berlatih sesuatu yang baru, keterampilan Anda secara alami dan nyata akan meningkat dalam waktu yang sangat singkat. Triknya adalah mulai berlatih secepat mungkin. Tidak berpikir untuk berlatih atau khawatir tentang berlatih, tetapi benar-benar berlatih. Terlalu mudah untuk merasa seperti Anda menginvestasikan banyak waktu dalam suatu keterampilan tanpa banyak berlatih sama sekali. Jika Anda ingin mempelajari sesuatu untuk waktu yang lama, Anda bermimpi menjadi ahli dalam hal itu, tetapi Anda ragu untuk memulai, Anda dapat menghabiskan energi mental dan emosional selama bertahun-tahun tanpa meningkat sedikit pun. Jika Anda tidak tahu ke mana Anda mencoba pergi atau tidak memiliki strategi yang solid untuk mencapainya, Anda dapat membuang-buang energi dalam jumlah yang sama dalam pengembaraan yang tidak produktif. Sepuluh prinsip ini dirancang untuk membantu Anda menghilangkan pukulan yang tidak produktif ini dan menggantinya dengan aktivitas yang mendasar untuk proses perolehan keterampilan. Semakin banyak waktu dan energi yang Anda habiskan untuk bergerak melalui dua fase pertama dari proses perolehan keterampilan dan semakin sedikit waktu yang Anda habiskan untuk melakukan hal-hal yang tidak membantu Anda, semakin cepat Anda akan memperoleh keterampilan tersebut. Sesederhana itu.

Bagaimana dengan Immersion?

Ini bukan satu-satunya cara untuk memperoleh keterampilan baru, tetapi tentu saja ini yang paling jelas. Metode lain dapat menghasilkan hasil yang serupa, tetapi mereka membutuhkan perdagangan yang lebih signifikan. Metode umum yang paling terkenal untuk akuisisi keterampilan cepat adalah perendaman: benar-benar mengubah lingkungan Anda dengan cara yang menghasilkan latihan yang disengaja secara konstan. Jika Anda ingin belajar berbicara bahasa Prancis, misalnya, belajar melalui pencelupan akan melibatkan tinggal di Prancis selama beberapa minggu atau bulan. Secara umum, perendaman berfungsi. Jika Anda pindah ke Prancis, Anda akan dipaksa untuk melatih keterampilan berbicara Anda setiap saat setiap hari selama Anda berada di sana. Setelah beberapa hari yang membuat frustrasi beradaptasi dengan lingkungan baru Anda, Anda akan melihat keterampilan Anda meningkat dengan cepat. Perendaman bekerja karena memastikan bahwa Anda menyelesaikan jam-jam latihan terpenting tanpa gagal: Anda tidak dapat melarikan diri dari lingkungan Anda, sehingga latihan terjadi secara otomatis. Kelemahan dari imersi adalah biasanya membutuhkan menjadikan keterampilan sebagai fokus utama Anda untuk jangka waktu yang lama. Jika meninggalkan semua komitmen Anda, mengemas tas Anda, dan pindah ke Prancis adalah pilihan yang bisa diterapkan, belajar bahasa Prancis melalui pencelupan adalah strategi yang baik. Sayangnya, sebagian besar dari kita memiliki komitmen yang tidak dapat (atau tidak ingin) kita tinggalkan: keluarga, pekerjaan, pembayaran hipotek, dan sebagainya. Dalam kasus ini, perendaman bisa menjadi sulit atau mustahil. Dalam skenario terburuk, gagasan imersi menjadi penghalang aktif: jika Anda terus menunggu kesempatan imersi sebelum berkomitmen untuk memperoleh keterampilan baru, Anda dapat membuang waktu berharga selama bertahun-tahun. Ambil kesempatan pencelupan saat datang, tetapi jangan mengandalkannya. Teknik-teknik ini dirancang untuk membantu Anda memperoleh keterampilan baru bahkan jika Anda hanya memiliki satu atau dua jam luang setiap hari.

Mengaktifkan Kembali Keterampilan Lama

Penting juga untuk dicatat bahwa prinsip-prinsip ini berguna bahkan jika keterampilan yang Anda coba peroleh tidak sepenuhnya baru bagi Anda. Sangat mungkin untuk menggunakan teknik ini untuk memperoleh kembali keterampilan lama dalam waktu singkat. Misalnya, saya belajar bermain terompet di sekolah menengah, dan saya cukup berlatih sehingga saya cukup pandai dalam hal itu. Sejak lulus dan kuliah, saya belum bermain sama sekali. Jika saya memutuskan untuk mengambil terompet lagi, tidak perlu banyak latihan untuk mengaktifkan kembali keterampilan tersebut. Saya sudah tahu subketerampilan yang diperlukan, jadi saya akan fokus pada embouchure (mengontrol otot-otot di sekitar bibir sambil meniup ke corong), membaca catatan dan mengingat posisi nger terkait, dan meninjau teori musik dasar (ketukan, tempo, dinamika, dan ekspresi). Hanya perlu beberapa jam latihan untuk memperoleh kembali keterampilan. Reaktivasi sebagian besar membutuhkan waktu luangkan waktu, menghilangkan hambatan untuk berlatih, dan berlatih dengan waktu.

Dimulai dengan baik adalah setengah selesai

Terkadang Anda ingin melepaskan gitar. Anda akan membenci gitar. Tetapi jika Anda tetap melakukannya, Anda akan dihargai.—JIMI HENDRIX, PEMAIN GITAR LISTRIK TERKENAL

Anda tidak perlu menggunakan semua prinsip ini untuk setiap keterampilan yang Anda peroleh, tetapi Anda akan selalu menemukan setidaknya beberapa dari mereka yang penting. Saya dan berguna untuk menganggap prinsip-prinsip ini sebagai daftar periksa. Setiap kali Anda memutuskan untuk mempelajari sesuatu yang baru, buka saja daftar periksa dan putuskan prinsip mana yang berlaku untuk proyek khusus Anda. Berikut daftar periksa untuk akuisisi keterampilan yang cepat:

1. Pilih proyek yang menyenangkan.
2. Fokuskan energi Anda pada satu keterampilan pada satu waktu.

3. Berikan Target Tingkat Kinerja Anda.4.
Keterampilan Dekonstruksi menjadi
Subketerampilan.5. Alat Kritis Obtane.6.
Menghilangkan Hambatan untuk Berlatih.7.
Luangkan waktu khusus untuk latihan.8.
Buat Loop Umpan Balik Cepat.9.
Berlatihlah dengan jam dalam semburan
singkat.10. Kuantitas dan kecepatan amfibi.

Itu saja. Terapkan daftar periksa ini ke keterampilan utama Anda saat ini, dan latihan Anda akan lebih efektif dan bermanfaat, memungkinkan Anda untuk memperoleh keterampilan lebih cepat. Seperti yang saya katakan, metode ini bukanlah ilmu roket. Ini adalah akal sehat, strategi, dan persiapan yang diterapkan pada keterampilan yang ingin Anda tingkatkan. Tidak lebih, tidak kurang. Sekarang, mari kita periksa bagaimana pembelajaran dan penelitian dapat membuat proses perolehan keterampilan Anda menjadi lebih efektif.



3

Sepuluh Prinsip Pembelajaran Eektif

Tidak ada masalah yang dapat menahan serangan
pemikiran yang berkelanjutan.—VOLTAIRE

■ ■ ■

Seperti yang kita bahas di bab 1, belajar tidak sama dengan perolehan keterampilan. Namun, itu tidak berarti belajar tidak penting. Melakukan sedikit penelitian sebelum Anda terjun ke latihan dapat menghemat waktu, energi, dan ketabahan emosional yang berharga. Belajar membuat latihan Anda lebih tua, yang memungkinkan Anda menghabiskan lebih banyak waktu latihan Anda untuk mengerjakan subketerampilan yang paling penting. Dengan semangat itu, berikut adalah sepuluh prinsip utama pembelajaran eektif:

1. Teliti keterampilan dan topik terkait.
2. Lompat di atas kepala Anda.
3. Identifikasi model mental dan kait mental.
4. Bayangkan kebalikan dari apa yang Anda inginkan.
5. Bicaralah dengan praktisi untuk menetapkan harapan.
6. Hilangkan gangguan di lingkungan Anda.
7. Gunakan pengulangan dan penguatan spasi untuk menghafal.
8. Buat scaold dan daftar periksa.
9. Membuat dan menguji prediksi.
10. Hormati biologi Anda.

1. Teliti keterampilan dan topik terkait.

Luangkan waktu dua puluh menit untuk mencari web, menjelajahi toko buku, atau memindai tumpukan di perpustakaan lokal Anda untuk buku dan sumber daya yang terkait dengan keterampilan. Tujuannya adalah untuk mengidentifikasi setidaknya tiga buku, DVD instruksional, kursus, atau sumber daya lain yang tampaknya berhubungan dengan keterampilan yang ingin Anda peroleh. Sebelum Anda panik, pahami bahwa Anda tidak perlu menghabiskan waktu berjam-jam untuk menghafal sumber daya ini. Sebaliknya: waktu yang dihabiskan untuk membaca atau menonton bukanlah waktu yang dihabiskan untuk berlatih. Anda tidak menjejalkan ujian. Tujuan dari penelitian awal ini adalah untuk mengidentifikasi subketerampilan yang paling penting, komponen kritis, dan alat yang diperlukan untuk latihan secepat mungkin. Semakin banyak Anda tahu sebelumnya tentang keterampilan, semakin cerdas Anda dapat mempersiapkannya. Tujuannya adalah untuk mengumpulkan pengetahuan yang luas tentang keterampilan secepat mungkin, menciptakan gambaran yang akurat tentang seperti apa proses perolehan keterampilan nantinya. Untuk perolehan keterampilan yang cepat, skimming lebih baik daripada membaca secara mendalam. Dengan memperhatikan ide dan alat yang muncul berulang kali teks indierent, Anda dapat mempercayai keakuratan pola yang Anda perhatikan dan mempersiapkan latihan Anda sesuai dengan itu. Jika Anda ingin bisa memanggang croissant yang sempurna, ambil beberapabuku bagus yang berkaitan dengan kue dan kue kering. Alih-alih menemukan kembali prosesnya, Anda akan menemukan teknik yang ada yang telah disempurnakan selama bertahun-tahun oleh para master senior. Jika Anda melihat teknik atau proses yang sama yang dijelaskan dalam beberapa sumber daya, kemungkinan besar penting untuk diketahui. Setelah Anda menemukan teknik yang tampaknya paling berguna, Anda dapat bereksperimen dengannya di dapur Anda sendiri, menghemat Anda dari coba-coba.

2. Lompat di atas kepala Anda.

Beberapa penelitian awal Anda akan berisi konsep, teknik, dan ide yang tidak Anda pahami. Seringkali, sesuatu akan tampak sangat penting, tetapi Anda tidak akan tahu apa artinya. Anda akan membaca kata-kata yang tidak Anda kenali, dan melihat praktisi melakukan hal-hal yang tidak dapat Anda pahami. Jangan panik. Kebingungan awal Anda benar-benar normal. Faktanya, itu bagus. Bergerak menuju kebingungan. Penelitian awal adalah salah satu cara terbaik untuk mengidentifikasi subketerampilan dan ide penting, tetapi kemungkinan besar Anda juga belum tahu apa artinya. Maknanya datang kemudian, setelah Anda mulai berlatih. Dr. Stephen Krashen, ahli akuisisi bahasa yang saya sebutkan sebelumnya, menyebut masukan yang dapat dipahami ini. Secara default, informasi baru yang Anda konsumsi tidak terlalu dapat dipahami, karena tidak terhubung dengan apa pun yang Anda ketahui atau alami. Seiring waktu, informasi yang sama akan menjadi mudah dipahami setelah Anda memiliki beberapa pengalaman di bawah ikat pinggang Anda. Dalam kata-kata guru yoga terkenal T. K. V. Desikachar: "Pengakuan kebingungan itu sendiri adalah bentuk kejelasan." Menyadari Anda bingung itu berharga. Mengenali kebingungan dapat membantu Anda menyangkal dengan tepat apa yang Anda bingungkan, yang membantu Anda mengetahui apa yang perlu Anda teliti atau lakukan selanjutnya untuk menyelesaikan kebingungan itu. Jika Anda tidak bingung dengan setidaknya setengah dari penelitian awal Anda, Anda tidak belajar secepat yang Anda mampu belajar. Jika Anda mulai merasa terintimidasi atau ragu-ragu tentang kecepatan yang Anda coba ulang, Anda berada di jalur yang benar. Asalkan Anda mengerjakan masalah atau proyek yang menyenangkan, semakin bingung Anda di awal, semakin banyak tekanan internal yang akan Anda rasakan untuk mengungkapkan sesuatu, dan semakin cepat Anda akan belajar. Tidak mau melompat di atas kepala Anda adalah satu-satunya penghalang emosional terbesar untuk memperoleh keterampilan yang cepat. Merasa bodoh tidaklah menyenangkan, tetapi mengingatkan diri sendiri bahwa Anda akan mengerti dengan latihan akan membantu Anda beralih dari kebingungan ke kejelasan secepat mungkin.

3. Identifikasi model mental dan kait mental. Saat Anda melakukan penelitian, Anda secara alami akan mulai memperhatikan pola: ide dan teknik yang muncul berulang kali. Konsep-konsep ini disebut model mental, dan mereka sangat penting. Model mental adalah unit pembelajaran yang paling dasar: cara memahami dan melabeli objek atau hubungan yang ada di dunia. Saat Anda mengumpulkan model mental yang akurat, menjadi lebih mudah untuk mengantisipasi apa yang akan terjadi ketika Anda mengambil tindakan khusus. Model mental juga membuatnya lebih mudah untuk mendiskusikan pengalaman Anda dengan orang lain. Berikut contohnya: Saya baru-baru ini membantu ayah saya menyiapkan situs web. Saat saya berjalan, saya mencoba menjelaskan apa yang saya lakukan. Atrst, itu membuat frustrasi bagi kami berdua: Saya terus menggunakan kata-kata seperti "server", dan dia sama sekali tidak tahu apa yang saya bicarakan. Begitu Ayah mengetahui bahwa server adalah komputer khusus yang mengirimkan halaman web kepada orang-orang yang memintanya, dan bahwa server adalah komputer yang lebih baik daripada mesin yang kami gunakan, dia merasa jauh lebih mudah untuk memahami apa yang kami lakukan. Dalam hal ini, server adalah model mental—setelah Anda terbiasa dengan istilah tersebut, lebih mudah untuk memahami proses penerbitan situs web. Anda juga akan melihat beberapa hal yang terlihat seperti sesuatu yang sudah Anda kenal. Ini adalah kait mental: analogi dan metafora yang dapat Anda gunakan untuk mengingat konsep baru. Dalam kasus server web, bayangkan seorang pustakawan. Ketika Anda pergi ke perpustakaan dan meminta buku khusus, pustakawan akan mencari rak yang berisi ratusan atau ribuan buku untuk menemukan buku yang tepat yang Anda cari. Ketika pustakawan menandai buku itu, dia membawanya kembali untuk Anda. Jika buku itu tidak dapat ditemukan, pustakawan akan memberi tahu Anda "Saya tidak dapat menemukan buku yang Anda cari." Begitulah cara kerja server web. Saat Anda meminta halaman specicweb, server akan mencari halaman itu di memori. Jika itu menderita halaman, itu akan mengirimkannya kepada Anda. Jika server tidak dapat mengoperasikan web

, itu akan mengembalikan pesan: "Kesalahan 404: Halaman Tidak Ditemukan." Menganggap perangkat lunak server sebagai "pustakawan komputer" sangat membantu ketika memikirkan cara kerja sistem. Semakin banyak model mental dan kait mental yang dapat Anda identifikasi dalam penelitian awal Anda, semakin mudah untuk menggunakannya saat Anda berlatih kembali.

4. Bayangkan kebalikan dari apa yang Anda inginkan. Cara yang berlawanan dengan intuisi untuk mendapatkan wawasan tentang keterampilan baru adalah merenungkan bencana, bukan kesempurnaan. Bagaimana jika Anda melakukan semuanya salah? Bagaimana jika Anda mendapatkan hasil terburuk? Ini adalah teknik pemecahan masalah yang disebut inversi, dan sangat membantu dalam mempelajari hal-hal penting dari hampir semua hal. Dengan mempelajari kebalikan dari apa yang Anda inginkan, Anda dapat mengidentifikasi elemen-elemen penting yang tidak langsung jelas. Ikuti kayak air putih. Apa yang perlu saya ketahui jika saya ingin bisa berkayak di sungai besar, bergerak cepat, bertaburan batu? Inilah pembalikannya: Seperti apa bentuknya jika semuanya berjalan salah?

Saya akan terbalik di bawah air, dan tidak bisa bangkit kembali. Saya akan mengorek kayak saya, menyebabkannya tenggelam atau rawa, mengakibatkan hilangnya kayak total. Saya akan kehilangan dayung saya, menghilangkan kemampuan manuver saya. Saya membenturkan kepala saya ke batu. Saya akan keluar dari kayak saya, terjebak di hidrolis (titik di sungai di mana sungai mengalir kembali ke dirinya sendiri, menciptakan lingkaran seperti mesin cuci) dan tidak bisa keluar.

Jika saya berhasil melakukan semua hal ini sekaligus di tengah sungai yang deras, saya mungkin akan mati—skenario terburuk. Garis pemikiran yang menyedihkan ini berguna karena menunjuk pada beberapa keterampilan kayak air putih yang mungkin sangat penting:

- Belajar menggulung kayak dengan sisi kanan ke atas jika terputar, tanpa mengeluarkannya. Mempelajari cara mencegah kayak membanjiri jika dikeluarkan diperlukan. Belajar bagaimana menghindari kehilangan dayung saya di air yang kasar. Mempelajari dan menggunakan tindakan pencegahan keselamatan saat arung jeram di sekitar bebatuan besar. Mengintai sungai sebelum berlari untuk menghindari fitur sungai yang berbahaya sepenuhnya.

Simulasi mental ini juga memberi saya daftar belanja: Saya perlu berinvestasi dalam rompi otation, helm, dan perlengkapan keselamatan lainnya. Sekarang, alih-alih (1) sungai rakit (2) bersenang-senang (3) jangan mati, saya memiliki daftar konkret subskill untuk dilatih dan tindakan yang harus diambil untuk memastikan saya benar-benar bersenang-senang, menyimpan perlengkapan saya, dan bertahan dalam perjalanan. Inversi berfungsi.

5. Bicaralah dengan praktisi untuk menetapkan harapan. Pembelajaran awal membantu Anda menetapkan harapan yang sesuai: Seperti apa sebenarnya kinerja yang wajar untuk seorang pemula? Ketika Anda melompat ke memperoleh keterampilan baru, sangat umum untuk meremehkan kompleksitas tugas, atau jumlah elemen yang terlibat yang diperlukan untuk bekerja dengan baik. Jika keterampilan melibatkan kemungkinan prestise sosial, mistik terkait juga dapat mengaburkan harapan awal. Banyak calon bintang rock telah mengambil gitar listrik, hanya saja yang sangat sulit untuk bermain dengan baik, bernyanyi di kunci, dan terlihat luar biasa pada saat yang bersamaan. Bagian dari masalahnya adalah bahwa "menjadi rockstar" bukanlah satu keterampilan. Ini adalah bundel dari banyak subskill terkait, yang masing-masing membutuhkan latihan khusus untuk dikembangkan. Berbicara dengan orang-orang yang telah memperoleh keterampilan sebelum Anda akan membantu menghilangkan mitos dan kesalahpahaman sebelum Anda menginvestasikan waktu dan energi Anda. Dengan mengetahui apa yang dapat Anda harapkan untuk dilihat saat Anda maju, Anda akan lebih mudah untuk mempertahankan minat Anda dalam latihan, dan menghindari keputusasaan di awal proses.

6. Hilangkan gangguan di lingkungan Anda.

Gangguan adalah musuh nomor satu dalam perolehan keterampilan yang cepat. Gangguan membunuh latihan terfokus, dan kurangnya latihan terfokus menyebabkan perolehan keterampilan yang lambat (atau tidak ada). Anda dapat menghindarinya dengan meluangkan beberapa menit untuk mengantisipasi dan menghilangkan (atau mengurangi) gangguan sebanyak mungkin sebelum Anda mulai berlatih. Sumber gangguan yang paling signifikan datang dalam dua bentuk: elektronik dan biologis. Televisi, ponsel, dan Internet Anda adalah gangguan elektronik. Putar o, cabut, blokir, atau hapus dari lingkungan Anda saat Anda berlatih kecuali jika benar-benar diperlukan untuk latihan itu sendiri. Anggota keluarga, kolega, dan hewan peliharaan yang bermaksud baik adalah gangguan biologis. Anda tidak dapat mengubah orang o, tetapi Anda dapat memberi tahu mereka sebelumnya bahwa Anda tidak akan tersedia saat Anda berlatih, yang membuat kemungkinan besar mereka akan menghormati waktu latihan Anda tanpa menyela. Semakin sedikit gangguan yang Anda miliki saat berlatih, semakin cepat Anda memperoleh keterampilan.

7. Gunakan pengulangan dan penguatan spasi untuk menghafal. Untuk memanfaatkan materi yang telah Anda pelajari saat berlatih, Anda harus dapat mengingat ide-ide terkait dengan cepat. Banyak keterampilan membutuhkan setidaknya beberapa tingkat hafalan. Inilah tangkapannya: ingatan Anda tidak sempurna. Setiap kali Anda mempelajari sesuatu yang baru, Anda mungkin akan melupakannya kecuali Anda meninjau konsep dalam jangka waktu tertentu. Pengulangan ini memperkuat gagasan, dan membantu otak Anda mengkonsolidasikannya ke dalam memori jangka panjang. Para peneliti telah menemukan bahwa memori mengikuti kurva peluruhan: konsep baru perlu diperkuat secara teratur, tetapi semakin lama Anda mengetahui suatu konsep, semakin sedikit Anda perlu meninjaunya secara teratur untuk mempertahankan ingatan yang akurat.

Penulangan dan penguatan jarak adalah teknik menghafal yang membantu Anda secara sistematis meninjau konsep dan informasi penting secara teratur. Ide-ide yang sulit diingat sering ditinjau, sementara konsep yang lebih mudah dan lama lebih jarang ditinjau. Program perangkat lunak kartu flash seperti Anki,¹ SuperMemo,² dan Smart3 membuat pengulangan dan penguatan spasi menjadi sangat sederhana. Sistem pengulangan jarak mengandalkan model tinjauan "kartu abu", dan Anda harus membuat kartu abu sendiri. Dengan membuat kartu abu saat Anda mendekonstruksi skill, Anda membunuh dua burung dengan satu batu. Setelah Anda membuat kartu abu Anda, hanya perlu beberapa menit setiap hari untuk meninjaunya. Dengan mensistematisasi proses peninjauan dan melacak penarikan, sistem ini dapat membantu Anda mempelajari ide, teknik, dan proses baru dalam waktu singkat. Jika Anda meninjau dek secara konsisten, Anda akan menghafal konsep dan ide yang diperlukannya sangat cepat. Penting untuk dicatat bahwa perolehan keterampilan biasanya jauh lebih terlibat daripada pembelajaran akademis. Jika Anda terutama tertarik untuk menghafal konsep, ide, atau kosakata untuk lulus ujian, Anda tidak memerlukan lebih dari pengulangan spasi.⁴ Penggunaan terbaik dari teknik ini adalah dalam kasus di mana mengingat informasi dengan cepat sangat penting. Jika Anda mempelajari kosakata umum untuk memperoleh bahasa baru, pengulangan dan penguatan berjarak sangat berharga. Dalam kasus di mana penarikan cepat tidak penting, Anda biasanya lebih baik melewati kartu abu demi memaksimalkan waktu latihan dan eksperimen.

8. Buat scaold dan daftar periksa. Banyak keterampilan melibatkan semacam rutinitas: menyiapkan, mempersiapkan, memelihara, menyimpan, dan sebagainya. Membuat sistem sederhana adalah

Cara terbaik untuk memastikan elemen-elemen penting ini terjadi dengan sedikit semangat tambahan. Daftar periksa berguna untuk mengingat hal-hal yang harus dilakukan setiap kali Anda berlatih. Mereka adalah cara untuk mensistematisasi proses, yang membebaskan perhatian Anda untuk fokus pada hal-hal yang lebih penting. Scaold adalah struktur yang memastikan Anda mendekati keterampilan dengan cara yang sama setiap saat. Pikirkan pemain bola basket yang menetapkan rutinitas pra-lemparan bebas. Lap tangan di celana, kendurkan bahu, tangkap bola dari wasit, memantul tiga kali, berhenti selama tiga detik, dan menembak. Itu scaold. Membuat scaold dan daftar periksa membuat praktik Anda lebih tua. Mereka juga membuat latihan Anda lebih mudah divisualisasikan, yang membantu Anda memanfaatkan latihan mental, yang dapat membantu dengan beberapa bentuk latihan fisik.

9. Buat dan uji prediksi. Bagian dari proses perolehan keterampilan melibatkan eksperimen: mencoba hal-hal baru untuk melihat apakah itu berhasil. Tes sebenarnya dari pembelajaran yang berguna adalah prediksi. Berdasarkan apa yang Anda ketahui, dapatkan Anda menebak bagaimana perubahan atau eksperimen akan berubah sebelum Anda melakukannya? Membiasakan diri membuat dan menguji prediksi akan membantu Anda memperoleh keterampilan dengan lebih cepat. Ini adalah variasi dari metode ilmiah, dengan empat elemen kunci:

Pengamatan—apa yang sedang Anda amati? Yang diketahui—apa yang sudah Anda ketahui tentang topik ini? Hipotesis—menurut Anda apa yang akan meningkatkan kinerja Anda? Tes—apa yang akan Anda coba selanjutnya?

Saya sarankan menggunakan buku catatan atau alat referensi lainnya untuk melacak eksperimen Anda dan membentuk hipotesis saat Anda berlatih. Dengan melacak prediksi Anda dan menghasilkan ide-ide baru, Anda akan memiliki eksperimen yang lebih bermanfaat untuk diuji.

10. Hormati biologi Anda. Otak dan tubuh Anda adalah sistem biologis yang memiliki kebutuhan biologis: makanan, air, olahraga, istirahat, dan tidur. Sangat mudah untuk memaksakan diri terlalu keras, yang kontraproduktif. Tanpa input yang tepat, tubuh dan pikiran Anda tidak akan menghasilkan output yang berguna. Menurut Tony Schwartz, penulis *The Power of FullEngagement* (2004) dan *Be Excellent at Anything* (2011), siklus pembelajaran yang optimal tampaknya sekitar sembilan puluh menit konsentrasi terfokus. Lagi, dan pikiran dan tubuh Anda secara alami akan membutuhkan istirahat. Gunakan kesempatan itu untuk berolahraga, beristirahat, makan atau camilan, tidur siang, atau melakukan hal lain. Prinsip ini sangat cocok dengan latihan dengan jam. Dengan mengatur timer Anda selama enam puluh hingga sembilan puluh menit sebelum Anda mulai berlatih atau meneliti, akan lebih mudah untuk mengingat untuk beristirahat setelah selesai. Anda juga dapat membagi latihan Anda menjadi beberapa bagian yang lebih kecil, dengan istirahat singkat di tengah jika diperlukan: dua puluh menit latihan, istirahat sepuluh menit, dua puluh menit latihan, istirahat sepuluh menit, dan sebagainya.

Menumpuk Dek

Tidak ada aturan di sekitar sini. Kami mencoba mencapai sesuatu.—
THOMAS EDISON, PENEMU

Anda tidak perlu menggunakan semua prinsip ini untuk setiap keterampilan yang Anda peroleh, tetapi Anda akan selalu menemukan setidaknya beberapa di antaranya penting. Saya dan berguna untuk menganggap prinsip-prinsip ini sebagai daftar periksa sekunder. Setiap kali Anda memutuskan untuk memperoleh keterampilan baru, cukup tinjau daftar periksa ini dan putuskan prinsip mana yang berlaku untuk proyek Anda. Berikut checklist untuk pembelajaran eective:

Teliti keterampilan dan topik terkait.
Lompat di atas kepala Anda. Identifikasi model mental dan kait mental.

- Bayangkan kebalikan dari apa yang Anda inginkan. Bicaralah dengan praktisi untuk menetapkan harapan. Hilangkan gangguan di lingkungan Anda. Gunakan pengulangan dan penguatan spasi untuk menghafal. Buat scaold dan daftar periksa. Buat dan uji prediksi. Hormati biologi Anda.

Itu saja: terapkan daftar periksa ini ke keterampilan utama Anda saat ini dan Anda akan mempelajari apa yang perlu Anda ketahui untuk berlatih secara ilmiah secara andeektif.

Mempraktikkan Teori

Betapa-duduk untuk menulis ketika Anda belum berdiri untuk hidup.—
HENRY DAVID THOREAU

Teori yang cukup: saatnya untuk berlatih. Kami telah membahas dasar-dasar perolehan keterampilan yang cepat, tetapi mengetahui cara melakukan hal-hal ini tidak sepenting melakukannya. Ingat: tidak ada latihan, tidak ada perolehan keterampilan. Alih-alih terus menerus tentang teori perolehan keterampilan, saya akan menunjukkan kepada Anda bagaimana benar-benar melakukannya. Saya akan menggunakan prinsip-prinsip ini untuk memperoleh beberapa keterampilan baru, dan Anda akan memiliki kursi barisan depan. Berikut adalah keterampilan yang ingin saya peroleh:

Yoga: mengembangkan latihan asana rumahan. Pemrograman: membuat aplikasi web yang berfungsi. Mengetik: belajar kembali mengetik sentuh dengan tata letak keyboard nonstandar. Strategi: bermain Go, permainan papan tertua dan terkompleks di dunia. Musik: bermain ukulele. Windsurung: berlayar dan bermanuver di air.

Saya tidak memiliki pengalaman dengan salah satu dari keterampilan ini. Dengan menggunakan teknik dan metode yang baru saja saya jelaskan, tujuan saya adalah untuk mendapatkannya masing-masing dalam tiga puluh hari atau kurang. Perkiraan waktu saya untuk memperoleh masing-masing keterampilan ini adalah sekitar dua puluh jam, rata-rata enam puluh hingga sembilan puluh menit latihan setiap hari.

Tentang Contoh Ini

Keterampilan khusus ini benar-benar unik. Itu adalah hal-hal yang saya minati untuk dipelajari karena berbagai alasan, yang akan saya jelaskan nanti secara rinci. Keterampilan yang ingin Anda peroleh mungkin cukup berbeda, tetapi proses perolehan keterampilan inti sebagian besar akan sama. Harapan saya adalah bahwa dengan menunjukkan kepada Anda bagaimana saya telah menggunakan proses ini untuk memperoleh banyak keterampilan yang berbeda di banyak bidang yang berbeda, Anda akan mendapatkan pemahaman yang lebih lengkap tentang cara menggunakan teknik ini untuk menjadi lebih baik dalam keterampilan yang penting bagi Anda. Jika Anda secara alami lebih tertarik pada satu atau dua contoh ini daripada yang lain, tidak apa-apa. Baca bab-bab itu rst. Jika Anda mulai membaca sebuah bab dan keterampilan membosankan atau tidak berlaku untuk situasi Anda, jangan ragu untuk melewatinya. Saya menggunakan metode samecore untuk semua contoh ini, jadi Anda tidak akan melewatkan bagian penting dari pendekatan ini. Saya menulis bab-bab ini dalam format tutorial, dengan asumsi Anda belum pernah melihat keterampilan ini sebelumnya, jadi Anda tidak harus memiliki pengalaman atau pengetahuan sebelumnya untuk mengikutinya. Selain mengilustrasikan metodenya, saya harap Anda juga dapat mengambil beberapa pengetahuan berharga tentang enam keterampilan menarik yang layak untuk dilatih. Jika Anda memiliki banyak pengalaman dalam salah satu mata pelajaran ini, kemungkinan Anda akan melihat kesalahan dalam pemahaman saya, kesalahan dalam myverbiage, atau ketidaksepakatan dengan metode saya. Itu benar-benar ne. Ingat, saya memulai sebagai pemula yang lengkap, dan saya tidak tahu apa yang saya lakukan. (Belum.) Apa yang akan Anda baca adalah ikhtisar proses pembelajaran saya untuk masing-masing keterampilan ini. Saya telah melakukan segala upaya untuk memastikan informasi dalam buku ini akurat dan lengkap, tetapi saya pasti akan membuat kesalahan. Dalam semua kasus, metode penelitian dan praktik yang disengaja adalah yang paling penting. Pertama: yoga.



4Yog

a

Pelajaran: Jangan Membuatnya Lebih Sulit Dari Yang Seharusnya

Saya tidak mengukur kemajuan saya dalam Yoga dengan seberapa jauh saya dapat membungkuk atau memutar, tetapi dengan bagaimana saya memperlakukan istri dan anak-anak saya.—T. K. V. DESIKACHAR, GURU YOGA TERKENAL

• • •

Untuk gambar tambahan, video, dan komentar tentang bab ini,
visithttp://first20hours.com/yoga.

Saya semakin tua. Untuk memperjelas: secara relatif, saya masih muda, tetapi saya mulai memperhatikan beberapa hal tentang tubuh saya yang menjadi perhatian saya. Ketika saya bangun di pagi hari, saya sering mengalami sakit tumpul di tengah punggung saya. Setelah seharian di depan komputer, leher dan bahu saya terasa sesak dan sakit. Ini bukan sensasi yang menyenangkan. Sampai saat ini, saya belum terlalu memikirkan tubuh saya. Itu hanya berhasil. Setelah meninggalkan sekolah menengah, dan dengan itu, olahraga terorganisir, saya belum berolahraga sama sekali. Begitu saya mulai kuliah, saya mulai memperlakukan tubuh saya sebagai kendaraan yang satu-satunya tujuan adalah untuk mengangkut otak saya dari satu kelas ke kelas berikutnya. Sejak lulus itu tidak berubah, dan sampai saat ini saya merasa baik-baik saja. Sekarang, tahun-tahun pengabaian fisik itu menyusul saya. Saya renyah, dan saya tahu itu.

"Anda Harus Benar-Benar Melihat Yoga ..."

Kelsey, istri saya, telah melakukan yoga sejak kuliah. Setiap hari atau lebih, dia akan berjalan menyusuri jalan ke studio lokal dan menghabiskan satu jam atau lebih untuk memutar dirinya ke berbagai posisi di ruangan yang sangat hangat. Dia menyukainya, dan selalu pulang dengan bahagia dan santai. Setiap minggu, Kelsey akan mengatakan kepada saya "Anda harus benar-benar mencoba yoga. Itu bagus. Anda akan sangat menyukainya." Saya ragu-ragu. Yoga, bagi saya, tidak terlalu menarik. Itu bukan peregangan. Saya berlari di lintasan di sekolah menengah, jadi peregangan bukanlah masalah besar. Acara favorit saya adalah rintangan 110 meter, yang membutuhkan tingkat kebolehan yang signifikan. 1Pelatihan untuk rintangan melibatkan banyak peregangan, dan saya telah mempertahankan banyak kemampuan yang saya kembangkan selama tahun-tahun itu, terutama di paha belakang saya. Bahkan sekarang, lebih dari satu dekade kemudian, saya dapat meletakkan telapak tangan saya di atas oor tanpa menekuk lutut saya. Keraguan saya dalam berlatih yoga adalah, terus terang, betapa anehnya kelihatannya.

Chakra, Aura, dan Kundalinis, Ya ampun!

Sh!t Yogis Say,2 sebuah video yang diproduksi oleh lululemon,3 salah satu pengecer pakaian atletik paling propertabel di dunia, adalah contoh lidah dan berlebihan tentang bagaimana yoga dapat terdengar bagi orang yang belum tahu. Berikut adalah beberapa sorotan:

"Aku khawatir tentang auramu ...""Bagaimana kamu mengatakan itu dalam bahasa Sanssekerta?"" Chakra saya sangat sejajar." Ingin melihat di mana saya bisa meletakkan kaki saya?"

Video tersebut telah ditonton lebih dari 2 juta kali sejak dirilis di YouTube pada Desember 2011, jadi jelas menyentuh akord. Saya orang yang cukup rasional dan bumi. Saya siap untuk percaya bahwa peregangan secara teratur baik untuk Anda. Mempelajari

urutan peregangan tampaknya sangat masuk akal. Semua pembicaraan tentang aura, chakra, dan devosional spiritual esoterik ini, bagaimanapun, adalah perubahan besar. Saya tidak tahu apa hubungannya dengan olahraga, dan terus terang, mistisisme hippie membuat yoga terdengar seperti kultus yang aneh. Bukan itu saja. Praktik yoga di Amerika telah berkembang menjadi gaya hidup di pusat industri dan pakaian senilai 8 miliar dolar. Menunjukkan betapa yoganya Anda telah menjadi sinyal status sosial yang trendi, setara dengan membawa tas tangan terbaru dari Louis Vuitton. No terima kasih. "Chakra" saya adalah ne, terima kasih banyak.

"Rilekskan Wajahmu"

Akhirnya, Kelsey meyakinkan saya untuk mencoba yoga sebagai resolusi Tahun Baru. Saya bekerja dari rumah, jadi dia melemparkannya sebagai cara yang baik untuk keluar dari apartemen dan bertemu orang-orang. Saya masih skeptis, tetapi saya memutuskan untuk mengolok-oloknya. Kami mendapatkan keanggotaan di Pure Yoga, sebuah studio kelas atas baru di Upper East Side New York. Banyak kelas diajarkan oleh Marco Rojas, guru favorit Kelsey dan salah satu instruktur berperingkat teratas di kota. Bayangkan versi Antonio Banderas yang lebih muda dan lebih ramping dengan celana yoga, dan Anda memiliki Marco. Dia tampan, karismatik, dan memiliki suara pembunuh dengan aksen Spanyol. Kelas Marco penuh sesak, baik karena dia adalah guru yang benar-benar hebat, dan karena 99 persen yogini di kota sangat naksir dia. (Dan siapa yang bisa menyalahkan mereka?) Karena saya baru mengenal yoga, saya mencoba yang terbaik untuk mengikuti seperti yang diajarkan Marco. Saya tidak tahu posenya, atau apa nama setiap pose, jadi saya hanya memperhatikan apa yang dilakukan orang lain dan mencoba menirunya. Butuh banyak konsentrasi. Saya bisa mengikuti, tetapi itu adalah perjuangan. Beberapa pose, seperti headstand, jauh di luar kemampuan saya.

Saya mencoba untuk tetap berada di belakang ruangan, sehingga saya bisa melihat apa yang dilakukan orang lain, dan juga mengurangi kerusakan tambahan jika Ifell, yang terjadi lebih sering daripada yang ingin saya akui. Saya menikmati kelas Marco, tetapi saya tidak benar-benar mengerti apa yang terjadi. Sesekali, Marco akan mampir ke matras saya dan menyesuaikan, membantu saya melakukan pose dengan benar. Pada satu kesempatan yang tak terlupakan, Marco menguatkan kakiku dengan kakinya sendiri, sehingga mereka tegak lurus dengan oor. Kemudian, dia meraih tanganku dan menariknya. Sulit." Libatkan paha depan Anda ... berputar di pinggul. Bagus. Sekarang... rileks wajahmu." Saya meringis, dan untuk alasan yang bagus: pergelangan tangan saya menyentuh jari-jari kaki saya. Saya mengendurkan wajah saya.

Dilema "Tuan Rumah"

Sebanyak saya menikmati kursus Marco, sulit untuk meluangkan waktu untuk mereka. Setiap kelas berlangsung satu setengah jam, tidak termasuk waktu yang dibutuhkan untuk berjalan ke studio, berganti pakaian, mandi setelahnya, dan berjalan pulang. Meskipun studio hanya beberapa blok jauhnya, itu masih merupakan komitmen dua jam yang solid. Saya bekerja dari rumah dengan pekerjaan yang menuntut untuk perusahaan Fortune 500 pada saat itu, dan saya sering mengadakan pertemuan yang tumpang tindih dengan waktu kelas yang dijadwalkan. Tidak membantu bahwa kelas di pagi dan sore hari, yang berarti saya harus melewatkan pekerjaan untuk hadir. Secara umum, pengaturan kerja saya dapat dihindari, jadi sebagian besar itu adalah masalah psikologis: sulit untuk meyakinkan diri sendiri bahwa dapat diterima untuk memutar tubuh saya menjadi simpul selama dua jam ketika ada pekerjaan yang harus dilakukan. Pada akhirnya, saya menghadiri mungkin tiga belas kelas tahun itu. Meskipun saya merasa hebat setelah setiap kelas, saya tidak bisa membenarkan komitmen waktu.

Selain itu, saya tidak berlatih di rumah. Saya tidak tahu bagaimana. Selama kelas, saya meniru siswa lain. Saya bisa menggulung tikar di atas oor di rumah dan melakukan peregangan, tetapi dalam pikiran saya, itu tidak benar-benar yoga.

Saat siswa siap, guru muncul

Setelah kami pindah ke Colorado, Kelsey menyebutkan kepada saya suatu malam bahwa dia merindukan New York. Dia merindukan pergi ke kelas Marco, dan dia merindukan kursus anatomi yoga Leslie Kamino, yang dia ambil setelah menyelesaikan sertifikasi guru yoganya. Leslie adalah rekan penulis Yoga Anatomy (edisi pertama, 2007; edisi kedua, 2011), salah satu buku yoga terlaris sepanjang masa, dan materinya digunakan dalam program pelatihan guru yoga di seluruh dunia. Sayangnya, untuk mengikuti kursus Leslie, Anda harus tinggal di New York City, jadi Kelsey harus berhenti ketika kami pindah." Seseorang harus benar-benar menempatkan kursus Leslie secara online," katanya. Inilah ironinya. Beberapa minggu sebelumnya, Kelsey telah membantu saya meluncurkan kursus online pertama saya, jadi dia tahu bagaimana melakukannya. Dia juga benar tentang kursus Leslie: karena hanya tersedia untuk siswa yang tinggal di New York City, tetapi siswa dari seluruh dunia ingin berpartisipasi, akan sangat masuk akal untuk menerbitkan kursus secara online. Saya menunjukkan yang jelas: Kelsey sepenuhnya mampu menciptakan kursus jika Leslie bersedia membiarkannya memproduksinya. Malam itu, dia mengirim lamaran kepada Leslie. Dua hari kemudian, Kelsey dan Leslie setuju untuk bergerak maju, dan Kelsey mendirikan bisnis pertamanya, sebuah perusahaan produksi kursus yoga online. Karena kami berdua bekerja dari rumah, saya tiba-tiba mulai mendengar lebih banyak tentang yoga. Bagian dari proses produksi melibatkan menonton rekaman mentah dari setiap kelas, kemudian membuat catatan terperinci untuk editor video dan transkripsi. Akibatnya,

Saya akhirnya mendengarkan potongan-potongan kursus Leslie saat Kelsey memproduksinya.

Saat Saya Memutuskan untuk Serius Tentang Yoga

Salah satu hal pertama yang menarik minat saya adalah nada suara Leslie. Dia tidak terdengar seperti hippie woo-woo, itulah yang saya harapkan. Sebaliknya, ia memiliki pengalaman langsung selama tiga dekade dalam anatomi, fisiologi, dan kedokteran olahraga, dan memiliki toleransi yang sangat sedikit untuk terminologi yang sangat kecil. Dari sudut pandang Leslie, yoga sangat berharga terutama karena berguna. Yoga, sebagai latihan, sangat baik dalam membangun kekuatan, meningkatkan kemampuan, dan mempertahankan rentang gerak. Yoga bahkan lebih efektif ketika Anda fokus pada aspek pernapasan dari latihan. Kebanyakan orang menganggap latihan yoga sebagai kombinasi aneh dari aerobik, senam, dan kontorsi, tetapi itu bukan gambaran yang akurat. Apa yang membuat yoga yoga adalah menggabungkan pernapasan, gerakan, dan kondisi mental yang penuh perhatian. Leslie juga menjelaskan beberapa keanehan dalam terminologi yoga. Praktisi awal bereksperimen dengan tubuh dan pikiran. Mereka mencoba melakukan sains sebelum disiplin ilmu modern seperti anatomi, biologi, dan psikologi kognitif diresmikan. Akibatnya, ketika praktisi awal ini menemukan sesuatu yang menarik, satu-satunya pilihan mereka adalah menjelaskan apa yang mereka temukan dalam istilah cerita atau metafora. Misalnya, ketika praktisi yoga menemukan bahwa pose spesifik dan teknik pernapasan cenderung memicu respons emosional tertentu, mereka menjelaskan apa yang mereka perhatikan dengan cara terbaik yang mereka bisa. Itulah asal usul konsep seperti chakra: secara ilmiah, ada genangan energi tersembunyi di usus, sakrum, solar plexus, jantung, tenggorokan, dahi, dan mahkota Anda. Namun, sebagai metafora, mereka membantu praktisi awal berbicara tentang sesuatu yang mereka miliki

mengalami secara internal. Metafora itu cukup berguna sehingga mereka bertahan. Sedotan terakhir datang dalam bentuk video YouTube yang disajikan Leslie selama salah satu kelas online pertamanya. Dalam video tersebut, GilHedley,⁴ seorang guru anatomi manusia, memimpin opsi pendidikan mayat manusia untuk sekelompok siswa.⁵ Pada satu titik dalam video, Gil menyoroti fitur menarik dari fascia otot, lapisan jaringan brous yang membungkus otot-otot kita. Sama seperti ligamen dan tendon, fascia adalah jaringan ikat: ia mengikat tubuh kita bersama-sama. Dengan membungkus kelompok otot, fascia kita membantu kita bergerak dengan membiarkan otot kita meluncur satu sama lain lebih mudah. Namun, ada kelemahan: ketika lapisan fascia ini beristirahat, untai jaringan ikat "kabur", yang kira-kira memiliki penampilan dan konsistensi permen kapas, mulai tumbuh di antara fascia. Biasanya, itu bukan masalah besar. Untai individu sangat tipis, sehingga geser otot satu sama lain mudah mematahkannya. Masalah datang ketika Anda tidak cukup bergerak. Jika Anda tidak menggerakkan otot Anda untuk jangka waktu tertentu, "bulu" tetap ada dan secara bertahap menumpuk seiring waktu. Ketika bulu menjadi cukup tebal, itu dapat mengeras, membatasi rentang gerak normal Anda. Saya tahu saya tidak cukup bergerak, jadi video ini menarik perhatian saya. Apakah "kerenyahan" yang saya rasakan adalah cara tubuh saya memberi tahu saya bahwa otot-otot saya kabur? (Itu menjeramkan!) Itu adalah salah satu penjelasan potensial, tetapi itu bukan satu-satunya kemungkinan. Menurut Leslie, nyeri punggung dan leher juga dapat disebabkan oleh kekurangan oksigen karena otot-otot yang bersangkutan, situasi called ischemia. Kurangnya gerakan dapat menyebabkan kekurangan darah ke otot. Ketika otot kekurangan oksigen, reseptor nyerinya

mulai berulang, menciptakan rasa sakit yang terus-menerus dan tumpul itu. Semakin lama kondisinya bertahan, semakin banyak rasa sakit yang akan Anda alami. Stres kronis dan pola pernapasan yang tertekan juga berkontribusi pada iskemia. Penyebab ini sering berjalan bersamaan: adalah umum bagi orang untuk menahan napas ketika mereka mengalami stres. Jika Anda membantu otot Anda mendapatkan lebih banyak oksigen dengan bergerak, bernapas, dan mengurangi tingkat stres umum Anda, nyeri otot akan hilang.⁶ Terlepas dari itu, gerakan yang sering dan asupan oksigen penting dalam meredakan nyeri otot dan mempertahankan rentang gerak. Yoga menggabungkan gerakan, pernapasan, dan meditasi, sehingga biasanya sangat efektif dalam mengurangi atau menghilangkan nyeri otot kronis. Itu bukti yang cukup bagi saya. Saya akan belajar cara melakukan yoga. Sekarang... bagaimana cara memulai?

Apa sebenarnya "yoga" itu?

Jika saya akan melakukan yoga, saya perlu memiliki pemahaman mendasar tentang apa yang saya lakukan. Jika yoga hanya terdiri dari peregangan, orang hanya akan menyebutnya "peregangan." Ternyata, kunci untuk mengidentifikasi esensi yoga adalah memahami dari mana asalnya dan bagaimana itu menjadi apa yang kita sebut "yoga" hari ini.⁷ Yoga telah ada dalam berbagai bentuk selama ribuan tahun. Bukti yoga paling awal yang dapat dikenali berasal dari artefak ditemukan dari situs arkeologi yang terkait dengan peradaban Lembah Indus, yang terletak di Pakistan saat ini dan India barat laut dari 3300 hingga 1300 SM. Yoga, sebagai filsafat dan praktik, adalah seperangkat teknik nonreligius yang terjalin dengan tradisi agama dan filosofis di wilayah tersebut. Pendeta Weda kuno menyelesaikan ritual fisik yang rumit untuk menghubungkan dunia fisik dengan ilahi untuk mencari Brahma, "dasar tertinggi dari semua

menjadi." Kata "yoga" berasal dari kata Sansekerta kuno foryoke. Dalam pengertian yang sama bahwa seekor lembu melekat pada bajak untuk bekerja, melalui ritual mereka, para imam mencoba mengikat dunia spiritual dengan dunia fisik. Seiring waktu, tradisi Weda memudar. Para pendeta mulai mengeksplorasi gagasan tentang atman, esensi dari diri manusia. Ritual fisik eksternal beralih ke meditasi introspektif, dan praktisi spiritual mulai meninggalkan dunia material dan ritual yang kompleks demi menjelajahi hutan, berlatih meditasi, dan mengambil sumpah kemiskinan dan pertapaan untuk mencari kebenaran tertinggi. Yājñavalkya, seorang bijak yang hidup di apa yang sekarang disebut periode Upanishadik, mengusulkan bahwa ego manusia yang mencengkeram mencegah mereka mengetahui diri kita yang sebenarnya. Dengan menghancurkan ego, kita dapat menjadi satu dengan atman kita, "jiwa" kita atau diri esensial. Yājñavalkya juga memperkenalkan gagasan karma, yang dia bantah sebagai tindakan yang diambil untuk mencapai pembebasan dari ego. Dengan memisahkan tubuh dari theatman, adalah mungkin untuk mencapai apa yang kemudian disebut Buddha sebagai pencerahan.

Masukkan Asana

Wawasan filosofis Yājñavalkya mengarah pada pengembangan apa yang sebagian besar praktisi anggap sebagai inti dari praktik yoga kontemporer: postur fisik yang dikombinasikan dengan pernapasan dan meditasi yang terkontrol. Seperti yang dijelaskan oleh sarjana sejarah agama Karen Armstrong dalam *The Great Transformation*:⁸

Yoga adalah salah satu pencapaian terbesar India dan, dalam bentuknya yang paling berkembang, hampir pasti pertama kali dirancang [oleh praktisi] untuk melepaskan purusha [diri esensial] dari keterikatan alam. Yoga klasik ini sangat berbeda dari versi yoga yang sering diajarkan di Barat saat ini. Itu bukan latihan aerobik, dan itu tidak membantu orang untuk merasa lebih baik tentang hidup mereka—justru sebaliknya. Yoga adalah serangan sistematis terhadap ego, rejimen yang menuntut yang selama periode waktu yang lama mengajarkan calon untuk menghapuskan kesadaran normalnya dengan kesalahan dan khayalan, dan menggantinya dengan penemuan purusha-nya yang gembira.

Yoga adalah pekerjaan penuh waktu. Itu bukan untuk percobaan atau yang lemah hati: itu adalah disiplin spiritual yang menuntut dan menuntut, domain eksklusif para bhikkhu dan guru. Yogi klasik tidak berlatih untuk menjadi lebih kuat atau lebih mudah dipercaya. Mereka berusaha untuk memutuskan hubungan antara tubuh mereka dan atman mereka. Itu berat.

Kodifikasi Yoga

Pada abad kedua SM, seorang sarjana bernama Patañjali mulai menyusun dan mengkurasi filosofi dan praktik yoga pada saat itu. Ringkasan Patañjali tentang praktik terbaik (raja, atau "raja") menjadi The Yoga Sutras of Patañjali, kumpulan pepatah yang kemudian menjadi teks dasar praktisi yoga. Dalam The Yoga Sutra, Patañjali menguraikan delapan aspek mendasar, atau "anggota badan" dari latihan yoga raja:

Yama—moralitas
Niyama—
pemurnian diri
Asana—
postur
Pranayama—pengendalian
napas
Pratyahara—pengendalian
indera
Dharana—niat
Dhyana—
meditasi
Samadhi—
penyerapan/kontemplasi

Patañjali menyebut sistem ini "ashtanga," atau yoga "delapan kaki". Dengan rajin mempraktikkan kedelapan anggota tubuh yoga, praktisi akan mengalami kaivalya: pelepasan sempurna dari jiwa praktisi dari dunia material, yang mengarah pada kebahagiaan abadi. Apa yang kebanyakan orang pikirkan ketika mereka mendengar kata "yoga"—membengkokkan diri ke dalam postur yang aneh—hanyalah satu anggota badan, asana. Asana biasanya dipraktekkan dengan teknik pernapasan khusus yang disebut pranayama. Maksud latihan adalah untuk mempersiapkan meditasi (dhyana).

Sistem Patañjali berfokus pada yoga sebagai filosofi, bukan sebagai sistem olahraga. Asana sebagian besar terbatas pada pose duduk statis—jauh dari putaran senam yang akrab dengan praktisi modern. Itu tidak berarti pose intens tidak ada. Yogapose hatha (kuat) telah ada sejak Yājñavalkya, tetapi atletis bukanlah perhatian utama. Kompilasi praktik hatha yoga yang paling terkenal adalah HathaYoga Pradipika oleh Maharishi Swatmarama, seorang bijak abad ke-11. Swatmarama menganggap hatha yoga sebagai cara ideal untuk mencapai raja yoga Patañjali. Dengan memurnikan tubuh melalui olahraga, berlatih pranayama, dan menghirup asap secara ritual saat melakukan frontstand, Swatmarama percaya praktisi hatha dapat mencapai tingkat kesadaran yang lebih tinggi. Bentuk awal hatha yoga tidak pernah benar-benar tertangkap. Latihan postur fisik yang intens tetap menjadi oshoot yang tidak jelas dari raja yoga sampai seorang Brahman dari Mysore, India, merevolusi praktik empat abad kemudian.

Pria yang "Menemukan" Yoga Modern

Tirumalai Krishnamacharya lahir pada tahun 1888, putra tertua SriTatacharya, seorang pendeta Brahmana yang terkenal.⁹ Krishnamacharya diperkenalkan dengan laku hatha oleh ayahnya pada usia ve, dan melanjutkan praktik hatha yang ketat sepanjang pendidikan formalnya. Pada tahun 1919, ia mengatur untuk melakukan perjalanan ke Gunung Kailash di Tibet untuk belajar di bawah salah satu master hatha yoga terakhir yang tersisa, Sri RamamohanaBrahmachari. Krishnamacharya belajar di bawah Brahmachari selama tujuh setengah tahun, menghafal Yoga Sutra Patañjali dan berlatih asana. Dia juga, seperti yang diceritakan, mengambil beberapa keterampilan yang menarik perhatian, seperti memperlambat pernapasannya dan menghentikan denyut nadinya untuk waktu yang lama. Krishnamacharya kemudian menggunakan ini

keterampilan untuk eec yang hebat, menunjukkannya di depan umum sebagai cara mempopulerkan latihan hatha.

Permintaan Guru

Hubungan guru/siswa memiliki sejarah yang panjang dan mapan di India. Pada akhir masa jabatan siswa dengan gurunya, adalah kebiasaan bagi siswa untuk memberikan pembayaran kepada guru, sebuah praktik yang disebut guru dakshina, sebagai tanda terima kasih dan rasa hormat yang mendalam terhadap ajaran guru. Seringkali, pembayaran terdiri dari uang atau barang-barang material, tetapi kadang-kadang guru meminta penyelesaian tugas khusus. Ketika studi Krishnamacharya selesai, permintaan Brahmachari sederhana: Krishnamacharya harus menikah, memiliki anak, dan mengajar yoga yang telah dia pelajari di Gunung Kailash. Permintaan Brahmachari sangat mengejutkan. Krishnamacharya telah ditunjuk untuk beberapa posisi agama dan pendidikan berstatus tinggi, dengan harapan dia akhirnya akan mengasumsikan posisi kakek buyutnya sebagai pemimpin agama utama India Selatan. Untuk memahami kejutan status, bayangkan ditunjuk sebagai CEO perusahaan bergengsi, hanya untuk diberitahu bahwa Anda berkewajiban untuk bekerja sebagai petugas kebersihan sebagai gantinya. Brahmachari's dakshina adalah hukuman permanen dari kesulitan dan status rendah. Krishnamacharya menghormati permintaan gurunya dan kembali ke Mysore, hidup dalam kemiskinan yang parah dan mencari kesempatan untuk mengajar hatha yoga. Pada tahun 1931, Krishnamacharya diundang untuk mengajar hatha di Perguruan Tinggi Sanskerta di Mysore. Di sana, ia diperkenalkan dengan Krishna Raja Wadiyar IV, maharaja Mysore yang berkuasa. Terkesan dengan kemampuan dan beasiswa Krishnamacharya, dan berterima kasih atas bantuan Krishnamacharya dalam mengelola diabetesnya, maharaja mengundang Krishnamacharya untuk membuka sekolah yoga di istana di bawah perlindungannya.

Karena sebagian besar siswa di yoga shala di Mysore adalah anak laki-laki yang energik, Krishnamacharya mengembangkan bentuk baru latihan yoga hatha yang menekankan membangun kekuatan dan meningkatkan kemampuan. Dia menggabungkan hatha asana dan pranayama tradisional dengan gerakan dari senam Inggris, mengadaptasi dan memodifikasi urutan untuk setiap siswa individu.¹⁰

Yoga Baru

Itulah asal-usul dari apa yang sekarang dikenal sebagai ashtanga vinyasayoga. Jika Anda pergi ke kelas yoga di gym atau mengambil buku apa pun yang berisi kata-kata ashtanga, vinyasa, "Kekuatan", "Aliran", "Inti", "viniyoga, atau Iyengar, Anda berlatih dalam tradisi Krishnamacharya. Krishnamacharya menyesuaikan pengajarannya dengan kebutuhan murid-muridnya. Ashtanga yoga dipopulerkan oleh salah satu pemuja pertama Krishnamacharya, K. Pattabhi Jois, yang belajar dengannya di Mysore. Ashtanga terutama berfokus pada enam urutan pose yang kuat, dimulai dengan "Seri Pratama," yang diajarkan Krishnamacharya di yoga shala. Praktik ashtanga saat ini mempertahankan penekanan pada urutan dan atletisme. Ketika menginstruksikan saudara iparnya, B. K. S. Iyengar, Krishnamacharya menekankan keselarasan tubuh dalam asana, mengingat kesehatan Iyengar yang rapuh pada saat itu. Iyengar melanjutkan untuk mengembangkan pendekatan ini menjadi gaya yoga yang berorientasi pada penyalarsan yang dipraktikkan saat ini. Di kemudian hari, ketika Krishnamacharya menghabiskan sebagian besar waktunya bekerja sebagai penyembuh, ia mengajari putranya, T. K. V. Desikachar, aspek terapeutik dari praktik asana. Saat ini, siswa Desikachar menekankan yoga sebagai alat untuk kesehatan, dan mengeksplorasi berbagai aplikasi kesehatan potensial, mulai dari terapi fisik hingga menghilangkan kecemasan. Akibatnya, hampir setiap bentuk utama modernyoga telah sangat dipengaruhi oleh ajaran Krishnamacharya.

Apa yang tidak berubah adalah penekanan pada orang normal dengan tanggung jawab sehari-hari. Yoga Krishnamacharya bukan untuk mereka atau pertapa: fokusnya pada latihan yang dipersonalisasi untuk orang sehari-hari membuat asana dapat diakses oleh jutaan orang sibuk di seluruh dunia. Dalam kata-kata T. K. V. Desikachar: "Anda melakukan yoga sehingga Anda dapat menjalani hidup Anda, bukan sebaliknya."

Yoga = Pernapasan + Gerakan + Meditasi

Kembali ke masa kini. Apa hubungannya sejarah ini dengan yoga modern? Inilah yang mengejutkan saya: praktik asana modern, secara historis, adalah penemuan baru-baru ini. Tentu, potongan-potongan filosofis yoga telah ada selama beberapa ribu tahun, tetapi praktik sebenarnya menghabiskan waktu untuk mengambil postur sambil bernapas dan bermeditasi, secara historis, praktis baru. Penekanan pada kekuatan dan keunggulan dalam latihan yoga juga baru. Sangat sedikit dari kita yang secara aktif mencoba untuk secara paksa memisahkan jiwa kita dari tubuh kita untuk mencari atman kita. Sebagian besar praktisi yoga modern bukanlah pertapa atau penyerah. Secara umum, praktisi yoga saat ini terutama peduli dengan tetap t, meningkatkan kemampuan, dan menghilangkan stres. Pada intinya, jelas bahwa latihan yoga modern terdiri dari menggabungkan gerakan, pernapasan, dan meditasi. Posenya sendiri tidak ajaib. Mereka telah banyak berubah selama berabad-abad, dan Krishnamacharya tidak malu menambahkan elemen baru atau memodifikasi praktik untuk siswa. Juga jelas bahwa yoga bukan tentang gerakan abu. Ada kecenderungan filosofis untuk praktik ini: pengakuan bahwa Anda menghabiskan waktu dengan cara ini untuk melayani menjadi orang yang lebih baik. Meskipun asana modern relatif baru, filosofi yoga telah ada selama ribuan tahun, dan itu

menarik dengan sendirinya. Yoga sangat dalam dengan cara yang tidak dimiliki oleh mode terbaru. Juga tidak perlu berusaha keras untuk berlatih pose gila. Menjadi yogi "sejati" tidak berarti dapat mengasumsikan posisi seperti Gumby: itu berlatih dengan cara yang membantu Anda menjalani kehidupan yang lebih baik. Tidak bisa berpose karena Anda tidak cukup berpose, cukup kuat, atau mengalami cedera? Tidak masalah: sesuaikan pose, atau lakukan yang berbeda. Selama Anda fokus pada napas Anda saat Anda bergerak, dan Anda memperhatikan apa yang dikatakan tubuh Anda, Anda melakukannya dengan benar. Pada intinya, latihan yoga modern adalah tentang mengintegrasikan pernapasan, gerakan, dan meditasi kesadaran. Ini bukan tentang senam, akrobatik, atau menjadi super exhibe. Menjadi super exhibe atau acrobatic tidak membuat Anda menjadi yogi yang lebih baik.

Menjernihkan Kesalahpahaman

Berikut adalah beberapa kesalahpahaman umum tentang yoga:

Anda tidak perlu mengetahui ribuan pose untuk melakukan yoga. Mengetahui banyak pose tidak membuat Anda menjadi yogi yang lebih baik. Krishnamacharya sendiri memulai dengan dua puluh empat. Anda tidak perlu mengetahui nama pose dalam bahasa Sanskerta, atau bahkan nama bahasa Inggris dari pose tersebut. Mengetahui banyak pose atau nama mereka tidak membuat Anda lebih baik yogi. Beberapa praktisi yoga menganut agama seperti Hindu, tetapi keyakinan agama tidak diperlukan untuk mendapatkan manfaat dari praktik asana. Beberapa praktisi yoga mengadopsi diet seperti vegetarianisme, veganisme, atau praktik Ayurveda, tetapi diet khusus bukanlah prasyarat untuk latihan asana. Beberapa praktisi yoga juga tertarik dengan pengobatan alternatif, astrologi, dan topik serupa lainnya, tetapi minat ini bukanlah prasyarat untuk latihan asana.

Semua hal ini adalah kesalahpahaman yang saya miliki ketika saya pertama kali mendengar tentang yoga. Sedikit waktu yang dihabiskan untuk mempelajari apa sebenarnya latihan yoga sudah cukup untuk meringankan kekhawatiran awal saya, serta mencegah saya membuang-buang waktu untuk hal-hal yang tidak penting.

Apakah Yoga Berbahaya?

Ada satu hal terakhir yang saya khawatirkan: Bisakah saya melukai diri saya sendiri saat melakukan yoga? Sekitar waktu saya meneliti asana, sebuah artikel tentang latihan yoga diterbitkan di New York Times. Artikel, berjudul "Bagaimana Yoga Dapat Menghancurkan Tubuh Anda,"¹¹ ditulis oleh jurnalis sains Times, William J. Broad. Esai itu pada dasarnya adalah kutipan dari buku Broad, *The Science of Yoga: The Risks and the Rewards* (2012), yang dijadwalkan untuk dirilis beberapa minggu setelah artikel tersebut diterbitkan. Artikel itu memicu banyak meremas tangan di dunia yoga. Broad mengutip beberapa penelitian ilmiah tentang manfaat asanapraktik, sebagian besar terkait dengan rentang gerak dan keseimbangan emosional. Dia kemudian menyerang, menyarankan latihan yoga bertanggung jawab atas berbagai cedera serius, mulai dari keseleo hingga stroke. Dalam beberapa kasus, Broad menyarankan yoga bertanggung jawab atas praktisi yang mengalami koma. Topik ini adalah contoh pembalikan yang berguna. Seperti apa rasanya melakukan yoga dengan tidak benar, dan apa risiko yang paling signifikan dari latihan asana? Tanggapan komunitas pengajar yoga terhadap artikel dan buku Broad cepat dan bersemangat: Mengingat bukti, Broad sensationalized. Yoga, seperti aktivitas fisik lainnya, memiliki risiko, tetapi risiko tersebut tidak mungkin membahayakan Anda secara serius jika Anda berlatih dengan cara yang aman. Meskipun memang mungkin untuk melukai diri sendiri saat melakukan yoga, masalah kesehatan utama seperti stroke dan koma sangat jarang terjadi, dan tidak ada bukti yang menunjukkan latihan asana meningkatkan risiko cedera besar. Dibandingkan dengan aktivitas fisik lainnya, bahkan olahraga nonkontak seperti golf, yoga secara statistik sangat aman. Adapun keseleo otot, ketegangan, dan robekan, jenis cedera tersebut tentu saja mungkin terjadi jika Anda memaksa tubuh Anda berpose, mengabaikan rasa sakit

dan ketidaknyamanan. Jika Anda percaya bahwa yogi "sejati" harus dapat menyentuh hidung mereka ke lutut mereka, dan Anda memaksa tubuh Anda ke posisi itu, Anda berisiko terkilir ligamen atau robek paha belakang Anda. Demikian juga, jika seorang guru yoga secara paksa menyesuaikan Anda di jarak jauh yang melebihi batas tubuh Anda, Anda kemungkinan besar akan mengalami cedera. Salah satu area potensial yang signifikan untuk cedera adalah tulang belakang serviks. Beberapa pose, seperti headstand, memberi tekanan pada kepala dan leher, dan jika Anda jatuh dengan cara yang salah saat melakukannya, Anda mungkin melukai leher atau tulang belakang Anda secara serius. Berdasarkan informasi ini, selama saya mendengarkan tubuh saya, tidak memaksakan diri saya untuk berpose, dan menghindari beban yang tidak perlu pada tulang belakang mycervical, saya dapat menuai manfaat latihan tanpa risiko cedera yang signifikan. Itu bagus untuk diketahui.

Asana Minimum yang Layak

Saya siap untuk mulai berlatih ... jadi dari mana saya harus memulai? Jelas bahwa ada banyak jenis, bentuk, dan filosofi seperti apa yoga "seharusnya" terlihat. Itu semua baik dan bagus, tetapi saya perlu mundur selangkah dan memutuskan apa tujuan yoga bagi saya. Mengapa saya melakukan ini? Saya ingin lebih aktif secara fisik, dan tubuh saya menjelaskan bahwa saya perlu bergerak lebih banyak. Dokter saya telah merekomendasikan latihan rentang gerak, dan menyebutkan yoga sebagai pilihan. Saya telah membaca tentang manfaat olahraga dan meditasi, dan saya suka yoga menggabungkan keduanya. Saya tidak punya waktu untuk pergi ke studio untuk kelas panjang, jadi saya ingin dapat berlatih di rumah dengan cara yang aman dan aman. Itu cukup detail untuk tingkat kinerja target yang berguna. Ini milikku:

1. Pelajari urutan postur fisik yang dapat saya latih untuk meningkatkan tingkat aktivitas fisik, kekuatan, dan kemampuan umum harian saya.

2. Gabungkan olahraga dengan pernapasan dan meditasi, menuai beberapa manfaat fisik dan mental utama dalam satu periode waktu.3. Berlatihlah dengan aman dari ingatan selama dua puluh hingga tiga puluh menit, sesuaikan durasi, kekuatan, kemampuan, dan tingkat energi umum.

Peralatan Yoga

Sebelum saya mulai berlatih, saya membutuhkan beberapa peralatan dasar. Pertama dan terpenting adalah tikar, yang memberikan beberapa bantalan dan mencegah tangan dan kaki tergelincir keluar dari posisinya. Latihan yoga bisa berkeringat, dan jika Anda terpeleset, itu dapat menyebabkan cedera, jadi mendapatkan goodmat itu penting. Ketika saya berlatih di Pure Yoga, mereka menyediakan tikar, yang bagus. Namun, saya perhatikan bahwa tikar itu agak terlalu pendek untuk saya. Saya sering tergelincir di tepi, dan saya harus berhenti dan menyesuaikan. Mungkin yang terbaik jika saya bisa menggunakan tikar yang lebih panjang. Setelah sedikit penelitian, saya memilih tikar MandukaPRO delapan puluh inci.¹² Ini ekstra panjang, sangat mencengkeram, sangat tahan lama, dan kebetulan berwarna hijau yang bagus. Bekerja untuk saya. Selain itu, saya mengambil tas tikar dan beberapa handuk tangan, untuk berjaga-jaga tangan saya berkeringat. Saya sudah memiliki celana pendek dan T-shirt, serta beberapa pasang celana yoga, jadi saya sudah siap dalam hal pakaian. Asana dipraktikkan tanpa alas kaki, jadi saya tidak membutuhkan sepatu khusus. Atas saran Kelsey, saya membeli tali katun panjang dan balok bambu, yang menurutnya dapat membantu saat memodifikasi pose tertentu. Itu semua peralatan yang saya butuhkan, yang bagus. Anda tidak perlu banyak itu untuk melakukan yoga.

Mempelajari Pose

Inilah strategi awal saya: Kelsey adalah guru yoga terlatih, jadi masuk akal untuk memulai dengan instruksinya. Kami menyisihkan sembilan puluh menit pada suatu sore untuk membahas dasar-dasarnya.

Jika saya tidak memiliki guru yoga dalam keluarga, saya akan memesan lestarsi privat dengan instruktur lokal. Pelatihan satu lawan satu untuk jenis pembelajaran ini sangat membantu, karena gerakan sulit untuk belajar dari sebuah buku. Itu tidak berarti saya melewatkan buku. Saya sudah mengambil beberapa sumber daya:

Anatomi Yoga oleh Leslie Kamino dan Amy Matthews (2011) Yoga untuk Kesehatan oleh Gary Kraftsow (1999) Ashtanga Yoga: Praktik dan Filsafat oleh Gregor Maehle (2007) Yoga Berpusat pada Napas dengan Leslie Kamino (DVD) oleh Leslie Kamino (2010)

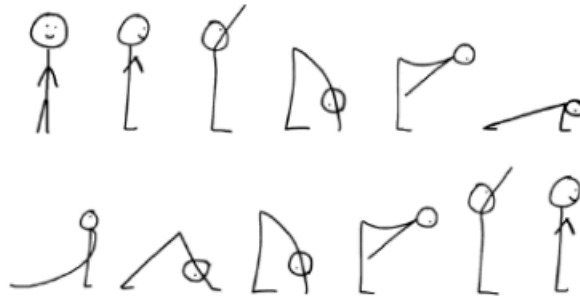
Saya menggunakan sumber daya ini terutama sebagai referensi. Mereka berisi foto dan ilustrasi pose, serta instruksi terperinci tentang cara melakukan setiap pose dengan benar, jadi saya akan mencari informasi seperlunya. Saya akan tahu lebih banyak tentang apa yang perlu saya pelajari setelah saya mulai berlatih. Kelsey dan saya menggulung tikar kami dan mulai. Pertama: Salam Matahari.

Urutan Salam Matahari

Salam Matahari adalah urutan postur yang dirancang untuk membantu Anda menghangatkan tubuh Anda, jadi sebagian besar latihan asana dimulai dengan beberapa putaran.

Catatan singkat jika Anda memutuskan untuk mencobanya sendiri: Setiap orang berbeda, dan cara "benar" untuk melakukan setiap pose tergantung pada batas unik tubuh Anda. Jika Anda memperhatikan apa yang dikatakan tubuh Anda, hindari melakukan ekstensi berlebihan, dan menyesuaikan setiap pose dengan batas pribadi Anda, Anda akan dapat berlatih asana dengan aman.

Begini caranya:



1. Tarik napas. Berdiri tegak di depan matras, lengan rileks di samping Anda, lututlembut.
2. Napas. Angkat tangan Anda ke posisi "doa" universal di depan dada Anda.
3. Tarik napas sambil mengangkat dan keluar lengan ke samping sampai terentang sepenuhnya di atas kepala Anda. Kemudian, lihat tangan Anda. (Jika itu terasa nyaman, lihat lurus ke depan.)
4. Buang napas dan lipat ke depan, berengsel di pinggul Anda untuk membawa tangan Anda ke bawah ke theoor untuk menyentuh jari-jari kaki Anda. Pergi sejauh yang Anda bisa tanpa terlalu panjang; Tidak apa-apa untuk menekuk lutut Anda. Rilekskan bahu dan leher Anda.
5. Tarik napas dan lihat dinding di depan Anda sementara tangan Anda tetap bersentuhan dengan oor. (Atau, cobalah untuk meluruskan kaki Anda sambil menjaga kepala dan leher Anda tetap rileks.)
6. Buang napas saat Anda kembali ke tikungan ke depan.
7. Tarik napas saat Anda menginjakkan kaki Anda kembali ke "posisi push-up", sehingga tubuh Anda terentang sepenuhnya dengan lengan dan bagian bawah jari kaki menopang berat badan Anda. Anda dapat berjalan kaki ke belakang atau melompat, sesuai keinginan Anda. (Pose ini disebut Plank.)
8. Buang napas dan turunkan diri Anda ke tanah, seolah-olah Anda sedang melakukan push-up. Jangan beristirahat di tanah; Berhentilah satu atau dua inci dari matras dan berhenti. Pastikan untuk menjaga sudut sembilan puluh derajat di bahu, siku, dan pergelangan tangan Anda untuk mencegah cedera. Jika ini terlalu berat pada saat pertama, tidak apa-apa untuk membawa lutut Anda ke ori. (Pose ini disebut Chaturanga Dandasana, biasanya disingkat menjadi Chaturanga.)
9. Tarik napas saat Anda mendorong ke depan, melengkungkan punggung Anda saat Anda mendorong ke atas sampai tubuh Anda tegak. Pada saat yang sama, "gulung" kaki Anda sehingga bagian atas setiap kaki menyentuh tanah, dan kaki Anda satu inci atau lebih dari tikar. (Pose ini disebut Anjing Menghadap ke Atas, karena menyerupai peregangan anjing.)
10. Buang napas dan gulung kaki Anda ke belakang, sehingga bagian bawahnya menyentuh matras. Pada saat yang sama, gerakkan tulang ekor Anda ke atas dan ke belakang hingga tubuh Anda dalam bentuk V terbalik. Rilekskan leher dan bahu Anda. Jika Anda merasakan sensasi peregangan di

paha belakang, dapat membantu membayangkan diri Anda "bernapas" ke dalamnya. Tetap dalam posisi ini selama tiga hingga tiga napas. (Pose ini sering disebut Anjing Menghadap ke Bawah.)¹¹. Tarik napas, lalu buang napas saat Anda berjalan atau melompat kaki ke depan, sampai kaki Anda berada di bawah bahu Anda.¹². Ulangi langkah 5 dan 6.¹³. Tarik napas sambil mengangkat dan keluar lengan ke samping sampai terentang sepenuhnya di atas kepala Anda, seperti pada langkah 3.¹⁴. Turunkan lengan Anda di depan Anda sampai tangan Anda berada dalam posisi "doa" universal di depan dada Anda, seperti pada langkah 2.

Anda akan melihat urutan berakhir pada posisi yang sama seperti dimulai. Setelah Anda menyelesaikan urutan, Anda dapat mengulanginya sebanyak yang Anda suka, atau dengan mudah beralih ke pose berdiri yang berbeda. Juga, perhatikan bagaimana setiap langkah dimulai dengan "menarik napas" atau "menghembuskan napas"? Itu disengaja: setiap gerakan sesuai dengan napas. Berfokus pada napas adalah apa yang membuat yoga berbeda dari aerobik. Anda dapat menganggap urutan Salam Matahari sebagai "asana minimum yang layak": jika Anda benar-benar mau, seluruh latihan Anda dapat terdiri dari melakukan Salam Matahari berulang kali. Urutan ini mengenai semua kelompok otot utama tubuh, membutuhkan perpaduan kekuatan dan kebolehpercayaan yang baik, dan menantang tanpa rumit. Krishnamacharya benar-benar tahu apa yang dia lakukan. Dalam praktik ashtanga tradisional, Salam Matahari (juga disebut avinyasa) dilakukan di antara setiap pose individu. Itu berarti Anda menghabiskan sebagian besar Seri Utama selama dua jam untuk melakukan urutan sederhana ini. Ini adalah contoh yang baik dari hukum kekuasaan dalam tindakan: minoritas pose membentuk mayoritas latihan. Namun, tubuh manusia rentan terhadap cedera stres berulang: jika Anda melakukan Sun Salutations secara berlebihan, Anda dapat melukai diri sendiri.¹³ Itulah sebabnya adalah ide yang baik untuk melakukan beberapa set Sun Salutations untuk melakukan pemanasan, lalu beralih ke pose yang berbeda.

Ingatlah untuk bernapas seperti Darth Vader

Saat berlatih asana, biasanya menggunakan teknik pernapasan khusus yang disebut ujjayi. Ini adalah teknik pernapasan paling sederhana (pranayama) yang digunakan dalam latihan yoga, dan dirancang untuk membuatnya lebih mudah untuk fokus pada napas. Di bagian belakang tenggorokan Anda ada struktur kecil yang disebut glotis, di area yang kira-kira sama dengan amandel dan pita suara Anda. Jika Anda fokus pada area itu sejenak, Anda dapat secara sadar mengontrak glotis dengan sangat sedikit, yang menyempitkan udara masuk dan keluar dari paru-paru Anda. Efek dari penyempitan ini terlihat: Anda dapat mendengar (dan merasakan) udara dingin mengalir ke paru-paru Anda saat Anda menarik napas, dan udara hangat keluar saat Anda menghembuskan napas. Yoga Sutra menggambarkan suara ujjayi sebagai "seperti lautan," tetapi saya lebih suka analogi modern: ujjayi terdengar seperti Darth Vader dari Star Wars. Jika Anda meniru pola pernapasan Dark Lord of the Sith yang terkenal, maka tutup mulut Anda, Anda akan melakukan ujjayi. Teks-teks yoga kuno mengaitkan kualitas mistis dengan ujjayi, serta teknik pranayama lainnya. Terlepas dari apakah ujjayi "membangun panas dalam tubuh" atau "mendorong ow prana [kekuatan hidup universal]", itu tentu membuatnya lebih mudah untuk memperhatikan pernapasan Anda. Tujuan utama ujjayi adalah untuk membantu menjaga pernapasan Anda tetap tenang, rileks, dan teratur. Dengan mempertahankan ujjayi saat Anda berlatih, jauh lebih mudah untuk menjaga pikiran Anda tetap fokus pada pernapasan Anda, yang membuat perhatian Anda tidak mengembara. Jika Anda mengikuti napas Anda dengan mengesampingkan yang lain, Anda bermeditasi tanpa benar-benar mencoba. Saya telah mempelajari urutan yang paling sering digunakan dalam asana, serta pola pernapasan utama, dalam waktu sekitar dua puluh menit. Sejauh ini baik-baik saja! Sekarang, mari kita pelajari beberapa pose berdiri yang umum.

Pose Berdiri

Prajurit 1



Pose prajurit adalah pokok dari kelas ashtanga dan vinyasa. Pose sering muncul setelah (atau di tengah) urutan Salam Matahari, dan dapat dilakukan sebagai urutan kecil itu sendiri. Inilah cara kerja postur Prajurit pertama: berdiri menghadap ke depan. Tarik napas dan mundur selangkah dengan satu kaki, biarkan jari-jari kaki belakang Anda mengarah sekitar empat puluh derajat ke samping. Angkat lengan Anda di atas kepala Anda, sejajar satu sama lain. Kaki depan harus ditekuk di lutut, kaki pada, dengan sinar depan kira-kira tegak lurus dengan oor. Kaki belakang harus berdetak di tanah. Tahan pose selama tiga hingga bernapas. Prajurit 2



Dari Warrior 1, buka lengan dan pinggul Anda untuk menghadap ke sisi tikar Anda. (Jika kaki kanan Anda berada di belakang, buka ke kanan, dan sebaliknya.)

Jaga agar lengan Anda sejajar dengan tanah, "melebar" melalui dada. Tahan pose selama tiga hingga bernapas. Prajurit 3



Warrior 3 bisa dengan mudah disebut pose Superman. Tujuannya kali ini adalah untuk menggerakkan tubuh Anda ke depan sambil menyeimbangkan dengan satu kaki, dengan bagian tubuh Anda sejajar dengan oor. Dari posisi berdiri netral (berdiri biasa atau Prajurit 1), angkat tangan Anda di atas kepala, lalu angkat satu kaki sedikit ke arah itu. Gerakkan kaki itu ke belakang di ruang angkasa sambil mencondongkan tubuh ke depan dengan tubuh Anda, berengsel di pinggul sampai Anda berada di posisi Superman. Berfokuslah untuk menyeimbangkan otot-otot di kaki Anda, dan jaga agar lutut bagian bawah Anda tetap kokoh, tetapi tidak terkunci. Tahan pose selama tiga hingga tiga napas, lalu kembali ke posisi berdiri netral. The Kamino □ Spiral



Dari posisi berdiri netral, tarik napas dan biarkan lengan Anda berputar keluar dan ke atas, seperti di awal Salam Matahari. Dari sana, buang napas dan biarkan lengan Anda berputar kembali ke dalam dan ke bawah, sambil berengsel

Maju di pinggul sampai dada Anda dekat dengan lutut Anda. Ulangi beberapa kali, tarik napas saat Anda berputar ke atas, dan buang napas saat Anda melipat ke bawah. Ini adalah pose berdiri favorit saya: rasanya sangat enak setelah bangun dari tempat tidur, atau setelah beberapa jam di depan komputer. Manfaat dari variasi ini adalah tidak membebani tangan atau lengan, yang bagus jika Anda terkena sindrom terowongan karpal, pulih dari cedera, atau perlu membangun kekuatan dan kemampuan sebelum mencoba pose yang lebih menantang. Tikungan Maju



Dari posisi berdiri netral, engsel di pinggul dan lipat ke depan, biarkan lutut Anda menekuk jika perlu. Rilekskan leher dan bahu Anda, dan biarkan kepala Anda menggantung saat Anda mencoba menyentuh theoor dengan ngers Anda. Mainkan dengan menekuk dan meluruskan lutut, dan perhatikan di mana Anda mulai merasakan sensasi peregangan. (Jika paha belakang Anda sangat ekspirible, Anda mungkin dapat menyentuh theoor dengan telapak tangan Anda, tetapi jangan terlalu meregangkan diri Anda.) Setelah tiga kali bernapas, kembali berdiri. Tikungan Lebar ke Depan



Dari posisi berdiri netral, gerakkan kaki Anda terpisah ke setiap sisi, menciptakan posisi berbentuk V lebar yang bagus. Kemudian, engsel di pinggul dan lipat ke depan. Rilekskan leher dan bahu Anda dan biarkan kepala Anda menggantung saat Anda mencoba menyentuh oor dengan telapak tangan Anda. Setelah tiga kali bernapas, kembali berdiri. Segitiga



Dari posisi berdiri netral, gerakkan kaki Anda terpisah ke setiap sisi, menciptakan posisi berbentuk V yang lebar, kaki mengarah ke depan. Kemudian, putar pinggul sehingga salah satu kaki Anda mengarah ke sisi matras. Tekuk lutut itu dan biarkan siku Anda beristirahat di atasnya. Raih tangan bebas Anda ke arah langit-langit. Terakhir, luruskan lutut Anda yang ditekuk, dan biarkan tangan bawah Anda jatuh ke kaki atau oor. Tahan pose selama tiga napas, kembali ke posisi lebar, lalu ulangi di sisi lain. Sudut Samping yang Diperpanjang



Dari posisi berdiri netral, gerakkan kaki Anda terpisah ke setiap sisi, menciptakan posisi berbentuk V yang lebar, kaki mengarah ke depan. Kemudian, putar pinggul sehingga salah satu kaki Anda mengarah ke sisi matras. Tekuk itu

lutut dan biarkan siku Anda beristirahat di atasnya. Jangkau tangan bebas Anda ke arah langit-langit, lalu (jika nyaman), bawa telapak tangan bawah Anda ke oor. Lengan Anda harus tegak lurus dengan oor, membentuk garis lurus. Ada variasi dari Extended Side Angle yang sangat saya sukai: Anda dapat menyatukan tangan Anda dalam posisi "doa" universal alih-alih mengulurkan tangan Anda ke oor dan langit-langit. Setelah menahan pose selama tiga hingga tiga napas, kembali ke posisi lebar ke depan, lalu ulangi di sisi lain. Kursi



Berdiri tegak, kaki menyatu dan lengan di samping Anda. Tarik napas dan angkat lengan Anda lurus di atas kepala Anda. Kemudian buang napas dan tekuk lutut, jaga agar kaki Anda tetap bersama, sampai Anda berada dalam posisi duduk yang kira-kira, seperti Anda sedang duduk di kursi imajiner. (Ini adalah dicult, dan semakin dicult semakin lama Anda memegang pose.)Tahan pose selama tiga napas hingga ve, lalu luruskan untuk berdiri. Pohon



Berdiri tegak, kaki menyatu dan lengan di samping Anda. Angkat satu kaki ke tanah, tekuk lutut, dan pegang pergelangan kaki Anda dengan tangan yang sesuai. Seimbangkan kaki lainnya, arahkan kaki bebas ke bagian dalam paha bagian atas Anda, arahkan lutut Anda ke samping. Anda mungkin akan melihat otot-otot di kaki Anda yang membumi aktif seperti orang gila sampai Anda kehilangan keseimbangan Anda. Setelah kaki Anda berada di posisinya, lepaskan pergelangan kaki Anda dan satukan telapak tangan Anda di depan dada Anda dalam posisi "doa" universal. Tahan pose selama tiga hingga tiga napas, lalu ulangi dengan kaki lainnya.

Pose Lantai

Kucing/Sapi



Berlututlah dengan tangan dan lutut, lengan selebar bahu, dan kaki selebar pinggul. Jaga agar leher dan bahu Anda tetap rileks, dan lihat ke tanah. Buang napas saat Anda melengkungkan punggung, menarik perut Anda ke arah pinus Anda. Anda akan melihat ke belakang ke arah kaki Anda. Ini adalah posisi kucing. Selanjutnya, tarik napas saat Anda memperhatikan punggung Anda dan kembali ke netral, lalu melengkung ke arah yang berlawanan, tarik perut Anda ke arah oor,

seperti yang Anda lakukan di awal Salam Matahari. Anda akan melihat ke depan, dan punggung Anda akan cekung. Ini adalah posisi Sapi. Bernapaslah dan bergerak bolak-balik di antara posisi ini setidaknya beberapa kali. Duduk Tekuk Depan



Duduklah di atas matras, kaki terentang lurus di depan Anda. Jaga agar kaki Anda tetap tegak lurus, tegak lurus dengan oor. Tanpa menekuk kaki Anda, putar dari pinggul saat Anda melibatkan paha depan (otot-otot di bagian depan paha), lalu jangkau ke depan dengan tangan Anda ke arah kaki Anda. Raih sejauh yang Anda bisa tanpa rasa sakit atau ketidaknyamanan besar. Tahan pose selama tiga hingga bernapas. Tekuk Depan Satu Kaki



Pikirkan pose ini sebagai kombinasi dari Pohon dan Tekuk ke Depan. Angkat satu kaki, tekuk lutut, lalu putar kaki Anda ke samping. Letakkan kaki Anda di bagian dalam paha Anda, lalu engsel ke depan di pinggul, seperti pada Forward Bend. Raih sejauh yang Anda bisa tanpa rasa sakit atau ketidaknyamanan besar, lalu tahan pose selama tiga kali untuk bernapas ve.

Apa



Pose ini lebih sulit dari yang terlihat. Duduklah di atas matras, kaki terentang lurus di depan Anda. Jaga agar kaki Anda tetap terbuka, tegak lurus dengan teo. Letakkan tangan Anda di atas oor, lalu dorong tangan Anda ke dalam oor, angkat tubuh Anda sedikit saat Anda mencoba menjaga kaki Anda tetap lurus ke atas. Tahan pose selama tiga hingga vebreaths. Sudut Terikat



Duduklah di atas oor, dan angkat lutut Anda ke dada. Kemudian, biarkan lutut Anda terbuka ke samping sampai dekat dengan oor, dan telapak kaki Anda bersentuhan. Pegang kaki Anda dengan tangan Anda, dan pertahankan posisi ini selama tiga hingga tiga napas. Putaran tulang belakang duduk



Duduklah di atas matras, kaki terentang lurus di depan Anda. Bawa satu lutut ke dada Anda, sampai kaki Anda dekat dengan lutut yang berlawanan. Kemudian, angkat kaki Anda dengan tangan Anda dan letakkan di sisi lain kaki Anda yang terentang. Letakkan tangan Anda dengan telapak tangan ke bawah di tanah di pinggul Anda, seperti dalam pose Sta. Setelah Anda berada di posisinya, gerakkan lengan yang berlawanan di seluruh tubuh Anda sampai siku Anda menyentuh bagian luar lutut Anda. (Jika kaki kanan Anda ditekuk, Anda akan menggerakkan lengan kiri Anda.) Tekuk lengan di siku, dan jaga agar anjing Anda tetap menunjuk ke langit-langit. Lanjutkan putaran dengan mendorong lengan Anda dengan lembut ke kaki Anda. Selesaikan putaran sampai ke tulang belakang Anda dengan memutar kepala ke arah yang sama saat Anda memutar. Hal yang rapi tentang posisi ini adalah Anda benar-benar dapat melihat perbedaan dalam rotasi di berbagai bagian tulang belakang Anda. Menurut Leslie Kamino, tulang belakang lumbal hanya memiliki derajat rotasi ve . Tulang belakang toraks memiliki tiga puluh dua derajat, dan tulang belakang leher memiliki delapan puluh hingga sembilan puluh derajat, jadi ketika Anda diputar sepenuhnya, Anda harus dapat melihat ke belakang Anda. Tahan putaran selama tiga kali bernapas, lepaskan, lalu ulangi di sisi yang berlawanan. Perahu



Ini adalah yang sulit. Duduklah di atas matras, kaki terentang lurus di depan Anda. Tekuk lutut Anda sehingga Anda menyeimbangkan kursi Anda, dan jari-jari kaki Anda menyentuh oor. Rentangkan tangan lurus ke depan Anda, lalu jaga agar kaki Anda tetap bersama saat Anda mengangkatnya ke ori. Jaga tulang kering Anda tetap bersama dan punggung lurus, dan sesuaikan seperlunya untuk menjaga keseimbangan Anda. Tahan selama tiga hingga lima napas, lalu lepaskan. Gagak



Berjongkok di atas matras, dengan telapak tangan Anda di atas oor. Posisikan siku Anda di bagian dalam setiap lutut yang sesuai, lalu condongkan tubuh ke depan hingga pusat gravitasi Anda bergeser ke depan, dan kaki Anda meninggalkan tanah. Seimbangkan lengan Anda selama mungkin, menjaga pernapasan Anda tetap stabil. Roda



Berbaringlah di atas matras telentang. Angkat lutut Anda sampai kaki Anda berada di bawah pinggul Anda. Pada saat yang sama, letakkan telapak tangan Anda di tanah di dekat telinga Anda. Saat Anda menarik napas, tarik dengan kuat dengan kaki Anda dan dorong dengan kuat dengan tangan Anda sehingga tubuh Anda melengkung ke atas matras. Tahan pose selama tiga hingga lima napas, lalu lepaskan, turunkan tubuh Anda perlahan untuk beristirahat di oor.

Adjusting the Difficulty

Perhatikan bagaimana pose ini biasanya berakhir dengan ditahan selama tiga hingga delapan pernapasan? Itu disengaja. Pernapasan adalah apa yang membuat yoga unik, jadi itu adalah bagian penting dari setiap pose. Tiga hingga delapan pernapasan adalah awal yang baik sebagai pemula. Praktisi asana tingkat lanjut tidak selalu melakukan pose yang lebih rumit, atau jumlah pose yang lebih besar: mereka melakukan pose lebih lambat, dengan kontrol lebih besar, dan menahan setiap pose lebih lama. Saat saya menjadi lebih baik dengan latihan, saya akan menahan setiap pose untuk beberapa napas lagi.

Shavasana



Shavasana adalah, dalam tradisi yoga, tujuan akhir dari semua pembengkokan dan pernapasan ini. Ingat, tujuan awal asana adalah untuk mempersiapkan meditasi. Tradisi itu terus berlanjut ke dalam praktik modern, dan untuk alasan yang bagus: itu cukup bagus. Shavasana adalah pelepasan lengkap dari semua ketegangan otot yang dikombinasikan dengan meditasi mendalam. Ini sering disebut sebagai pose Mayat, tetapi jangan biarkan label yang tidak menyenangkan itu membuat Anda goyah. Posisinya sendiri sederhana: berbaring di atas matras, kaki terentang lurus, lengan di samping Anda. Tutup mata Anda dan lepaskan semua ketegangan dari tubuh Anda. Otot Anda seharusnya tidak bekerja sama sekali. Pindai tubuh Anda dari ujung kepala sampai ujung kaki, dan di mana pun Anda melihat otot berkontraksi, lepaskan. Ini termasuk kelompok otot yang tidak biasa kita perhatikan, seperti leher dan bahkan lidah Anda. Pada titik ini, Anda juga berhenti mempertahankan ujjayi, dan mulai bernapas secara normal. Sensasi terasa seperti Anda melebur ke dalam lantai. Setelah sekian lama menjaga otot-otot Anda bergerak terus-menerus, shavasana adalah

relief. Melelahkan tubuh Anda tampaknya memungkinkan Anda untuk merilekskan pikiran Anda, sehingga lebih mudah untuk bermeditasi ketika Anda mencapai shavasana. Tetap di shavasana selama tiga menit atau lebih, lalu bangun. Asanapractice Anda selesai.

Meninjau Metode

Pada titik ini, saya telah mencapai ambang batas kinerja target saya. Setelah total tiga jam pengajaran, saya mampu berlatih di rumah dengan cara yang aman dan efisien. Mari kita tinjau inti dari metode yang saya gunakan untuk belajar yoga:

Saya memperoleh peralatan yang diperlukan: tikar, tali, dan balok. Saya menghabiskan beberapa jam dengan instruktur berpengalaman untuk mempelajari dasar-dasar dan memperbaiki kesalahpahaman utama. Saya belajar bahwa latihan asana yoga melibatkan bergerak melalui pose sambil fokus pada napas dan mempertahankan kondisi mental yang penuh perhatian. Saya mempelajari urutan pose vinyasa dasar (Salam Matahari), sepuluh pose umum, dan sepuluh pose duduk / berbaring umum. Saya mempelajari teknik pernapasan dasar (pranayama) yang membuatnya lebih mudah untuk tetap fokus pada napas saat berlatih. Saya belajar mencegah cedera dengan memperhatikan tubuh saya, memodifikasi pose seperlunya, menghindari ekstensi berlebih, dan menghindari beban yang tidak perlu pada tulang belakang serviks. Saya berlatih sampai saya dapat menyelesaikan urutan dari ingatan, yang memakan waktu sekitar dua puluh puluh menit.

Ke Mana Saya Pergi Dari Sini

Saat ini, saya senang dengan praktik asana saya. Setelah tiga jam penelitian dan pengajaran, saya dapat berlatih yoga kapan pun saya mau atau perlu, dan saya tidak perlu bergantung pada pergi ke kelas terstruktur di studio. Dua puluh menit di pagi atau sore hari, dan saya sudah siap. Saya suka yoga menggabungkan olahraga dengan meditasi. Setelah saya berlatih, terlihat betapa jauh lebih baik tubuh saya terasa, serta

bagaimana perasaan saya secara emosional. Terus berlatih akan memastikan saya menuai manfaat dari meditasi dan latihan rentang gerak. Itulah triknya: Saya perlu berlatih secara teratur untuk menuai ke sana. Menyisihkan waktu di pagi dan sore hari untuk berlatih adalah kuncinya. Alih-alih memulai hari dengan melompat online untuk memeriksa email saya atau langsung bekerja, saya mengubah jadwal saya sehingga saya dapat berlatih sebelum melakukan hal lain. Merawat tubuh saya perlu menjadi prioritas pertama saya, bukan yang terakhir. Yoga akan membantu menjaga saya dalam kondisi baik selama beberapa dekade mendatang. Saya tidak harus menjadi biksu atau kontorsionis untuk menikmati manfaat yoga. Saya tidak harus mengetahui setiap pose, atau bisa menempelkan kaki saya di belakang kepala saya. Saya hanya perlu mempraktikkan dasar-dasarnya, yang sama sekali tidak sulit untuk dipelajari. Lebih baik lagi: Saya belajar apa yang ingin saya pelajari hanya dalam tiga jam, yang jauh lebih baik dari yang saya perkirakan. Saya memulai proyek ini dengan harapan untuk mempelajari serangkaian panjang postur yang rumit tetapi, ternyata, itu tidak diperlukan sama sekali. Saya dapat mencapai apa yang ingin saya capai dengan menyederhanakan, yang memungkinkan saya mendapatkan hasil yang saya inginkan dalam waktu yang lebih singkat. Ketika datang untuk mempelajari sesuatu yang baru, tidak ada gunanya membuatnya lebih sulit daripada yang sebenarnya seharusnya.



5Pemrograma

n

Pelajaran: Hal-hal yang kompleks menjadi sederhana ketika Anda memecahnya

Jika debugging adalah proses menghapus bug perangkat lunak, maka pemrograman harus menjadi proses memasukkannya.—EDSGER DIJKSTRA, ILMUWAN KOMPUTER TERKENAL

• • •

Untuk gambar, video, dan komentar tambahan tentang bab ini, visit<http://first20hours.com/programming>.

Saya telah mencari nafkah di web sejak tahun 2007, ketika saya berhenti dari pekerjaan penuh waktu saya sebagai manajer pemasaran di Procter & Gamble demi memulai perusahaan penerbitan dan konsultasi saya sendiri. Situs web utama saya, PersonalMBA.com, adalah mata pencaharian saya: Saya secara menegaknya seorang profesor bisnis, tetapi saya tidak bekerja di universitas. Setiap tahun, saya memperbarui daftar buku bisnis terbaik yang tersedia untuk pembaca yang ingin mengajari diri mereka sendiri dasar-dasar bisnis.¹ Daftar bacaan yang direkomendasikan MBA Pribadi telah menjadi favorit pembaca abadi sejak edisi pertama diterbitkan pada tahun 2005, dan pembaruan daftar tersebut disambut dengan gelombang pasang pengunjung dari seluruh dunia. Sejak 2005, PersonalMBA.com telah dikunjungi oleh lebih dari 2 juta pembaca. Buku pertama saya, *The Personal MBA: Master the Art of Business* (2010), adalah perpanjangan alami dari PersonalMBA.com, dan kemudian menjadi buku terlaris internasional. Bagian dari proses penulisan

buku sedang mengungkapkan cara menyebarkan berita, jadi selama bertahun-tahun saya telah bekerja keras untuk menarik pembaca baru. Akibatnya, The Personal MBA telah ditampilkan di New YorkTimes, Wall Street Journal, Fortune, Forbes, dan FastCompany, dan di banyak situs web dan blog populer. Setiap kali buku atau situs web saya ditampilkan, ribuan pembaca mengunjungi PersonalMBA.com dalam waktu yang sangat singkat.

Harga Kemajuan

Memiliki ribuan pengunjung simultan ke situs web Anda adalah hal yang hebat, asalkan pengunjung tersebut benar-benar dapat mencapai situs. Itu adalah masalah saya: setiap kali situs web saya menerima jumlah trac yang signifikan, itu akan jatuh dalam kobaran kemuliaan, meninggalkan pengunjung hanya dengan pesan kesalahan samar. Berikut adalah contoh khas: Lifestacker.com, blog produktivitas populer, telah menampilkan The Personal MBA tiga kali dalam tujuh tahun terakhir. Setiap kali, ribuan orang akan mencoba visitPersonalMBA.com pada saat yang sama, membanjiri server web saya (komputer yang mengirimkan halaman web ketika diminta oleh pengunjung). Alih-alih mengirimkan informasi yang diminta, server saya akan mengembalikan pesan "Kesalahan membangun koneksi database" atau "Error503", setara digital dari server yang memohon belas kasihan. Setiap kali server saya jatuh di bawah beban berat, sepotong kecil jiwa saya mati bersamanya. Semua waktu yang dihabiskan untuk memasarkan situs web saya terbuang-. Ribuan pembaca yang penasaran tidak tertarik untuk mempelajari apa yang harus saya lakukan, tetapi dengan server saya turun, mereka pergi dengan kecewa dan tangan kosong. Pemasaran saya bekerja dengan sangat baik, dan sistem saya tidak dapat mengatasi permintaan yang tiba-tiba.

Memeriksa Masalah

Pada rst, pendekatan saya adalah memperkuat server dengan menambahkan lebih banyak daya pemrosesan dan lebih banyak memori. Itu membantu, tetapi hanya untuk menunjuk. Di luar titik itu, situs saya akan mogok dan terbakar, yang kebetulan bertepatan dengan hampir setiap kemenangan pemasaran besar. Pada saat itu, PersonalMBA.com berjalan di sistem manajemen situs web populer yang disebut WordPress.² WordPress dioptimalkan untuk memudahkan instalasi dan penggunaan, bukan kinerja di bawah beban berat. Di bawah kongurasi WordPress default, setiap permintaan halaman web menendang kaskade aktivitas server, menghasilkan ratusan permintaan tersembunyi untuk mengirimkan setiap halaman web ke pembaca. Itu membuat setiap permintaan halaman web individu "mahal". Artinya, setiap permintaan membutuhkan jumlah memori dan daya pemrosesan yang signifikan untuk menyelesaikannya. Jika satu pengunjung melihat ve halaman di situs web, pengguna tersebut menghasilkan permintaan yang sangat mahal. Jika seribu pengunjung meminta halaman yang sama persis pada waktu yang sama, server akan mencoba untuk menendang seribu proses mahal yang sama sekaligus.

Sistem Down

Dalam situasi ini, server yang terkepung yang buruk akan mencoba mengembalikan setiap permintaan, tetapi karena setiap permintaan membutuhkan banyak sumber daya, server akan kehabisan memori sebelum setiap permintaan dipenuhi. Pada saat itu, server menaikkan keserahan putih, dan pengunjung kurang beruntung. Untuk mengatasi masalah ini, saya beralih perusahaan hosting web berkali-kali dan menghabiskan ratusan jam belajar bagaimana menggabungkan server WordPress agar tetap online di bawah beban berat. Setiap kongurasi server baru meningkat dalam kompleksitas, dan setiap instalasi baru membutuhkan lebih banyak pemeliharaan berkelanjutan.

Akhirnya saya membangun kongurasi server khusus saya sendiri, memanggil serangkaian panjang perintah sistem misterius untuk menginstal, mengatur, dan memodifikasi aplikasi server kompleks yang hampir tidak saya pahami. Setiap kesalahan atau masalah yang saya alami dalam pengaturan atau pemeliharaan berubah menjadi penelitian dan pemecahan masalah selama berjam-jam. Jika itu belum cukup, popularitas dan relatif ketidakamanan WordPress menjadikannya target populer bagi peretas dan spammer. Setiap minggu atau lebih, seorang programmer yang tuduh menemukan beberapa kerentanan baru, kemudian menggunakannya untuk mengambil alih akun pengguna atau menghapus jutaan arsip blog WordPress dengan spam. Mengamankan instalasi WordPress dan menjaga pembaruan perangkat lunak tetap teratas bisa menjadi pekerjaan penuh waktu tersendiri, terutama jika Anda memelihara banyak situs web. (Saya memelihara dua belas situs web pada saat itu dan gagal total.) Pada titik tertentu, saya menyadari bahwa saya menghabiskan lebih banyak waktu untuk menyimpan situs saya secara online daripada saya meneliti dan menulis untuk pembaca saya. Itu tidak masuk akal. Saya tidak hanya membuang-buang kapasitas produktif, saya tidak benar-benar belajar cara memprogram. Saya mempelajari banyak peretasan dan solusi khusus situasi yang hanya berlaku untuk menjalankan WordPress. Tidak keren.³ Saya memutuskan untuk mencari cara lain untuk memelihara situs web saya, dan tidak butuh waktu lama untuk menemukan alternatif yang menjanjikan.

Solusi Potensial

Suatu hari, saya menemukan esai tentang Jekyll,⁴ sebuah program manajemen situs web yang dibuat oleh Tom Preston-Werner, yang paling dikenal sebagai pendiri repositori kode sumber terbuka GitHub.⁵ Jekyll dirancang untuk menggantikan sistem seperti WordPress dengan memudahkan untuk menjalankan situs web yang tidak bergantung pada permintaan mahal. Bayangkan Anda memiliki ratusan dokumen pengolah kata yang berisi informasi penting dan Anda perlu membuat semuanya terlihat sama—font yang sama, gaya judul yang sama, dan sebagainya. Jika Anda menulis

program yang dapat menerapkan desain halaman tertentu yang Anda pilih untuk setiap orang secara otomatis (versus memperbarui setiap le dengan tangan), itu akan menghemat banyak waktu Anda. Itulah yang pada dasarnya dilakukan Jekyll untuk halaman web. Jalankan satu perintah dan Jekyll menghasilkan situs web lengkap menggunakan les di komputer Anda yang berisi informasi situs web dan template desain Anda. Jika Anda membuat perubahan pada desain atau konten halaman, Anda cukup menjalankan Jekyll lagi, dan seluruh situs akan diperbarui dengan perubahan secara otomatis, menghemat waktu berjam-jam. Jekyll menghadirkan peluang yang menjanjikan. Secara teori, saya bisa mengganti WordPress dengan folder teks sederhana di komputer saya. Situs web saya akan sangat cepat, sangat stabil, dan saya akan menghemat diri saya lebih dari seratus jam pemeliharaan server setiap tahun. Namun, ada tangkapan: Jekyll ditulis dalam Ruby, bahasa pemrograman yang saya tidak tahu. Saya tidak tahu cara menulis kode di Ruby atau menjalankan aplikasi Ruby yang melayani pengguna sungguhan. Beberapa hal yang perlu saya lakukan untuk menjalankan PersonalMBA.com membutuhkan lebih dari sekadar pemformatan sederhana. Untuk menjalankan situs web saya menggunakan Jekyll, saya harus belajar cara memprogram dan menerapkan aplikasi web Ruby. Sepertinya saya telah menemukan "masalah yang menyenangkan".

Belajar Kode

Saya sudah cukup lama ingin belajar cara memprogram, tetapi proyek lain selalu diprioritaskan. Jika saya dapat belajar kode, penerbitan dan peluang bisnis saya akan berkembang pesat, karena semua yang saya lakukan untuk membangun bisnis saya berjalan di web. Penting untuk dicatat bahwa semua yang telah saya lakukan hingga saat ini bukanlah pemrograman. HTML dan CSS, bahasa yang saya gunakan untuk membangun halaman web, disebut bahasa "markup". Kode HTML dan CSS tidak pintar dengan cara apa pun: itu hanya memberi tahu komputer untuk menunjukkan kepada pengguna

teks le dengan cara khusus (yaitu, "buat teks ini tebal" atau "bagian ini adalah judul dengan ukuran font 24 poin"). Hal yang sama berlaku untuk pengaturan server gila saya. Meskipun saya sedang menyusun perangkat lunak, saya tidak benar-benar memprogram. Sebagai gantinya, saya menginstal program yang telah ditulis sebelumnya, lalu mengubah beberapa pengaturan. Pemrogram menulis aplikasi yang saya gunakan, tetapi saya tidak memerlukan pengetahuan pemrograman untuk menggunakannya. Kongurasi dan administrasi server adalah keterampilan yang berguna, tetapi bukan pemrograman.

Jadi apa itu "pemrograman"?

Pemikiran pertama saya adalah bahwa pemrograman menyuruh komputer untuk melakukan sesuatu, tetapi itu tidak terlalu spesifik atau membantu. Lakukan apa? "Benda" apa? Sepuluh tahun yang lalu, saya mengambil dua kursus pemrograman dasar di perguruan tinggi, jadi tidak dapat menemukan penyangkalan yang berguna adalah hal yang memalukan, terus terang. Saya ingat beberapa istilah dasar, seperti variabel, loop, input, output, fungsi, orientasi objek, dan sesuatu yang disebut abubble sort, tetapi tidak banyak lagi. Tugas kursus membutuhkan pembelajaran bahasa pemrograman yang disebut C++, dan saya ingat frustrasi ketika butuh waktu berjam-jam untuk menemukan titik koma yang hilang yang terus merusak program saya. Saya ingat profesor saya mengatakan hal-hal seperti "Anda tidak akan pernah menggunakan bubblesort untuk aplikasi nyata, tetapi kami akan tetap mempelajarinya." Saya menulis beberapa program dasar untuk kursus ini karena saya harus: tujuannya adalah untuk lulus kelas dengan nilai bagus, dan saya melakukannya. Sayangnya, program yang kami tulis tidak memiliki kegunaan nyata di luar kelas, dan saya sudah lama tidak menggunakan konsep ini. Meskipun saya ingat beberapa kata, saya harus memulai dari awal dalam hal memahami ide-ide kunci. Karena saya tidak dapat secara spesifik menyangkal apa yang saya coba lakukan ketika saya mengatakan saya ingin "belajar pemrograman", saya akan memiliki

kesulitan menentukan tingkat kinerja target yang berguna. "Buat program komputer" sedikit lebih spesifik, tetapi masih tidak terlalu berguna. Saatnya membongkar apa yang saya ketahui saat ini tentang pemrograman:

Saya tahu programmer "menulis" program, yang menunjukkan bahwa itu adalah latihan kreatif yang dapat dilakukan dengan banyak cara. Program sering disebut "aplikasi", dan kata-kata tersebut digunakan secara bergantian. Ketika program komputer "dijalankan" atau "dieksekusi", mereka melakukan hal yang harus mereka lakukan, apa pun yang terjadi. "Input" dan "output" mudah diingat, karena digunakan secara umum. Input adalah informasi atau data yang digunakan program, dan output adalah apa yang Anda dapatkan ketika program selesai berjalan. "Variabel" pada dasarnya adalah placeholder untuk sesuatu yang berubah. Anda dapat membuat variabel sebanyak yang Anda inginkan, dan membuat variabel tersebut mewakili apa pun yang Anda inginkan. "Program" itu sendiri pada dasarnya adalah seperangkat instruksi dan aturan terperinci yang memberi tahu komputer dengan tepat apa yang harus dilakukan untuk (atau dengan) input. Ketika program selesai berjalan, itu memberi Anda output. Program "mogok" atau menampilkan pesan kesalahan ketika terjadi kesalahan atau komputer tidak dapat mengetahui apa yang harus dilakukan selanjutnya.

Sekarang kita sampai ke suatu tempat. Ini adalah dekonstruksi yang sangat mendasar: alih-alih "pemrograman", kita sekarang memiliki tiga subkonsep untuk dikerjakan:

Input—informasi yang Anda gunakan untuk menjalankan proses. Proses—serangkaian langkah yang diambil program, berdasarkan inputnya. Output—hasil akhir dari program.

Rincian ini jauh lebih berguna. "Menulis program komputer" berarti mengubah informasi apa yang Anda mulai, menggambarkan serangkaian langkah yang menjelaskan dengan tepat apa yang akan dilakukan komputer dengan input itu, dan mengganti output yang akan dikembalikan komputer ketika program tidak berjalan. Pikirkan tentang grafik burung hantu, yang tampaknya menjadi kait mental yang berguna untuk cara kerja program. Anda memulai proses dengan input tertentu. Sepanjang jalan, Anda mengambil tindakan tertentu ketika kondisi spesifik benar atau salah. Proses berakhir saat Anda mencapai akhir

owchart, dan Anda tersisa dengan output: hasil akhir dari proses lengkap yang dijelaskan oleh Owchart. Membuat program komputer tampaknya merupakan cara yang berbeda untuk melakukan pemikiran yang sama seperti yang Anda lakukan ketika Anda membuat aowchart. Anda mengajukan jenis pertanyaan yang sama:

Apa yang saya mulai? Apa yang terjadi di awal proses? Apa yang terjadi setelah itu? Kemudian? Kapan prosesnya berakhir? Apa yang saya miliki ketika proses selesai?

Diagram alur menggambarkan jawaban atas pertanyaan-pertanyaan ini dalam format visual, dan program menggambarkannya menggunakan teks, tetapi proses pemikirannya sama. Analogi owchart juga berguna karena memberikan beberapa petunjuk tentang konsep lain yang mungkin penting. Kondisional adalah pernyataan seperti:

"Jika X benar/salah maka lakukan Y""Jika X adalah/bukan Y maka lakukan Z""Ketika X benar/salah maka lakukan Y""Ketika X adalah Y maka lakukan Z""Sementara X benar/salah maka lakukan Y""Sementara X adalah Y maka lakukan Z"

X, Y, dan Z dalam hal ini adalah variabel, yang dapat mewakili apa saja. Variabel mungkin mewakili angka, seperti dalam aljabar dasar, atau mereka mungkin singkatan dari kata-kata. Terkadang variabel adalah huruf tunggal atau simbol, dan terkadang adalah kata. Apa pun itu, mereka mewakili apa pun yang kami kerjakan. Kondisional (bagian IF, THEN, WHEN, dan WHILE) seperti panah dengan pertanyaan pada bagan owchart. Pikirkan mengendarai mobil: JIKA lampu thetrac berwarna merah, MAKA berhenti. KETIKA lampu trac berubah menjadi hijau, LALU pergi. KETIKA lampu trac berwarna kuning, pelan-pelan, dan bersiaplah untuk berhenti.

Ada baiknya menggali pernyataan bersyarat ini sedikit lebih banyak, karena ada beberapa pola umum. Benar/salah muncul banyak, dan WHILE tampaknya menyiratkan kita terus melakukan sesuatu alih-alih mengambil satu tindakan. Dalam semua kasus, kondisional berisi beberapa pernyataan yang menolak apakah akan mengambil tindakan atau tidak. Pernyataan ini disebut kondisi, dan dapat mengambil berbagai bentuk. Terkadang kondisinya adalah perbandingan dasar benar/salah (Apakah merah muda?), terkadang perbandingan matematis (Apakah X lebih besar dari 100?), dan terkadang berisi logika (Apakah lampu lalu TIDAK merah?). Tujuan dari kondisi ini adalah untuk menentukan apakah proses terkait harus terjadi atau tidak. Jika kondisinya benar atau valid, program memproses instruksi. Jika tidak, instruksi terkait dilewati, dan program beralih ke instruksi berikutnya. Variabel benar/salah disebut variabel Boolean, yang merupakan frasa fantastis untuk sesuatu yang hanya memiliki dua opsi. Ya/tidak dan on/off juga variabel Boolean. Variabel Boolean cukup penting dalam program komputer, karena mereka adalah unit dasar dari proses sederhana (seperti benar/salah dalam grafik owchart) dan on/off pipping sakelar elektronik kecil di inti komputer. WHILE, dalam hal ini, adalah jenis kondisional khusus yang disebut loop. Loop menyebabkan proses yang dimaksud berulang sampai kondisi terpenuhi. Kembali ke analogi mengemudi: SAAT lampu trac menyala, jangan bergerak. Cukup mudah, bukan? Jika kita berpikir untuk menulis program komputer seperti menggambar bagan burung hantu, proses dasarnya lebih mudah untuk divisualisasikan. Ada satu kerutan terakhir: bagaimana jika komputer macet, tidak tahu apa yang harus dilakukan, atau instruksi yang dicoba komputer pada suatu titik tidak berfungsi atau masuk akal? Apa yang terjadi? Program "mogok"—program berhenti sepenuhnya, dan sering menghasilkan pesan kesalahan, bukan hasil yang diinginkan. Kita semua pernah mengalami kesalahan "layar biru kematian" atau "Kesalahan 404: Halaman Web Tidak Ditemukan" Windows yang ditakuti saat menjelajahi web.

Sesuatu yang tidak terduga terjadi, komputer panik, dan program mogok. Sebagai pemrogram komputer, tugas Anda adalah mencegah terjadinya crash dan kesalahan. Cara terbaik untuk melakukannya adalah dengan memastikan program selalu memiliki informasi yang dibutuhkan untuk menyelesaikannya proses seperti yang direncanakan, tetapi itu tidak selalu memungkinkan. Dalam kasus di mana ketidakpastian tidak dapat dihindari, akan berguna untuk memiliki cara untuk memulihkan ketika program gagal saat mencoba menyelesaikan proses. Pernyataan pemulihan kesalahan ini disebut pengecualian, dan sangat berguna. Anda dapat menganggapnya sebagai kondisional khusus kesalahan: JIKA program akan mogok dengan cara X, KEMUDIAN lakukan Y alih-alih crash. Pengecualian sedikit seperti menambahkan generator cadangan ke gedung rumah sakit. Sebagian besar waktu, generator hanya duduk di sana tanpa melakukan apa-apa. Namun, jika listrik padam, generator akan pulih, dan rumah sakit menggunakan daya dari generator alih-alih menjadi gelap sepenuhnya. Itu hal yang sangat baik jika ada pasien di rumah sakit dengan mesin pendukung kehidupan yang membutuhkan listrik. Kegagalan lengkap berbahaya, jadi rencana pencadangan sangat penting. Itu pemrograman dasar, sungguh. Masukan dening. Pengaturan variabel. Membuat proses yang mengarah pada output yang diinginkan. Memikirkan proses-proses tersebut seperti grafik owchart, menambahkan kondisional dan pengecualian seperlunya. Jika semuanya berjalan dengan baik, Anda memasok input, menjalankan program, dan mendapatkan output yang diinginkan. Ini adalah implikasi berlebihan besar-besaran dari aktivitas yang sangat kompleks, tetapi cukup rinci untuk berguna bagi seseorang yang baru memprogram. Dengan mendekonstruksi pemrograman dengan cara ini, lebih mudah untuk mengetahui dari mana harus memulai.

Bahasa Pemrograman

Inilah bagian yang sulit: komputer tidak berbicara bahasa dengan cara yang sama seperti manusia. Pada intinya, komputer bekerja dengan ipping kecil

Sakelar elektronik menyala dan O dengan cara yang sangat spesifik. Kecuali komputer memiliki beberapa cara untuk menerjemahkan perintah bahasa manusia kita ke dalam switching elektronik, komputer tidak akan dapat melakukan apa yang kita perintahkan. Itulah yang dilakukan bahasa pemrograman: mereka memberi pemrogram manusia cara khusus untuk memberi tahu komputer kapan harus memulai, apa yang harus dilakukan, dan kapan harus berhenti. Mereka juga memungkinkan programmer untuk menyangkal seperti apa input, proses, dan output, dan apa yang harus dilakukan ketika program selesai berjalan. Setiap bahasa pemrograman memiliki cara menulis perintah tertentu, yang disebut sintaks. Sintaks bahasa berisi aturan yang digunakan komputer untuk menerjemahkan perintah program ke dalam switch-ip elektronik kecil. Hampir setiap bahasa memiliki cara untuk mengurangi variabel, kondisional, dan pengecualian. Detailnya bervariasi, tetapi konsep intinya sama.

Berpikir Seperti Seorang Programmer

Pemrogram sering memikirkan masalah dalam apa yang disebut kode semu: bahasa yang terlihat atau terdengar seperti kode, tetapi tidak cukup spesifik untuk benar-benar dieksekusi oleh komputer. Anggap saja sebagai bentuk sketsa. Pseudocode membantu Anda memikirkan proses pemecahan masalah. Contoh mengemudi yang saya gunakan adalah contoh pseudocode. Jika saya masuk ke mobil saya dan berkata, "JIKA saya memutar kunci, LALU nyalakan mesin" dengan keras, tidak akan terjadi apa-apa. Itu tidak berarti pernyataan itu tidak berguna. Pernyataan tersebut hanyalah cara memikirkan langkah-langkah yang diperlukan untuk memecahkan masalah atau mendapatkan hasil yang diinginkan. Anda dapat menggunakan konsep pemrograman dasar yang baru saja kita bahas untuk membuat sketsa beberapa program sederhana untuk tugas-tugas umum. Inilah cara yang menyenangkan untuk mencobanya sendiri. Carilah teman, dan minta mereka untuk membantu Anda menyelesaikan tugas sederhana seperti "membuat sandwich".

Satu-satunya aturan adalah bahwa teman Anda hanya dapat melakukan persis apa yang Anda katakan kepadanya, tidak lebih, tidak kurang. Dia tidak diizinkan untuk mengasumsikan pengetahuan tentang apa pun, dan harus mengambil setiap permintaan secara harfiah.

Dalam hitungan menit, Anda akan melakukan percakapan seperti ini:

Anda: Ambil roti. Teman: Saya tidak mengerti "mengambil". Anda: Gerakkan tangan Anda ke tempat yang saya katakan dan genggam. Teman: Saya tidak mengerti "tangan." Anda: (Menghela nafas) Benda ini di sini. [Anda menunjuk ke tangan teman.]Teman: Mengerti.Anda: Gerakkan tangan Anda ke tempat yang saya katakan dan genggam. Teman: Saya tidak mengerti "menggenggam". Anda: Lenturkan nger Anda seperti ini. [Anda mendemonstrasikan, exing, dan melepaskan nger Anda.]Teman: Mengerti.Anda: Gerakkan tangan Anda ke roti dan peganglah. Teman: Saya tidak mengerti "roti". Anda: Benda ini di sini! [Anda menunjuk ke roti.]Teman: Mengerti.Anda: Gerakkan tangan Anda ke roti dan peganglah. Teman: [Bergerak dengan tangan ke roti, mantan nger, dan melepaskan. Roti tidak bergerak.]Anda: INI ADALAH PERMAINAN BODOH!!

Ini adalah contoh konyol, tetapi sangat dekat dengan seperti apa pemrograman terlihat dan terasa, terutama di awal. Komputer, seperti teman Anda, tidak dapat memahami apa pun yang tidak Anda tolak secara eksplisit. Setiap proses kompleks yang Anda coba dene perlu ditentukan secara lengkap dan tidak ambigu. Itulah yang membuat pemrograman dicult: satu perintah ambiguitas orstray dapat menyebabkan seluruh program gagal. Pemrograman adalah kerajinan yang tak kenal ampun dalam hal ini: kode Anda benar atau salah, dan harus diungkapkan dengan detail yang tepat. Komputer tidak terkesan dengan pesona atau kecerdasan. Jika kode Anda tidak lengkap atau tidak terbentuk, aplikasi Anda akan mogok, Anda akan kehilangan data, dan/atau Anda akan membuat bug: sedikit kode nakal yang menghasilkan konsekuensi yang tidak terduga atau tidak terduga. Seperti matematika, logika berfungsi atau tidak. Tidak ada A untuk eort.

Konon, dalam pemrograman tidak ada solusi universal tunggal untuk masalah yang diberikan, seperti yang ada dalam matematika. Ada ribuan cara untuk menghasilkan output yang Anda inginkan mengingat input yang ditentukan. Sebagai pemrogram, Anda dapat memilih pendekatan Anda berdasarkan alat yang Anda miliki. Setelah Anda terbiasa dengan fakta bahwa komputer tidak dapat membaca pikiran Anda, Anda belajar untuk mulai memberikan perintah yang dapat dipahami sistem, dengan cara yang sama Anda belajar untuk secara eksplisit menyangkal istilah-istilah dan mendemonstrasikan tindakan dasar kepada teman Anda di "SandwichGame" sebelum memberikan perintah yang rumit.

Apa yang Membuat Pemrograman Aplikasi Web Berbeda? Kami sekarang memiliki penolakan pemrograman yang berfungsi. Ini sangat sederhana, tetapi memberikan pemahaman yang cukup baik tentang apa yang kami coba lakukan untuk saat ini. Saya tidak tertarik dengan setiap jenis pemrograman: Saya ingin menulis program yang berjalan di situs web. Jika Anda pernah menggunakan program email berbasis web seperti Gmail, Hotmail, Yahoo Mail, atau sejenisnya, Anda tahu apa yang saya bicarakan. Program-program ini berjalan di browser Internet Anda. Anda tidak perlu mengunduh program perangkat lunak ke komputer Anda untuk menggunakannya. Anda cukup mengarahkan browser Anda ke situs web, masuk, dan Anda siap untuk pergi. Perbedaan antara perangkat lunak yang berjalan secara lokal dan perangkat lunak yang berjalan dari jarak jauh di server web ini adalah hal yang penting. Untuk mengembangkan aplikasi web, Anda harus menulis aplikasi dan kemudian mengujinya untuk melihat apakah itu berfungsi. Semua pengembangan dan pengujian terjadi di komputer Anda sendiri. Setelah program berfungsi, Anda kemudian mengirimkannya ke server web "produksi" sehingga orang lain dapat mengaksesnya. Orang-orang di Internet tidak dapat masuk ke komputer pribadi Anda dari web, jadi mengunggah perangkat lunak ke server web yang dapat diakses publik diperlukan bagi orang lain untuk menggunakan program Anda.

Itu berarti proses pengembangan memiliki dua fase utama: pemrograman dan pengujian lokal, dan mendorong program yang diluncurkan ke server produksi jarak jauh untuk penggunaan aktual. Saya harus menjelaskan bagaimana kedua fase ini bekerja. Inilah hal lain yang saya ketahui dari pengalaman saya sebelumnya dengan HTML dan CSS: mereka "bodoh", karena Anda tidak dapat meminta halaman web dasar untuk menyimpan informasi apa pun. Katakanlah Anda memiliki halaman web le yang mengatakan, "Halo, Dunia!" 6 dan Anda ingin mengganti "Dunia" dengan nama orang yang mengunjungi halaman. Ini adalah pemikiran yang bagus, tetapi halaman web dasar tidak memiliki cara untuk menyimpan informasi untuk diingat nanti. Mereka hanya menampilkan teks di thele, dan le tidak diizinkan untuk memperbarui dirinya sendiri. Istilah teknis untuk ini adalah state. Halaman web dasar yang dibuat menggunakan HTML dan CSS tidak memiliki status, sehingga disebut sebagai sumber daya "tanpa kewarganegaraan". Anda dapat menambahkan formulir kecil ke halaman web yang meminta nama pengguna dengan tombol Simpan di sebelahnya, tetapi tombol tersebut tidak akan benar-benar menyimpan informasi apa pun kecuali Anda membuat tempat terpisah untuk informasi tersebut. Itu sebabnya aplikasi web menggunakan dua pendekatan umum untuk menyimpan data untuk nanti: database dan cookie. Cara terbaik untuk memahami apa yang dilakukan database adalah dengan membayangkan setumpuk kartu indeks. Katakanlah Anda mencoba membuat buku alamat, dan Anda ingin mencatat nama, nomor telepon, alamat email, jenis kelamin, dan usia masing-masing teman Anda. Setiap teman mendapatkan kartu indeks mereka sendiri, dan informasi mereka dicatat di kartu itu. Jika salah satu teman Anda mengubah alamat emailnya, misalnya, Anda dapat menghapus informasi lama dan memperbarui kartu dengan informasi baru. Saat Anda melihat kartu teman tertentu, Anda dapat melihat semua informasi mereka sekaligus. Anda dapat menganggap seluruh tumpukan kartu indeks sebagai database. Setiap kartu dalam tumpukan disebut catatan. Anda dapat memiliki catatan sebanyak yang Anda suka dalam database, tetapi pada titik tertentu, tumpukan

menjadi sulit untuk dikelola. Seringkali masuk akal untuk membagi tumpukan menjadi beberapa subtumpukan yang lebih kecil: teman dan keluarga dalam satu tumpukan, rekan kerja dari tempat lain, misalnya. Jelas sejauh ini? Di sinilah menjadi keren: bayangkan tumpukan kartu indeks Anda sekarang menjadi tumpukan kartu indeks ajaib. Anda dapat berbicara detumpukan sihir dan memerintahkannya untuk menunjukkan kepada Anda kartu yang tidak memiliki kondisi tertentu, seperti

"Tunjukkan padaku kartu John Smith." Tunjukkan padaku semua kartu di mana jenis kelamin adalah perempuan." Tunjukkan padaku semua kartu di mana usia lebih besar dari lima puluh."

Cukup berguna, bukan? Intinya, itulah yang dilakukan database: mereka memberi Anda cara untuk menyimpan informasi terstruktur, serta cara untuk mengambil informasi itu sesuka Anda. Setiap potongan data yang kita masukkan ke kartu disebut bidang. Semakin banyak orang yang Anda miliki dalam database Anda, semakin banyak cara Anda dapat mengambil data saat Anda menginginkannya. Database adalah cara paling umum untuk menyimpan data dalam aplikasi web. Jika Anda ingin menyimpan informasi seperti nama pengguna, alamat email, dan informasi lainnya, menggunakan database adalah hal yang wajar. Setelah John Smith masuk ke aplikasi Anda, Anda dapat meminta aplikasi Anda mengambil nama dari catatan database John, lalu menampilkan "Halo, John Smith!" Cara umum lainnya untuk menyimpan informasi dalam aplikasi web adalah dengan menggunakan cookie: teks yang sangat kecil yang disimpan di komputer pengguna. Cookie berguna untuk menyimpan sejumlah kecil data yang tidak perlu ada dalam waktu lama. Dalam kasus program buku alamat kami, akan tepat untuk menyimpan cookie saat John Smith masuk. Cookie John le akan berisi informasi seperti nama pengguna = johnsmith dan loggedin = true. Jika Smith meninggalkan aplikasi, tetapi kembali nanti, aplikasi akan mengenali cookie dan memberinya akses tanpa memerlukan login lain. Cookie dapat diatur untuk kedaluwarsa dalam jangka waktu tertentu,

yang membuatnya berguna untuk pemrograman semacam ini. (Jika Anda pernah melihat fitur "Ingat kata sandi ini" di situs web, inilah cara kerjanya.) Apa yang kami lakukan di sini adalah dekonstruksi dasar. Ini bukan daftar lengkap tentang apa yang membuat pemrograman web unik, tetapi cukup lengkap untuk memberikan kerangka kerja sederhana tentang apa yang perlu saya pelajari: variabel, kondisional, pengecualian, lingkungan lokal/produksi, database, dan cookie. Lihat bagaimana perincian ini berguna? Saya mulai dengan gagasan yang sangat samar tentang apa yang ingin saya lakukan, dan sekarang saya memiliki daftar subketerampilan kunci khusus yang penting untuk dipelajari. Saya belum siap untuk terjun. Ingat saya menyebutkan bahwa komputer tidak dapat memahami bahasa manusia? Saya perlu memilih bahasa pemrograman untuk menulis instruksi program saya, yang akan membutuhkan sedikit lebih banyak penelitian.

Memilih Bahasa untuk Pemrograman Aplikasi Web Ada ribuan bahasa pemrograman yang tersedia, dan bahasa baru ditemukan setiap hari. Sintaks setiap bahasa berbeda, dan sangat diwarnai oleh apa yang dirancang untuk dicapai oleh bahasa. Beberapa bahasa lebih dioptimalkan untuk tugas-tugas tertentu daripada yang lain. Sebelum terjun, saya memutuskan untuk meluangkan waktu satu jam untuk mencari-cari beberapa situs web pemrograman utama untuk melihat bahasa mana yang direkomendasikan oleh pengembang aplikasi web yang berfungsi. Penelitian awal ini akan membantu saya memutuskan bahasa apa yang harus dipelajari dan keterampilan awal mana yang perlu saya praktikkan. Dua tujuan paling populer bagi pemrogram adalah StackOverlow⁷ dan Hacker News,⁸ jadi saya mulai dengan mencari saran tentang bahasa terbaik untuk dipelajari. Stack Overow adalah situs web tanya jawab: tempat untuk bertanya "Bagaimana cara melakukan X?" jenis pertanyaan. Lebih berpengetahuan dan

Pembuat kode berpengalaman biasanya menanggapi pertanyaan dengan saran, pendekatan, atau bug tertentu, menjadikan Stack Overow tujuan terbaik untuk mendapatkan bantuan pada masalah pemrograman yang rumit. Hacker News adalah situs web berita sosial: kumpulan tautan dengan diskusi terkait. Topik di Hacker News berubah dari menit ke menit, tetapi biasanya berkisar pada perkembangan baru dalam pemrograman, teknologi, dan bisnis, menjadikan Hacker News tempat yang ideal untuk menelusuri setidaknya opini semi-informasi tentang perkembangan baru dalam pemrograman. Bahasa, perpustakaan, dan teknik baru dibuat oleh pemrogram di seluruh dunia setiap hari. Beberapa kombinasi teknologi dan pendekatan berguna untuk beberapa masalah, dan yang lain tidak. Seringkali, Anda tidak bisa mengetahuinya sampai Anda mencobanya." Terbaik," dalam istilah pemrograman, relatif terhadap masalah yang ingin Anda pecahkan dan prioritas khusus Anda. Secara umum, sarannya adalah (1) memilih alat yang memungkinkan Anda memecahkan masalah secara kuantikdan (2) jika Anda memiliki pilihan, pilih alat yang Anda sukai digunakan. Menjelajahi arsip Stack Overow dan Hacker News memberi saya kasus besar kelebihan informasi: ada terlalu banyak informasi untuk diproses sekaligus, terutama jika Anda tidak terbiasa dengan terminologinya. Saya perlu mengurangi kebisingan jika saya menginginkan saran yang lebih spesifik. Berikut adalah trik penelitian taktis yang kebanyakan orang tidak tahu:mesin pencari populer memungkinkan Anda membatasi pencarian Anda ke situs web tertentu, bukan ke seluruh web. Kode untuk melakukan ini di Googleterlihat seperti ini:

```
"frasa pencarian" site:example.com
```

Ganti "frasa pencarian" dengan istilah yang Anda cari, dan "example.com" dengan situs web yang ingin Anda cari. Tanda kutip berarti mencari kecocokan persis dari frasa pencarian.

Tanpa tanda kutip, Google akan menampilkan halaman yang berisi semua kata dalam frasa, tetapi tidak harus dalam urutan tersebut. Dengan menggunakan teknik ini, saya mencari beberapa variasi frasa "pemrograman aplikasi web", "belajar kode", dan "pemrograman untuk pemula", lalu menghabiskan satu jam atau lebih untuk membaca hasilnya. Inilah yang saya temukan: pengembang web berpengalaman saat ini merekomendasikan untuk memulai dengan salah satu dari dua bahasa umum: Ruby atau Python. Ruby dan Python, berdasarkan reputasi, relatif mudah dipelajari, kuat, dan memberi Anda dasar yang baik dalam konsep pemrograman penting. Ruby sedikit lebih populer di kalangan programmer yang fokus pada aplikasi web, sedangkan Python lebih populer di kalangan ilmuwan dan matematikawan karena beragamnya perpustakaan sains, matematika, dan grafik. Ruby dan Python masing-masing memiliki komunitas pengembang yang aktif, banyak sumber daya gratis yang tersedia dan buku yang ditulis dengan baik, serta program dan alat yang sudah ada sebelumnya yang membuat fitur-fitur penting lebih mudah diterapkan. Memilih bahasa mana yang akan digunakan tampaknya sebagian besar merupakan masalah preferensi. Setelah membaca beberapa contoh kode yang ditulis dalam masing-masing bahasa ini, saya memutuskan untuk belajar Ruby. Bagi mata saya yang tidak terlatih, kode Ruby terlihat bersih, mudah dibaca, dan tampaknya relatif mudah dipahami. Karena ide dan teknik utama yang saya pelajari menggunakan Ruby akan ditransfer ke bahasa lain yang mungkin saya putuskan untuk dipelajari nanti, saya mungkin juga membuat proses pembelajaran menyenangkan di awal. Selain itu, ada sejumlah program dan alat yang ingin saya gunakan yang mendukung atau membutuhkan Ruby. Secara khusus, Jekyll ditulis dalam Ruby, jadi mempelajari Ruby akan membantu saya memecahkan masalah yang mendesak. Alat serupa ada untuk Python, tetapi terlihat lebih rumit untuk digunakan.

Memilih Kerangka Kerja

Selain saran tentang bahasa pemrograman, pemrogram aplikasi web yang bekerja memiliki pendapat yang kuat tentang kerangka kerja:

pustaka kode yang memudahkan untuk melakukan hal-hal yang perlu dilakukan sebagian besar aplikasi. Perpustakaan penting karena komputer melakukan persis apa yang Anda perintahkan untuk mereka lakukan. Tidak lebih, tidak kurang. Itu rumit, karena kode yang Anda berikan adalah semua yang ada sejauh menyangkut program komputer. Ini seperti kutipan oleh Carl Sagan, fisikawan terkenal: "Jika Anda ingin membuat pai apel dari awal, Anda harus terlebih dahulu menemukan alam semesta." "Alam semesta" program Anda dirusak oleh (1) instruksi dan perintah dalam basis kode, (2) perpustakaan yang diimpor program Anda, dan (3) sistem tempat program berjalan. Jika kode yang diperlukan untuk menyelesaikan operasi tertentu tidak ada di suatu tempat di sistem, program Anda akan mogok atau mengembalikan kesalahan. Sebagian besar bahasa pemrograman berisi banyak perpustakaan umum yang dibutuhkan sebagian besar program, tetapi hanya sedikit alat khusus. Di situlah kerangka kerja masuk. Alih-alih mengkodekan semuanya dari awal, yang akan memakan waktu yang sangat lama, menggunakan kerangka kerja memungkinkan Anda mengimpor dan menggunakan pustaka yang teruji dan andal untuk tugas-tugas khusus, memungkinkan Anda untuk fokus pada inti aplikasi Anda alih-alih menemukan kembali alam semesta. Kerangka kerja bisa besar atau kecil. Beberapa kerangka kerja menyertakan banyak fungsi dan perintah dalam upaya untuk menyelamatkan eort programmer, dan yang lainnya lebih minimal, hanya mencakup beberapa fungsi penting. Saat ini, Ruby menawarkan beberapa kerangka kerja pengembangan aplikasi web utama, di mana dua di antaranya adalah yang paling populer: Ruby on Rails⁹ dan Sinatra.¹⁰ Ruby on Rails (sering disingkat Rails) adalah salah satu kerangka kerja aplikasi web utama pertama yang dikembangkan untuk Ruby. Dibuat oleh David Heinemeier Hansson pada tahun 2004, Rails dengan mudah menjadi Ruby framework paling populer dan digunakan untuk mengembangkan beberapa aplikasi yang sukses di 37signals,¹¹ sebuah perusahaan aplikasi web swasta di mana

Hansson adalah mitra. Hingga saat ini, ribuan bisnis telah mengembangkan aplikasi web penting bisnis besar menggunakan Rails. Rails sangat bergantung pada "generator": program bawaan yang membuat kode boilerplate dalam jumlah besar dengan satu perintah. Boilerplate kemudian dimodifikasi berdasarkan persyaratan unik programmer. Alih-alih menghabiskan waktu berjam-jam membuat aplikasi dari awal, Rails membantu pengembang membuat aplikasi yang berfungsi tanpa banyak semangat, asalkan mereka tahu apa yang mereka lakukan. Sinatra, di sisi lain, adalah kerangka kerja minimal yang dirancang dan dikembangkan oleh Blake Mizerany. Alih-alih mengandalkan generator, Sinatra berfokus untuk memberi pengembang beberapa fungsi umum sederhana yang dibutuhkan sebagian besar aplikasi web, lalu menyingkir. Aplikasi Sinatra terlihat dan terasa sederhana dibandingkan dengan aplikasi Rails. Satu perintah di Rails dapat menghasilkan sepuluh atau lebih folder dan dua puluh atau lebih file. Sebaliknya, tidak jarang aplikasi Sinatra sepenuhnya terkandung dalam satu file. Alih-alih menghasilkan banyak kode yang mungkin perlu dihapus, Sinatra development menghargai menjaga proyek tetap sederhana dan menambahkan kode yang cukup untuk melakukan pekerjaan itu. Seperti memilih bahasa, memilih kerangka kerja sebagian besar adalah masalah preferensi dan memilih alat terbaik untuk pekerjaan itu. Rails cenderung lebih disukai untuk proyek besar dengan banyak pemrogram, dan Sinatra umumnya lebih cocok untuk proyek kecil. Ada banyak fitur yang tumpang tindih, jadi analisis terbaru di RubySource.com menyimpulkan bahwa pada akhirnya ini adalah pilihan subjektif. 12GitHub adalah repositori kode sumber terbuka yang digunakan banyak pemrogram untuk merilis dan memelihara proyek mereka. Sangat mudah untuk menemukan contoh aplikasi yang ditulis di Rails dan Sinatra untuk merasakan setiap kerangka kerja, jadi saya menghabiskan satu jam lagi menjelajahi proyek publik. Ada risiko yang tidak sepele di sini: untuk membuat kemajuan dalam pemrograman, Anda harus berkomitmen pada sesuatu. Setelah Anda

Memilih bahasa dan kerangka kerja, jauh lebih mudah untuk mulaimempelajari semua yang perlu Anda ketahui untuk menulis program. Namun, jika Anda menolak untuk membuat pilihan, Anda dapat menghabiskan waktu bertahun-tahun untuk mencoba mengubah lingkungan pemrograman yang "sempurna". Lebih baik memilih bahasa dan kerangka kerja yang menarik bagi Anda, berkomitmen untuk menjelajahnya untuk sementara waktu, dan menerima tradeos yang tak terhindarkan daripada menghabiskan bertahun-tahun "meneliti" dan membuat kemajuan nol. Menjelajahi Stack Overow dan Hacker News sepanjang hari bukanlah pemrograman. Pada akhirnya, saya memutuskan untuk memulai dengan Sinatra. Meskipun Railsgenerators dapat menghemat banyak waktu jika Anda tahu apa yang Anda lakukan, saya tidak tahu apa yang saya lakukan. Intuisi saya memberi tahu saya bahwa Sinatra adalah pilihan terbaik pada saat ini. Sintaksnya jelas, sederhana, dan mudah dipahami. Satu-satunya kode yang ada di aplikasi Sinatra adalah kode yang ditambahkan pengembang. Kerangka kerja didokumentasikan dengan baik, dan mudah untuk menemukan contohaplikasi kerja di GitHub, serta bantuan di Stack Overow. Saya mungkin bereksperimen dengan Rails di beberapa titik di masa depan. Untuk saat ini, saya mulai dengan Sinatra.

Mendekonstruksi Hasil Akhir

Sekitar dua jam penelitian pendahuluan telah menghasilkan semua yang saya butuhkan untuk memulai: dekonstruksi pemrograman web, bahasa, kerangka kerja, dan proyek khusus. Saatnya untuk mulai bekerja." Kode aplikasi Sinatra yang melayani situs web Jekyll "mewakili tingkat kinerja target yang layak, tetapi saya perlu mendekonstruksi apa yang masuk ke dalam pernyataan itu untuk mengetahui apa yang harus dilakukan selanjutnya. Apa yang harus saya lakukan? Satu jam lagi penelitian menentukan hal-hal berikut:

1. Menjalankan Jekyll membuat situs web nished dari les statis lokal. Saya perlu membuat template HTML untuk situs web dengan tag pemformatan khusus, dan mengekspor

arsip posting saya dari PersonalMBA.com, yang dijelaskan dalam tutorialdibuat oleh Paul Stamatiou.¹³² Aplikasi Sinatra menangani permintaan dari pengunjung situs web, mengirimkan les yang diminta. Saya harus menulis aplikasi ini dari awal.³ Baik situs web Jekyll yang diinstal dan aplikasi server Sinatra perlu diunggah ke host situs web.⁴ Untuk menyelesaikan semua tugas ini, saya perlu menjelaskan cara menginstal Ruby versi terbaru di mesin saya, serta Sinatra dan program lain yang saya butuhkan.

Persyaratan terakhir ini adalah contoh yang baik untuk mendapatkan alat penting. Jika saya tidak dapat mengetahui cara menginstal Ruby di mesin saya, saya tidak dapat melakukan langkah selanjutnya, jadi ini adalah tempat terbaik untuk memulai.

Penting untuk dicatat bahwa teknologi web berubah setiap hari. Sangat mungkin bahwa rangkaian perintah tertentu di bagian ini akan usang pada saat Anda membaca ini. Jangan khawatir: metodenya adalah yang penting, bukan perintahnya.

Demikian juga, Anda akan tergoda untuk mengabaikan bagian kode dari bab ini. Ini adalah dorongan alami: terlihat rumit, dan Anda tidak akan segera mengenali apa artinya. Saya mendorong Anda untuk melawan dorongan ini. Nama, perintah, dan simbol ini area yang tidak asing bagi saya seperti halnya bagi Anda. Bab ini adalah tentang proses mencari tahu apa arti barang ini dan bagaimana menggunakannya. Jika Anda mencoba membaca kodenya, Anda akan mendapatkan lebih banyak hal dari bab ini.

Seterusnya!

Meningkatkan Ruby

Saya sudah memiliki komputer, yang merupakan permulaan: Anda tidak dapat memprogram tanpanya. Saat ini, saya menggunakan Apple MacBook Air yang menjalankan sistem operasi Mac OS X 10.6. Pencarian cepat menunjukkan bahwasistem operasi hadir dengan Ruby versi 1.8.7 yang sudah diinstal sebelumnya, yang merupakan kabar baik: secara teoritis, saya dapat segera mulai menjalankan program Ruby di komputer saya sendiri. Masalahnya adalah Ruby 1.8.7 bukanlah versi terbaru dari Ruby.Ketika saya mencoba menginstal Jekyll, sistem memberi tahu saya bahwa program tersebut membutuhkan

setidaknya versi 1.9.1, yang berarti saya harus mencari cara untuk meningkatkan. Kembali ke Google saya pergi. Beberapa pencarian dasar menemukan dua program yang dirancang untuk membuat pengelolaan instalasi Ruby lebih mudah: rbenv dan ruby-build. Kedua program dikelola oleh Sam Stephenson, seorang pengembang Ruby di 37signals. Bersama-sama, program-program ini membantu Anda menginstal versi baru Ruby dan memberi tahu komputer Anda versi Ruby yang digunakan. Ada tutorial di dokumentasi rbenv halaman 14 yang menunjukkan untuk menginstal program di mesin Anda. Begini tampilan perintah instalasi:

```
$ cd ~$ git clone git://github.com/sstephenson/ruby-build.git$ cd ruby-build$
sudo ./install.sh$ cd .. $ git clone git://github.com/sstephenson/rbenv.git
.rbenv$ mkdir -p ~/.rbenv/plugins$ cd ~/.rbenv/plugins$ git clone
git://github.com/sstephenson/ruby-build.git$ echo 'export
PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashprofile$ echo 'eval "$(rbenv
init -)"' >> ~/.bashprofile$ exec $SHELL$ rbenv install 1.9.3-p125$ rbenv
rehash$ rbenv global 1.9.3-p125
```

Ini terlihat mengintimidasi, tetapi itu hanya daftar perintah. Mari kita uraikan. Perintah-perintah ini dimasukkan ke dalam program bernama Terminal, yang sudah diinstal sebelumnya di komputer Apple. Jika Anda pernah melihat peretas dalam film mengetik dengan panik ke komputer yang menampilkan baris teks yang panjang, komputer tersebut menjalankan program Terminal. 15 Saya membuka Terminal dan memasukkan perintah pertama:

```
$ cd ~
```

Perintah ini mudah dipahami. \$ adalah apa yang ditampilkan Terminal ketika siap untuk perintah baru, jadi sudah ada. CD adalah singkatan dari "ubah direktori", yang merupakan istilah lain untuk folder. Pencarian cepat memberi tahu saya ~ adalah singkatan dari "user's homefolder", folder di komputer saya tempat user prole saya disimpan. Saya mengetik perintah dan menekan tombol Enter. Sekarang, Terminal menampilkan ini:

```
Joshkaufman $
```

Itu kabar baik: Saya berada di direktori beranda saya. Sejauh ini baik-baik saja. Saya ketik perintah kedua:

```
$ git clone git://github.com/sstephenson/ruby-build.git
```

Komputer mengembalikan:

```
git: perintah tidak ditemukan
```

Sepertinya program git tidak diinstal di mesin saya. Saya harus mencari cara menginstalnya.

Apa itu "Git"?

Setelah mencari instruksi tentang cara menginstal git di Mac OS X, dan the Heroku Toolbelt.¹⁶ Heroku¹⁷ adalah perusahaan hosting server aplikasi web, jadi mereka memiliki kepentingan untuk memudahkan pengembang membuat aplikasi web. Heroku Toolbelt adalah program yang menginstal beberapa alat pengembangan perangkat lunak umum yang dibutuhkan pemrogram untuk mengembangkan aplikasi di Heroku. Git¹⁸ adalah salah satu program tersebut. Saya mengunduh paket penginstal, menjalankannya, dan menerima informasi bahwa semuanya diatur dengan benar. Saatnya mencobaperintah lagi:

```
$ git clone git://github.com/sstephenson/ruby-build.git
```

Saya mendapatkan output ini:

```
Kloning menjadi ruby-build ... remote: Menghitung objek:
1004, done.remote: Mengompresi objek: 100% (453/453),
done.remote: Total 1004 (delta 490), digunakan kembali 937
(delta 431)Objek penerimaan: 100% (1004/1004), 108,77
KiB, selesai. Menyelesaikan delta: 100% (490/490), selesai.
```

Keberhasilan! "Selesai" adalah pertanda baik, dan saya tidak mendapatkan pesan kesalahan. Maju. Saya melanjutkan dengan perintah lainnya. Berdasarkan informasi dalam tutorial instalasi, yang saya lakukan hanyalah mengunduh les yang diperlukan, menggunakan perintah yang disebut echo untuk secara otomatis menambahkan sedikit teks ke les kongurasi komputer saya, lalu memulai ulang program di komputer saya yang disebut SHELL untuk menyimpan perubahan. Setelah SHELL di-reboot, rbenv dan ruby-build diinstal. Yay! Sekarang saatnya menginstal Ruby versi terbaru:

```
$ rbenv menginstal 1.9.3-p125
```

Program ini secara otomatis mengunduh kode sumber Ruby dan membangunnya, memuntahkan sejumlah informasi pengguliran yang mengesankan dalam program Terminal dalam prosesnya. (Sekarang saya mulai merasa seperti programmer Hollywood yang tepat.)¹⁹

```
$ rbenv pengulangan ulang
```

Perintah ini, dari dokumentasi, membantu komputer mengenali ada versi baru Ruby yang diinstal.

```
$ rbenv global 1.9.3-p125
```

Perintah ini menetapkan versi 1.9.3-p125 sebagai versi default Ruby di komputer ini. Tutorial ini memberi tahu saya untuk menjalankan perintah ini untuk memastikan komputer saya menggunakan versi baru:

```
$ rubi -v
```

Ini yang saya dapatkan:

```
Ruby 1.9.3P125 (2012-02-16 revisi 34643) [x8664-darwin11.3.0]
```

Keberhasilan! Itulah yang seharusnya dikatakan, berdasarkan tutorialnya. Menurut dokumentasi, jika saya ingin menginstal versi baru Ruby di komputer ini, yang harus saya lakukan adalah menjalankan `rbenv install`, `rbenv rehash`, dan `rbenv global` lagi. Cukup mudah. Meskipun perintahnya terlihat mengintimidasi pada saat pertama, mereka sebenarnya cukup sederhana. Apa yang terlihat seperti omong kosong hanyalah singkatan. Setelah Anda mengetahui apa singkatan itu, perintahnya sendiri mudah dipahami. Ingat, tidak ada yang terlahir mengetahui itu. Sebagian besar waktu, yang perlu Anda lakukan adalah menghabiskan beberapa menit membaca dokumentasi, lalu mencoba apa yang diberitahukannya kepada Anda.²⁰

Menginstal Perpustakaan Ruby (Permata)

Sekarang setelah versi terbaru Ruby diinstal, saatnya untuk mengetahui cara menginstal perpustakaan yang saya butuhkan, termasuk Sinatra. Perpustakaan Ruby disebut permata, dan ternyata sangat mudah dipasang. Berikut perintah yang menginstal permata Sinatra:

```
$ permata instal sinatra
```

Untuk memperbarui Sinatra, berikut perintahnya:

```
$ permata memperbarui sinatra
```

Tidak ada yang lebih mudah dari itu! Namun, sebelum saya menginstal terlalu banyak perpustakaan, saya ingin memastikan program permata sudah diperbarui. Karena komputer saya dikirimkan dengan Ruby versi lama, sepertinya perangkat lunak terkait juga memerlukan pembaruan. Setelah sedikit mencari, saya menggunakan perintah untuk memperbarui program Rubygem:

```
$ permata update--sistem
```

Cukup mudah.

Saat saya menjalankan perintah instal permata, saya melihat bahwa perintah tersebut juga menginstal permata tambahan, seperti rak, perlindungan rak, dan kemiringan. Pustaka tambahan ini disebut dependensi. Sinatra mengandalkannya untuk menjalankan, jadi perintah instal permata menginstalnya secara otomatis.

Memukul Buku

Sekarang setelah saya siap untuk menjalankan aplikasi Ruby, saya memutuskan untuk mengambil dua buku referensi umum Ruby yang sangat direkomendasikan di Stack Overow: *The Well-Grounded Rubyist* (2009) oleh David A.Black dan *Eloquent Ruby* (2011) oleh Russ Olsen. Kedua buku tersebut adalah primer pengantar yang dirancang untuk memperkenalkan pembaca pada konsep dan teknik Ruby yang umum, serta teks referensi dasar. Saya juga membeli *Sinatra: Up and Running* (2011) oleh Alan Harris dan Konstantin Hasse. Buku ini dirancang untuk menjadi pengantar dasar untuk kerangka kerja Sinatra. Meskipun Sinatra didokumentasikan dengan sangat baik secara online, buku ini berisi banyak contoh, yang akan memudahkan untuk mengungkapkan cara menggunakan Sinatra untuk tugas-tugas umum. Saat menelusuri buku, saya juga menemukan beberapa situs web referensi yang mencakup sintaks dasar Ruby:

Panduan Pemula Ocial untuk
Ruby21Penyegaran Ruby22Panduan
Peninjau Keamanan Ruby23

Itu banyak bahan referensi untuk memulai, jadi saya menetapkan menit tambahan untuk melakukan pratinjau singkat dari semua yang telah saya kumpulkan sejauh ini. Saya duduk dengan setiap buku, dan melakukan pemindaian cepat daftar isi dan indeks, mencatat istilah dan ide yang terlihat penting. Saya juga menangkap konsep yang muncul berulang kali, serta urutan pengenalan. Saya membaca berita utama dan bilah samping. Setelah saya selesai dengan buku, saya melakukan hal yang sama dengan situs web.

Inilah yang saya pelajari. Selain variabel, kondisional, pengecualian, dan dasar-dasar pemrograman lainnya, Ruby dibangun di sekitar dua ide inti: objek dan metode. Objek adalah kata benda dari dunia pemrograman: itu adalah hal-hal yang dapat kita lakukan untuk (atau dengan). Katakanlah saya ingin membuat variabel baru di Ruby yang disebut `rstname`, dan saya ingin itu berisi nama saya. Di Ruby, perintah itu terlihat seperti ini:

```
rstname = "Josh"
```

Cukup sederhana. Dengan menempatkan "Josh" dalam tanda kutip, saya memberi tahu Ruby bahwa `rstname` adalah string: urutan karakter alfanumerik. Itu membuat `rstname` menjadi objek di kelas "string". (Aclass hanyalah jenis objek khusus dengan karakteristik tertentu.)String bukan satu-satunya kelas objek. Berikut adalah objek yang ada di kelas bilangan bulat:

```
juta = 1000000
```

Jika objek adalah kata benda dalam pemrograman, metode adalah kata kerja: itu adalah hal-hal yang dapat kita lakukan untuk (atau dengan) suatu objek. Katakanlah saya memiliki dua objek string yang berisi `rst` dan nama belakang saya:

```
rstname = "Josh"nama  
belakang = "Kaufman"
```

Saya dapat menggunakan tanda tambah (+) untuk menggabungkan string ini, yang merupakan istilah mewah untuk "menyatukannya":

```
nama lengkap = nama rst+ nama belakang
```

Kuis pop: apa yang terkandung dalam nama lengkap? Jika Anda menebak "JoshKaufman", Anda salah. Ingat, komputer hanya akan melakukan persis seperti yang Anda perintahkan. Kami tidak memberi tahu komputer untuk menambahkan spasi antara "Josh" dan "Kaufman", jadi tidak. nama lengkap sama dengan "JoshKaufman." Jika kita ingin memperbaiki bug kecil ini, kita harus mengubah kode untuk menambahkan spasi:

nama lengkap = nama rst, + " " + nama belakang

+ adalah metode, dan cara kerja metode tergantung pada objek yang kita gunakan. Jika kita menggunakannya pada bilangan bulat alih-alih string, itu melakukan penjumlahan alih-alih penggabungan:

```
jumlah = juta + juta
```

Apa yang sama dengan jumlah? "2000000" Metode bawaan Ruby dapat membantu Anda melakukan banyak hal keren segera. Katakanlah saya ingin melihat seperti apa nama lengkap saya terbalik. Alih-alih menuliskannya dengan tangan atau menulis program kecil saya sendiri untuk membalikkan huruf, saya dapat menggunakan metode terbalik yang tersedia untuk setiap objek string:

```
nama lengkap.terbalik
```

Inilah outputnya: namfuaKhsoJl juga dapat menggunakan lebih dari satu metode sekaligus. Jika saya ingin membalikkan huruf dalam nama saya dan mengonversi semua karakter menjadi huruf kecil secara bersamaan, saya dapat menjalankan ini:

```
nama lengkap.terbalik.huruf bawah
```

Keluaran: namfuakhsoj. Rapi! Sebagian besar belajar membuat kode di Ruby tampaknya melibatkan penggunaan, pembuatan, dan memanipulasi objek, kelas, dan metode. The language memiliki banyak dari mereka bawaan, dan Ruby memungkinkan Anda untuk membuat, memodifikasi, atau menghapus objek, kelas, dan metode hampir sesuka Anda, yang memberi bahasa banyak kekuatan dan kebolehppercayaan.²⁴ Dokumentasi resmi Ruby²⁵ berisi daftar kanonik dari semua objek dan metode yang tersedia untuk digunakan. Sekilas sekilas memang luar biasa, tetapi ada baiknya untuk menyadari bahwa Anda tidak perlu menggunakan semuanya. Sebaliknya, kebanyakan dari mereka aman untuk diabaikan untuk saat ini. Mereka adalah opsi, siap digunakan saat Anda membutuhkannya. Dokumentasi melayani tujuan lain: ketika Anda mencoba melakukan sesuatu yang tidak dapat dipahami Ruby, pesan kesalahan yang dihasilkan

akan memberi tahu Anda di mana program itu rusak.

Katakanlah kita mencoba menjalankan program seperti ini:

```
hewan = "Zebra"angka
= 7menempatkan
hewan + angka
```

Perintah menempatkan adalah cara lain untuk mengatakan cetak. Kami hanya ingin program menampilkan apa yang dipikirkannya hewan + angka.

Inilah yang saya dapatkan ketika saya mencoba menjalankan program ini:

```
TypeError: tidak dapat mengonversi Fixnum menjadi String dari program.rb:3:in '+'
```

Dalam bahasa non-komputer: Anda tidak dapat menggunakan aritmatika untuk menambahkan angka ke kata dengan cara yang masuk akal, sehingga komputer menampilkan kesalahan. Ini seperti mencoba membagi dengan nol: Anda tidak bisa melakukannya, jadi program berhenti. Untuk x program, kita perlu mengubah angka menjadi astring sehingga metode + akan menggabungkan kedua variabel alih-alih mencoba menggunakan aritmatika, atau memodifikasi program untuk melakukan sesuatu yang lain. Berikut adalah program yang direvisi:

```
hewan = "Zebra"angka
= 7.to_sputs hewan +
angka
```

Saat kita menjalankan program, kita mendapatkan output "Zebra7." Metode bawaan `.to_s` mengubah angka 7 menjadi string, sehingga Ruby dapat menggunakan penggabungan. Kita juga bisa melakukan sesuatu yang benar-benar berbeda, seperti ini:

```
animal = "Zebra"number =
7number.times { menempatkan "#
{hewan}" }
```

Berikut hasilnya:

```
Zebra
Zebra
Zebra
Zebra
```

ZebraZ
ebraZeb
raZebra

Kami baru saja menggunakan perulangan bersyarat dasar, yang dibangun ke dalam Ruby: `number.times` berarti "lakukan X kali ini, di mana X sama dengan nilai variabel angka." Jika kita mengubah nilai hewan atau angka, kita akan mengubah output. (Ya, Anda dapat memodifikasi program ini untuk mencetak "wombat" satu miliar kali jika Anda benar-benar mau.)

Mengomentari dan Men-debug

Saat saya membaca, saya juga mengambil fitur dasar lain di Ruby: berkomentar. Setiap kali Anda memulai baris program dengan `#` (sering disebut "tanda pon" atau, lebih jarang, "octothorpe"), Ruby menafsirkan baris itu sebagai komentar dan melewatkannya. Menambahkan komentar ke program membuatnya lebih mudah untuk diikuti, karena Anda dapat menjelaskan dalam bahasa sederhana apa yang Anda coba lakukan. Berikut adalah tampilan komentar dalam program "Animal Print" saya:

```
# Tetapkan variabelanimalanimal =  
"Wombat"number =  
10000000000# Cetak  
loopnumber.times { menempatkan  
"#{animal}" }
```

Berkomentar juga merupakan teknik pemecahan masalah dasar: Anda dapat mengomentari beberapa baris kode sekaligus untuk mengisolasi masalah atau bug. Dikombinasikan dengan pernyataan cetak atau `puts` yang ditempatkan dengan baik, Anda dapat mengikuti eksekusi program langkah demi langkah untuk memastikan semuanya berfungsi seperti yang diharapkan. Setelah total delapan jam penelitian dan instalasi, saya sekarang menjalankan versi terbaru Ruby, saya dapat menginstal perpustakaan apa pun yang saya butuhkan, dan saya memiliki pemahaman dasar tentang cara kerja program Ruby. Penting untuk dicatat bahwa saya belum benar-benar memprogram substansi apa pun. Waktu sejauh ini telah dihabiskan untuk melakukan

penelitian, menginstal Ruby, dan merasakan seperti apa rasanya menulis program Ruby. Mari kita jelajahi program yang lebih rumit.

Menendang Ban dengan IRB

Dalam penelitian Stack Overow saya, saya menemukan tutorial Ruby online berjudul Learn Ruby the Hard Way²⁶ oleh Rob Sobers dan Zed Shaw. Tutorial menggambarkan cara kerja Ruby dengan memberi Anda contoh program Ruby sederhana dan meminta Anda untuk memodifikasi dan menjalankannya untuk menghasilkan hasil yang spesifik. Jika Anda tidak mendapatkan hasil yang benar, tugas Anda adalah bereksperimen dengan memodifikasi program sampai Anda mendapatkan hasil yang diinginkan. Metode "kode, uji, jalankan, debug" ini adalah contoh yang baik dari perulangan fastfeedback. Saat Anda menjalankan program, komputer akan memberi tahu Anda dalam milidetik apakah itu berhasil atau tidak. Jika ada bug dalam kode Anda, Anda dapat mengubahnya dan menjalankan program lagi, mengujibeberapa variasi dalam waktu satu menit. Bab pertama dari Learn Ruby the Hard Way melibatkan penyiapan Ruby, menginstal program pengeditan teks dasar, dan mempelajari cara menggunakan IRB: program yang menjalankan program Ruby di komputer Anda sendiri. Begini cara kerjanya. Anda mengetikkan program Anda ke dalam editor teks dan menyimpannya dalam le. (Mari kita asumsikan nama le adalah program.rb.) Saat Anda siap untuk menjalankan program, Anda mengetikkan ini ke Terminal:

```
$ irb program.rb
```

IRB akan menjalankan program dan memberi Anda hasilnya. Ini juga akan menunjukkan kepada Anda langkah-langkah yang diambil komputer untuk mendapatkan hasil itu, yang berguna untuk debugging. Jika program tidak benar, IRB akan memuntahkan pesan kesalahan terperinci. Pelajari Ruby dengan Cara Sulit dimulai dengan menetapkan variabel, melakukan aritmatika dasar, memanipulasi string, dan menyiapkan dasar

pernyataan bersyarat, mirip dengan contoh yang baru saja saya sebutkan. Ini adalah pendekatan yang sangat terstruktur dan logis untuk mempelajari dasar-dasarnya. Rencana awal saya adalah membaca Eloquent Ruby dan The Well-Grounded Rubyist, lalu menyelesaikan semua latihan di Learn Ruby the HardWay sebelum mencoba menulis program "nyata" pertama saya. Namun, di sekitar pelajaran 10, saya melihat sesuatu yang penting: Saya menjadi gelisah dan kehilangan minat. Inilah inti dari masalahnya: Saya menyalin program yang dibuat orang lain dan memecahkan masalah yang dibantah orang lain. Program-program ini terkadang menarik, tetapi tidak menyelesaikan masalah saya. Pemrograman mulai terasa seperti latihan akademis alih-alih keterampilan yang berguna. Saya harus keluar dari mode penelitian dan ke mode implementasi. Saya tidak perlu membaca semua buku, kursus, tutorial, dan sumber daya lain yang saya temukan sebelum saya mulai memprogram. Saya harus segera mulai menulis program nyata, lalu merujuk ke sumber daya saya jika dan ketika saya terjebak. Saatnya mengotori tanganku. ...

Aplikasi #1: Situs Web Statis di Sinatra

Saya sudah punya ide untuk aplikasi web pertama saya: aplikasi Sinatra yang menyajikan situs web HTML dasar.

Berikut adalah tingkat kinerja target saya untuk aplikasi ini:

1. Buat aplikasi Sinatra dasar yang berfungsi yang mampu memberikan situs web sederhana kepada pengguna akhir (pembaca).
2. Uji aplikasi di komputer saya untuk memastikannya berfungsi.
3. Terapkan aplikasi itu ke produksi di Heroku, membuatnya "langsung" sehingga pembaca sungguhan dapat menggunakannya.

Itu saja. Tidak ada fitur mewah, hanya program Sinatra yang sangat sederhana berjalan di server publik. Jadi dari mana saya harus memulai? Mari kita tinjau daftar periksa praktik saya:

1. Pilih proyek yang menyenangkan.
2. Fokuskan energi Anda pada satu keterampilan pada satu waktu.
3. Tentukan tingkat kinerja target Anda.
4. Dekonstruksi keterampilan menjadi subketerampilan.
5. Dapatkan alat penting.
6. Hilangkan hambatan untuk berlatih.
7. Luangkan waktu khusus untuk latihan.
8. Buat loop umpan balik yang cepat.
9. Berlatihlah dengan jam dalam semburan singkat.
10. Tekankan kuantitas dan kecepatan.

Saya memiliki satu proyek yang terbentuk dengan baik. Saya telah mendekonstruksi keterampilan, dan saya tahu seperti apa program ini akan terlihat ketika saya selesai. Itu membawa saya ke alat penting: Apakah ada sesuatu yang saya perlukan untuk menyelesaikan proyek ini yang belum saya miliki? Ternyata, ya: Saya belum memiliki akun Heroku. Itu mudah untuk x: Saya mengunjungi Heroku.com, mengklik tombol Daftar, memverifikasi alamat email saya, dan membuat kata sandi. Karena saya telah mengunduh Heroku Toolbelt (program Saya gunakan untuk menginstal git), permata Heroku sudah ada di komputer saya, jadi saya juga berada di sana. Berdasarkan instruksi, ada satu hal terakhir yang perlu saya lakukan untuk mengizinkan komputer saya berbicara dengan Heroku: menghasilkan sesuatu yang disebut "kunci SSH", le khusus yang tampaknya berfungsi sebagai kata sandi. Setelah saya memiliki kunci, saya seharusnya mengunggahnya ke Heroku sehingga sistem dapat mengenali komputer saya dan memberinya akses. Untungnya, Heroku memiliki tutorial tentang cara melakukan ini.²⁷ Saya menjalankan perintah ini untuk menghasilkan kunci:

```
$ ssh-keygen -t rsa
```

... perintah ini untuk masuk ke Heroku:

```
$ Login Heroku
```

... dan perintah ini untuk menambahkan kunci ke akun Heroku saya:

```
$ Heroku Kunci:Tambahkan
```

Hebat: Saya masuk. Sekarang bagaimana cara mulai menulis aplikasi?

Membuat Aplikasi Dasar

Saatnya menelusuri dokumentasi Heroku. Hebat: ada dua panduan yang terlihat berguna:

"Memulai Ruby di Heroku"²⁸"Menerapkan Aplikasi Berbasis Rak"²⁹

Berdasarkan panduan ini, sepertinya saya perlu:

1. Buat program les di komputer saya.
2. Tambahkan mereka ke "repositori git". (Apa pun itu ...)
3. Gunakan perintah `git push heroku` untuk mengirim aplikasi yang dihapus ke Heroku.

Untungnya, tutorial ini menyertakan sebuah contoh, dan ini adalah aplikasi Sinatra! Ini akan lebih mudah dari yang saya kira. ... Saya membuat folder baru di komputer saya. Folder ini disebut folder "root", dan setiap file dalam program akan disimpan di sini. Selanjutnya, saya membuka editor teks saya (saya menggunakan TextMate³⁰) dan membuat tiga file, mengikuti instruksi:

```
Aplikasi.bkon  
g.rugamele
```

Inti dari program ini akan masuk ke `application.rb`. Ruby aplikasi selalu berakhir dengan `.rb`. `config.ru` adalah tempat pengaturan konfigurasi Rack pergi. Ingat, Sinatra dibangun di atas Rack, jadi masuk akal jika ia memiliki konfigurasi terpisah. "Rackup" file diakhiri dengan `.ru`. Gemfile adalah tempat untuk menentukan permata mana yang akan digunakan program. Program Anda hanya akan memiliki satu Gemfile, jadi selalu disebut "Gemfile". Sepertinya cukup sederhana. Setelah membuat file, dokumentasi Heroku menyarankan menulis program dasar "Halo, Dunia!" untuk menguji pengaturan. Inilah yang ada di `application.rb`:

```
membutuhkan
'sinatra'get '/'
do "Hello World!"
ujung
```

Inilah yang masuk cong.ru:

```
require './application.rb'run
Sinatra::Application
```

Dan inilah yang ada di Gemle:

```
Sumber 'http://rubygems.org'gem
'Sinatra'
```

Itu tidak banyak kode, dan cukup mudah dipahami.

Gemle memberi tahu server untuk menyertakan permata Sinatra, yang diperlukan untuk menjalankan aplikasi. Permata akan diunduh dari RubyGems.org.cong.ru le menyiapkan aplikasi utama, lalu menjalankan program. Setelah program berjalan, program akan menampilkan "Halo, Dunia!" setiap kali seseorang mengunjungi "/", yang merupakan cara singkat untuk merujuk ke halaman beranda situs web.

Bisakah semudah ini? Tutorial ini memberi tahu saya ada satu hal lagi yang harus dilakukan: menyimpan lesin dalam repositori git (kadang-kadang disingkat menjadi "repo"). Saya tidak yakin apa artinya, tetapi saya tahu git sudah diinstal, dan mereka menyediakan perintah:

```
$ git init .$ git add -A$ git commit
-m "Initial Commit"
```

Perintah `rst` membuat repositori git baru di rootfolder saat ini. Perintah `add -A` menambahkan semua les di folder ke therepo. Perintah `commit -m` melakukan les ke repo, di samping pesan dari programmer yang merinci apa yang sedang dilakukan. (Saya belum jelas tentang perbedaan antara "tambah" dan "komitmen", jadi saya membuat catatan untuk mengeksplorasi ini nanti.)Setelah memasukkan perintah ini, komputer memberi tahu saya:

```
[master (komitmen root) 8ed1099] Penerapan awal
```

```
3 les diubah, 9 penyisipan (+), 0 penghapusan (-)
membuat mode 100644 mode Gemlecreate
100644 mode application.rbcreate 100644
cong.ru
```

Sepertinya berhasil! Hanya ada satu hal lagi yang harus dilakukan: membuat server kosong baru di Heroku, lalu "mendorong" program saya ke server itu. Saya menjalankan perintah ini untuk membuat server:

```
$ Heroku Buat
```

Dan saya mendapatkan tanggapan ini:

```
Membuat pegas terlindung-2049 ... selesai, tumpukan adalah tumpukan adalah bambu-
ree-1.8.7http://shielded-springs-2049.herokuapp.com/ | git@heroku.com:shielded-springs-
2049.gitGit remote heroku ditambahkan
```

Keberhasilan! "Tumpukan" merinci perangkat lunak yang dijalankan server, dan pesan memberi saya URL publik server.

Berikut perintah nal:

```
$ git dorong master heroku
```

Jika semuanya berjalan dengan baik, saya akan secara ocially mendorong aplikasi pertama saya ke produksi di Heroku.Inilah yang saya dapatkan:

```
Heroku menerima pushAplikasi Ruby/Sinatra terdeteksiGemle
terdeteksi, menjalankan Bundler versi 1.0.7Dependensi yang
belum terselesaikan terdeteksi; Menginstal... Menggunakan--
tanpa pengembangan:uji! Gemle.lock akan segera diperlukan!
Centang Gemle.lock ke git dengan 'git add Gemle.lock!' Lihat
indeks sumber
http://devcenter.heroku.com/articles/bundlerFetching untuk rak
http://rubygems.org/Installing (1.4.1)Menginstal perlindungan
rak (1.2.0)Menginstal kemiringan (1.3.3)Menginstal Sinatra
(1.3.3)Mengggunakan bundler (1.0.7)Bundel Anda selesai! Itu
diinstal ke ./bundle/gems/
```

Ukuran siput yang dikompilasi: 500KLaunching ... selesai,
v4http://shielded-springs-2049.herokuapp.com disebarkan ke
Heroku

Sekarang, saat kebenaran ... Saya membuka browser web, menavigasi
tohttp://shielded-springs-2049.herokuapp.com, dan inilah yang saya lihat:

"Halo, Dunia!"

KEMENANGAN!

Peringatan, Peringatan!

Program ini berhasil, tetapi saya juga mendapat pesan peringatan.
Apa itu Gemle.lock? Dokumentasi Ruby Heroku menunjukkan
bahwa sistem menggunakan perpustakaan yang disebut bundler³¹
untuk menginstal permata di Heroku. Ini adalah permata, jadi saya
dapat menginstalnya secara lokal dengan menjalankan:

```
$ Bundler Instal Permata
```

Bundler diperlukan karena Heroku tidak menginstal permata apa pun di
aplikasi Anda secara default. Untuk alasan keamanan, Heroku tidak
memberi saya tingkat akses komputer yang sama seperti yang saya
miliki di mesin saya, jadi tidak mungkin saya dapat menjalankan
instalasi permata Sinatra langsung di akun saya. Alih-alih memberi saya
(atau pengguna lain) tingkat akses sistem yang berbahaya, Heroku
menggunakan bundler untuk menginstal permata yang ditentukan di
Gemfile. Setelah Anda mengidentifikasi permata mana yang ingin Anda
instal di aplikasi Anda, Anda menjalankan perintah ini di komputer Anda:

```
$ bundler instal
```

Perintah ini membuat le baru yang disebut Gemfile.lock di program
Anda. Saat Anda mengunggah les Anda ke Heroku, sistem melihat
Gemfile dan Gemfile.lock, sangat mereka sama untuk keamanan,
mengunduh permata, lalu menginstalnya untuk Anda. Jika Anda
melihat output ketika saya mendorong program ke Heroku, Anda dapat
melihat bahwa sistem menginstal bundler secara otomatis sebagai

Ketergantungan. Alih-alih menampilkan pesan kesalahan, insinyur Heroku menambahkan pengecualian ke program untuk menginstal program secara otomatis dan menampilkan peringatan alih-alih mogok. Sistem bekerja kali ini, tetapi di masa mendatang, saya harus menambahkan Gemfile.lock ke repositori git sebelum saya mendorong aplikasi. Senang mengetahuinya!

Sinatra Naik ke Panggung

Sekarang setelah aplikasi sederhana saya aktif dan berjalan, saya tidak dapat mulai mempelajari cara kerja Sinatra. Dokumentasi Sinatra32 sangat komprehensif, dan penuh dengan contoh, jadi di situlah saya memutuskan untuk memulai. Inti dari aplikasi Sinatra disebut rute. Cara terbaik untuk memahami gagasan ini adalah dengan memberi contoh. Aplikasi Sinatra dasar kami memiliki satu rute, yang berisi "akar" situs web kecil kami. Pengguna internet biasanya menyebutsitus web root sebagai halaman beranda situs web. Jika Anda mengunjungi google.com atau yahoo.com, browser web Anda mengirimkan permintaan ke server Google atau Yahoo. Permintaan ini disebut GETrequest, dan meminta server untuk menunjukkan kepada Anda apa pun yang ada di direktori root situs web. Protokol, atau format yang digunakan komputer untuk mengirim permintaan, disebut HTTP, yang merupakan singkatan dari "hypertexttransfer protocol". Itulah arti "http://" yang sering Anda lihat di awal alamat web. GET adalah jenis permintaan HTTP yang paling umum, tetapi bukan satu-satunya jenis. Ada tiga "kata kerja" HTTP tambahan:

- POST—mengirim sumber daya ke serverPUT
- memperbarui sumber daya di serverDELETE
- menghapus sumber daya di server

Jika Anda pernah memposting komentar publik di situs web, komentar jenaka Anda dikirim ke server menggunakan perintah POST. Jika Anda

membuat kesalahan dan mengedit komentar, pembaruan Anda dikirim melalui perintah aPUT. Jika Anda memutuskan komentar itu bodoh dan memilih untuk menghapusnya, browser mengirimkan perintah DELETE. Rute yang berisi perintah GET, POST, PUT, dan DELETE adalah inti dari cara kerja aplikasi Sinatra. Setiap rute yang Anda buat bersyarat: "Lakukan X jika perintah GET/POST/PUT/DELETE diterima pada rute Y." Rute Sinatra juga dapat berisi variabel, yang disebut parameter. Aplikasi Sinatra biasanya menggunakan parameter sebagai input untuk proses yang terkandung dalam setiap rute. Mari kita modifikasi aplikasi Sinatra sederhana kita "Halo, Dunia!" untuk menyapa pengguna kita dengan nama. Berikut rute yang akan melakukannya:

```
Gate/hello/:name' do"Halo, #  
{params[:name]}!" Dan
```

Anda dapat melihat ini beraksi di <http://rst20hours.com/hello/name>. Feel gratis untuk mengganti "nama" dengan nama Anda. Itu berhasil! Aplikasi ini bekerja dengan memungkinkan server untuk melihat apa pun yang ada di bagian "nama" rute, kemudian menggunakannya di aplikasi. Itu perintah dalam rute adalah instruksi sederhana untuk menampilkan parameter "nama" kepada pengguna. Sinatra memungkinkan Anda untuk memberi nama parameter (seperti :name), tetapi juga memiliki parameter "wildcard" (juga disebut "splat") yang dapat berisi apa saja. Inilah cara kami menggunakannya di aplikasi "Hello" kami yang dimodifikasi:

```
dapatkan '/hello/*' do"Halo, #  
{params[:splat]}!" ujung
```

Itu cukup keren. Antara parameter bernama dan wildcard, Anda dapat membuat beberapa rute yang sangat cerdas. Rute yang Anda buat menentukan cara kerja aplikasi Sinatra Anda. Itu cukup detail untuk menjelaskan cara menulis program yang memuaskan tingkat kinerja target saya. Karena Jekyll menciptakan

sebenarnya les yang akan disampaikan aplikasi sebagai tanggapan atas permintaan user GET, yang perlu saya lakukan adalah menulis beberapa rute yang menerima permintaan ini, dan le yang benar dalam sistem, dan mengirimkannya ke pembaca. Berdasarkan dokumentasi Jekyll, program ini menempatkan halaman nishedweb dalam folder yang disebut "site" di direktori root. Rute ke halaman itu secara otomatis dibuat oleh Jekyll. Jika kita ingin halaman About situs web kita tersedia di <http://example.com/about>, we atur rute /about di Jekyll les kita, dan program akan membuat le di `_site/about/index.html` di folder root situs web kita. Itu berarti saya harus membuat rute baru di Sinatra untuk membaca respons lein terhadap permintaan GET pengguna. Begini tampilannya:

```
# Index handlerget '/' do##  
File.read("_site/index.html")end# Post handlerget  
/*/*? do## File.read("_site/#  
{params[:splat]}/index.html")end
```

`File.read("")` adalah perintah yang dibangun ke dalam Ruby. File adalah objek, dan `.read` adalah metode. Penggunaannya cukup mudah: apa yang ada di bagian `(")` adalah lokasi le yang Anda inginkan untuk dibaca oleh program, relatif terhadap folder root aplikasi. Mudah. Bagaimana jika le tidak ada? Itu membutuhkan pengecualian, dan Sinatra memiliki dua rute kesalahan dasar bawaan: `not_found` dan `kesalahan`. Mari kita buat kedua rute mengembalikan halaman kesalahan yang sama:

```
not_found do##  
File.read("_site/error/index.html")end  
error  
doFile.read("_site/error/index.html")e  
nd
```

Yang lainnya tetap sama. Saya tidak akan membuat perubahan apa pun pada cong.ru atau Gemle kami. Saya hanya menambahkan rute baru ke repositori Git program, melakukan perubahan, lalu mendorong program yang diperbarui ke Heroku. Selesai. Ingin melihat seperti apa program kami yang diperbarui sedang beraksi? VisitPersonalMBA.com: situs sekarang menjalankan Jekyll alih-alih WordPress menggunakan program yang tepat ini. Menggunakan program pengujian beban yang disebut seige, situs web saya sekarang dapat melayani lebih dari dua ribu pembaca bersamaan tanpa berkerenget. Sebagian besar permintaan halaman dikirimkan dalam delapan belas hingga dua puluh milidetik, jadi situs saya sekarang sepenuhnya terlindungi dari crash karena berat trac.My aplikasi web pertama kali selesai. Butuh waktu sekitar satu jam untuk mengungkapkan cara melakukan langkah-langkah ini, dan satu jam lagi untuk mengetahui cara mentransfer informasi dan desain situs web saya dari WordPress dan ke Jekyll. Total waktu untuk menyelesaikan: sekitar sepuluh jam, yang mencakup penelitian dan tinjauan konsep pemrograman. Tidak buruk!

Aplikasi #2: Codex, Database Catatan Pribadi

Aplikasi pertama saya berfungsi, dan berfungsi dengan baik. Sederhana, tetapi itu adalah manfaat utama dalam hal ini. Lebih sedikit bagian yang bergerak berarti lebih sedikit cara untuk program rusak. Mari kita lihat sesuatu yang sedikit lebih kompleks. Ingat diskusi database kita sebelumnya? Halaman web dasar tidak dapat memperbarui dirinya sendiri, sehingga tidak dapat menyimpan informasi. rstatik aplikasi hanya berfungsi karena les statis: mereka tidak berubah. Setiap perubahan pada les dilakukan melalui Jekyll, yang merupakan program terpisah. Aplikasi ini cepat dan stabil karena tidak mengandalkan database. Bagaimana dengan aplikasi yang menggunakan database? Database adalah bagian besar dari aplikasi web secara umum, jadi saya perlu memahami caranya

mereka bekerja. Untuk mempelajari cara kerjanya, saya harus mulai mengerjakan proyek yang mengandalkan database. Salah satu aplikasi yang saya gunakan setiap hari adalah Backpack,33yang dikembangkan oleh 37signals. Manfaat utama Backpack adalah membuat "halaman" yang dapat berisi hampir semua hal: teks, daftar, gambar, les, dan sebagainya. Saat Anda menyimpan informasi di halaman diRansel, Anda dapat mengaksesnya nanti dari komputer mana pun, karena semua informasi disimpan dalam database aplikasi. Saya bertanya-tanya: Bisakah saya membuat sesuatu yang serupa sendiri? Patut dicoba ... tetapi bagaimana cara memulai? Saat meneliti Jekyll, saya membaca esai oleh Tom Preston-Werner, programmer yang menciptakan Jekyll, berjudul "Readme DrivenDevelopment." 34 Inti dari esai ini adalah bahwa bertentangan dengan gelombang mode manajemen proyek yang melanda industri perangkat lunak setiap beberapa tahun, cara terbaik untuk membuat aplikasi adalah dengan menulis dokumen Readme sebelum Anda melakukan hal lain. Readme adalah le yang disertakan oleh pemrogram dalam folder root aplikasi di samping kode. Le berisi informasi tentang cara mengatur, mengatur, dan menggunakan program. Baca juga penting karena banyak program tidak cukup jelas. Tanpa sedikit dokumentasi, biasanya sangat sulit untuk mengungkapkan cara menggunakan program. Menggali kode untuk menemukannya sendiri tidak sekuno membaca penjelasan terperinci yang ditulis oleh programmer aslinya. Tom berpendapat bahwa yang terbaik adalah menulis Readme lerst program Anda, sebelum Anda mulai membuat kode. Sebagian besar programmer membuat kode rst, kemudian (mungkin) menulis Readme. Itu adalah kesempatan yang terlewatkan: menulis dokumentasi pertama membantu Anda mengetahui dengan tepat bagaimana program akan bekerja. Readme dapat menjadi alat desain sebanyak alat dokumentasi. Itu masuk akal bagi saya. Salah satu teknik pengembangan produk yang saya pelajari dalam proses mengerjakan The Personal MBA

Menulis salinan penjualan sebelum membuat OER, bukan setelah selesai. Dengan mengungkapkan apa yang diinginkan calon pembeli dan memasukkannya ke dalam salinan penjualan Anda, Anda mendapatkan pemahaman yang lebih lengkap tentang apa yang dibutuhkan produk untuk menarik pelanggan. Riset pemasaran menginformasikan perkembangan OER itu sendiri. Saya mengeluarkan buku catatan dan menulis daftar hal-hal yang saya inginkan aplikasi untuk dilakukan, serta kualitas yang saya inginkan untuk dimiliki aplikasi tersebut:

Program ini adalah aplikasi referensi dan pencatatan sederhana. Aplikasi ini dirancang untuk satu pengguna. Aplikasi ini menggunakan Sinatra dan database untuk membuat, menyimpan, memperbarui, dan menghapus catatan halaman. Aplikasi ini memungkinkan pengguna untuk membuat halaman yang memiliki format mewah seperti tebal, miring, judul, dan sebagainya. Aplikasi ini memerlukan kata sandi untuk mengaksesnya, dan menjaga informasi di basis data seaman mungkin. Aplikasinya terlihat bagus. Aplikasi ini dapat dengan mudah diterapkan ke Heroku atau host serupa lainnya.

Saya akan menyebut aplikasi ini Codex, istilah lama untuk "buku", karena aplikasi ini akan berguna terutama untuk referensi informasi, daftar, dan sejenisnya. Istilah seni pemrograman web untuk aplikasi semacam ini adalah "CRUD", yang merupakan singkatan dari Create, Read, Update, Delete. Perlu disebutkan bahwa fungsi-fungsi ini pada dasarnya sama dengan GET, POST, PUT, DELETE, jadi membangun aplikasi semacam ini tentu saja dimungkinkan menggunakan rute Sinatra. Perbedaan besar adalah pengenalan database. Jenis opsi database apa yang tersedia di Heroku? Saya tidak tahu: kembali ke dokumentasi. Secara default, Heroku menggunakan database yang disebut Postgres.³⁵ Setiap aplikasi baru diberi database pengembangan kecil secara default. Itu berhasil untuk saya, tetapi bagaimana cara menggunakannya, dan apa yang saya gunakan untuk menguji programnya di komputer saya sendiri?

Masukkan DataMapper

Untuk menjawab pertanyaan-pertanyaan ini, saya memutuskan untuk mencari Stack Overow. Konsensus adalah bahwa menggunakan perpustakaan yang disebut DataMapper³⁶ membuat pengembangan semacam ini jauh lebih mudah. DataMapper adalah jenis program yang disebut "object relationalmapper", biasanya disingkat ORM.³⁷ ORM memecahkan masalah mendesak bagi pemrogram: database sering menggunakan bahasa mereka sendiri, yang berbeda dari bahasa yang digunakan programmer untuk membuat aplikasi. Bahasa database yang paling umum disebut SQL,³⁸ tetapi ada ratusan lainnya. Mari kita asumsikan kita seorang programmer untuk Amazon.com, dan kita ingin menampilkan daftar buku karya J. K. Rowling, penulis seri HarryPotter. Berikut adalah tampilan perintah SQL:

PILIH * DARI Buku DI MANA penulis = "J.K. Rowling" URUTKAN BERDASARKAN judul;

Perintah ini mengambil semua catatan dari database Buku yang memiliki author eld yang berisi "J.K. Rowling" dan mengembalikannya dalam urutan abjad berdasarkan judul. Sayangnya, mendapatkan SQL atau bahasa kueri database lainnya untuk bermain dengan baik dengan bahasa seperti Ruby bisa jadi rumit. Cukup sulit untuk memprogram dalam satu bahasa, apalagi beberapa bahasa sekaligus. Di situlah ORM masuk: mereka memungkinkan programmer untuk menulis kode dalam satu bahasa, yang kemudian diterjemahkan oleh ORM ke dalam bahasa database. Jauh lebih mudah. DataMapper, kemudian, adalah perpustakaan yang membuatnya lebih mudah untuk berkomunikasi dengan database menggunakan Ruby. Secara default, DataMapper menyediakan banyak fitur berguna untuk membuat, membaca, memperbarui, dan menghapus catatan database. Karena DataMapper telah ada untuk sementara waktu dan telah diuji secara menyeluruh, dalam banyak kasus DataMapper lebih dapat diandalkan daripada mencoba menulis kode database Anda sendiri. DataMapper tersedia sebagai permata, yang diinstal seperti ini:

```
$ permata instal data_mapper
```

Karena DataMapper adalah perpustakaan yang sangat besar, dimungkinkan juga untuk menginstal sepotong demi sebagian. Ini adalah konsep yang disebut "modularitas", dan itu adalah ciri khas pemrograman yang baik.

Berikut adalah perintah yang menginstal semua permata individu:

```
$ gem install dm-core dm-aggregates dm-constraints dm-migrations  
dm-transactions dm-serializer dm-timestamps dm-validations dm-types
```

Alih-alih menginstal seluruh perpustakaan, Anda hanya dapat menginstal bagian-bagian yang akan digunakan program Anda, yang jauh lebih kuno.

Menggunakan DataMapper

Sekarang DataMapper diinstal, saya harus mencari cara menggunakannya untuk (1) berbicara dengan database, dan (2) mengatur database untuk menyimpan dan mengambil informasi yang saya inginkan. Berdasarkan dokumentasi Postgres Heroku, perintah berikut akan memungkinkan aplikasi Sinatra saya untuk berbicara dengan database:

```
DataManager.setup(:d default, ENV['DATABASE_URL'] || "sqlite3://#{  
{Dir.pwd}/database.db")
```

Dalam hal ini, || adalah cara lain untuk mengatakan "atau".

ENV['DATABASEURL'] adalah variabel yang digunakan Heroku untuk mendukung database aplikasi Anda. Jika database itu tidak tersedia, itu akan menggunakan opsi kedua, database yang disebut SQLite. SQLite diinstal di komputer Mac secara default, jadi siap digunakan. DataMapper dapat berbicara dengan Postgres dan SQLite jika saya menginstal dua permata ini:

```
$ permata instal dm-sqlite-adapter dm-postgres-adapter
```

Ini berarti bahwa aplikasi saya akan menggunakan Postgres saat berjalan di Heroku, tetapi SQLite saat berjalan di komputer saya. Dalam kedua kasus apa pun, kode saya akan sama, meskipun database berbicara dalam bahasa yang berbeda. Itu sangat keren.

Berbicara tentang menjalankan aplikasi ini di komputer saya ...
bagaimana Ido itu?

Tahanan perang!

Saya mencari Stack Overow dan Hacker News untuk informasi tentang cara menjalankan aplikasi semacam ini secara lokal di mesin saya. Untungnya, ada beberapa opsi. Tampaknya saya dapat menginstal perpustakaan (sepertiMandor atau Shotgun) yang akan menjalankan aplikasi ketika saya memasukkan perintah ke Terminal, atau saya dapat menginstal program yang membuat program tetap berjalan sepanjang waktu. Sebuah program yang mengambil pendekatan kedua adalah Pow,40 "server Rack tanpa kongurasi untuk Mac OS X." Situs ini menjanjikan untuk memudahkan pengaturan hosting pengembangan lokal di komputer saya dalam waktu kurang dari satu menit.

Kedengarannya bagus bagi saya! Menginstal Pow membutuhkan waktu sekitar sepuluh detik: diperlukan perintah singleTerminal untuk mengunduh dan menginstal aplikasi. Setelah diinstal, Anda menjalankan perintah untuk menautkan program Anda ke Pow, dan Pow akan memungkinkan Anda menjalankannya di mesin Anda. Ada permata Ruby bernama Powder41 yang membuat proses ini semakin mudah:

```
$ Instal bubuk permata
```

Setelah permata diinstal, Anda menjalankan ini untuk menginstal Pow:

```
$ bubuk instalasi
```

Kemudian, Anda pergi ke direktori root aplikasi Anda, dan masukkan perintah ini:

```
$ tautan bubuk
```

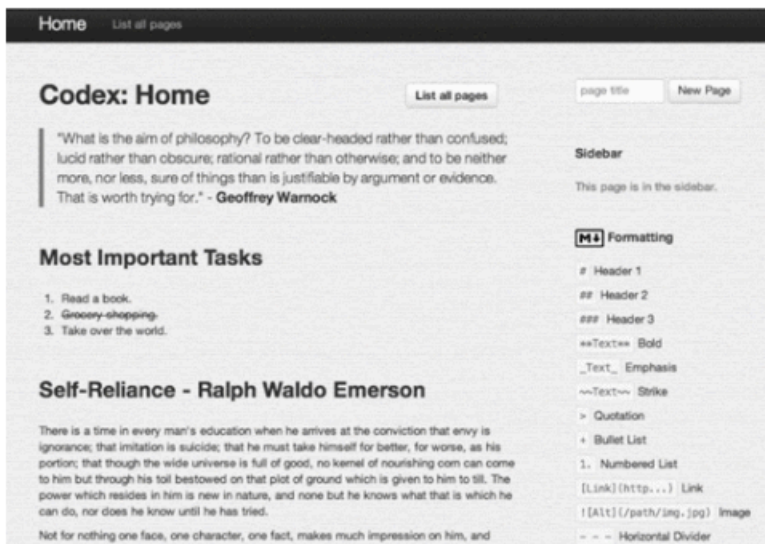
Itu saja. Direktori root saya disebut "codex", jadi aplikasi saya sekarang berjalan di mesin pribadi saya pada <http://codex.dev>, dan saya dapat menguji pekerjaan saya. Jika saya membuat perubahan, perintah ini memulai ulang program:

\$ bubuk restart

Mudah. Saya sekarang siap untuk mulai membangun. Saya menyisihkan satu setengah jam setiap malam untuk membuat kode, dan saya akan terus berjalan sampai selesai.

Kode, Uji, Tinjau

Pada titik ini, saya akan menggambarkan apa yang saya lakukan versus bagaimana saya melakukannya. Anda dapat melihat kode lengkap at <https://github.com/rst20hours/codex> jika ingin mengikuti. Inilah yang saya inginkan untuk aplikasi ini terlihat ketika saya selesai:



Anda akan melihat desain ini memiliki tiga bagian: bilah navigasi atas, area konten utama, dan bilah samping. Saya menyusun desain dasar ini menggunakan Bootstrap⁴² yang dibuat oleh Mark Otto dan Jacob Thornton, keduanya pengembang di Twitter. Alih-alih membuat desain web dari awal, Bootstrap adalah perpustakaan HTML dan CSS yang telah ditulis sebelumnya yang gratis untuk digunakan. Menggunakan Bootstrap menghemat banyak waktu: Anda dapat menyusun dasar

prototipe dari tampilan aplikasi yang Anda inginkan dalam hitungan menit, bukan hari. Unit dasar dari aplikasi ini adalah "Halaman", yang menampilkan catatan yang disimpan dalam database. Informasi Halaman masuk ke area konten utama. Ada tombol yang mengarah ke layar yang mencantumkan semua Halaman dalam database, dan dua tombol di bagian bawah. `rstbutton` memungkinkan Anda untuk mengedit Halaman saat ini, dan yang kedua memungkinkan Anda menghapusnya. Bilah sisi berisi tiga fitur utama. Di bagian atas, ada formulir yang memungkinkan Anda membuat Halaman baru, yang memerlukan judul. Kedua, ada daftar Halaman yang telah ditambahkan pengguna ke bilah samping, yang bertindak sebagai bagian referensi cepat. Ketiga, ada referensi pemformatan, yang membantu pengguna mengingat cara menggunakan fitur pemformatan umum. Bilah navigasi di bagian atas cukup sederhana. Ini berisi tautan ke halaman beranda, serta tautan sekunder ke layar "Daftar semua halaman". Saya dapat menambahkan item tambahan ke bilah navigasi nanti jika saya mau, tetapi hanya itu yang benar-benar saya butuhkan untuk saat ini. Setiap aplikasi web memiliki halaman beranda, jadi saya perlu memutuskan apa yang ingin saya miliki di halaman itu. Dalam hal ini, saya hanya ingin menampilkan Catatan rumah di database. Jadi apa yang ada di Halaman? Karena setiap Halaman adalah catatan basis data, dan catatan memiliki `eld` yang berisi informasi aktual, saya perlu memberi tahu `DataMapper` `eld` mana yang harus disiapkan. Berikut kodenya:

```
class Pageinclude
  DataMapper::Resourceproperty
  :id, Serialproperty :title,
  Stringproperty :content,
  Textproperty :lastupdated,
  DateTimeendDataMapper.nalize
```

Kode ini menggunakan `DataMapper` untuk membuat jenis objek baru yang disebut `aPage`. Ruby sekarang dapat menggunakan Halaman seperti objek lainnya, dan saya bisa

membuat dan menggunakan metode yang membangun, memodifikasi, dan menghapus Halaman. Saat saya membuat perubahan pada Halaman, perubahan tersebut disimpan dalam database viaDataMapper.Perintah Datamapper.nalize memberi tahu aplikasi untuk menyiapkan eld-e-mail ini dalam database aktual jika belum ada. Sekarang setelah saya menyiapkan database, saatnya untuk mencari tahu rute mana yang harus diharapkan Sinatra untuk melayani. Berikut daftar saya, berdasarkan apa yang saya ketahui sejauh ini:

```
# Tampilkan halaman berandaget '/'#  
Membuat catatan baru dari "halaman  
baru" formpost '/'# Menampilkan catatan  
yang diminta '/:url/'# Mengedit catatan  
yang diminta '/:url/edit'# Menyimpan  
suntingan pengguna ke notepost  
('/:url/edit'# Menghapus catatan yang  
ditentukan '/:url/'# Mencantumkan semua  
halaman di databaseget '/all/'# Kesalahan  
handlingnot_founderror
```

Itu daftar yang cukup bagus. Aplikasi saya akan berputar di sekitar perintah apa yang saya buat untuk setiap rute.

Siput, siput di mana-mana!

Ingat ketika saya menyebutkan bahwa database agak seperti tumpukan kartu indeks sihir, dan Anda dapat mencarinya dengan cara apa pun yang Anda inginkan? Kita membutuhkan cara untuk mencari catatan halaman tertentu, itulah sebabnya Anda melihat

:url di rute ini. Isi parameter url akan memberi tahu database catatan mana yang akan diambil. Kita bisa menggunakan judul halaman sebagai parameter, tetapi ada masalah dengan itu: browser web tidak menyukai hal-hal seperti spasi, huruf kapital, dan karakter khusus (seperti \$ dan %) di alamat web. Judul halaman mungkin sangat baik menyertakan hal-hal itu, jadi Anda harus memiliki cara untuk menghapusnya. String yang secara unik mengidentifikasi halaman situs web disebut slug.⁴³ My page slug akan didasarkan pada judul halaman, dengan aturan berikut untuk membuatnya ramah alamat web:

1. Semua karakter akan huruf kecil.
2. Tidak ada karakter khusus—hanya alfanumerik.
3. Tidak ada spasi—semua spasi harus diganti dengan tanda hubung.

Cara melakukannya adalah dengan membuat metode yang menerima judul halaman, lalu mengubahnya ke format slug. Substitusi rumit ditangani oleh jenis pemrograman yang disebut ekspresi reguler, yang mengubah atau teks dalam string berdasarkan aturan yang diberikan.⁴⁴ Ekspresi reguler bisa sangat misterius, tetapi ini adalah penggunaan umum, jadi saya dapat menemukan contoh boilerplate. Berikut metode saya:

```
# Mengonversi nama halaman menjadi post slugdef
slugify(content)content.downcase.gsub(/ /, '-').gsub(/[^a-z0-9-]/,
'').squeeze('-')end
```

Sekarang, saya dapat menggunakan metode slugify untuk mengubah string seperti "PageTitle" menjadi "page-title", sehingga sesuai untuk digunakan dalam alamat web. Selain itu, jika kita menyimpan slug di samping judul halaman, kita dapat menggunakannya untuk mengambil catatan halaman menggunakan parameter :url. Saya menambahkan eld ini ke kelas DataMapper:

```
properti :slug, String
```

Sekarang, setiap kali kita membuat halaman, kita dapat "slugify" judul halaman, menyimpannya di database, dan menggunakannya untuk menampilkan halaman lagi ketika kita ingin mengambilnya. Begitulah cara aplikasi akan mengetahui halaman mana yang akan ditampilkan.

Membuat Halaman

Saya mulai dengan mengerjakan daftar rute yang diperlukan. Rute "rumah" mudah: Saya mengarahkannya ke siput `/home/`, karena saya ingin beranda menampilkan catatan beranda. Rute "buat Halaman" ditautkan ke formulir kecil di bagian atas bilah sisi. Pengguna memasukkan judul halaman dalam formulir, lalu mengklik tombol. Sistem menangkap judul halaman, slugies, lalu menyimpan judul, slug, dan waktu pembuatan di database. Kemudian mengirimkan permintaan GET yang berisi slug, menampilkan halaman baru. Ada detail kecil dalam rute "buat Halaman" yang sangat penting: bagaimana jika Halaman sudah ada? Saya tidak ingin menimpa Halaman jika sudah berisi data. Kehilangan data yang tidak disengaja tidak dapat diterima. Untungnya, `Mapper` memecahkan masalah khusus ini dengan metode bawaan `.rst_or_create`. Sebelum membuat Halaman, `Mapper` akan memeriksa apakah sudah ada. Jika halaman itu ada, `Mapper` tidak akan menyimpannya, dan Sinatra akan mengarahkan browser ke halaman yang ada sebagai gantinya. Masalah terpecahkan. Rute "Halaman Tampilan" membaca slug di URL, mengambil rekaman dari database, lalu menampilkan informasi di area konten utama. Nanti, saya akan menambahkan beberapa pemformatan yang mewah, tetapi untuk saat ini, saya hanya ingin itu berfungsi.

Mengedit Halaman

Mengedit Halaman melibatkan dua rute terpisah. Rute pertama GETs `thePage` yang ingin diedit pengguna, lalu menampilkan konten catatan halaman dalam formulir yang dapat diedit pengguna.

Untuk menampilkan halaman-halaman ini, aplikasi mengandalkan sintaks templat disebut ERB, yang pada dasarnya adalah perintah HTML + Ruby. ERB memungkinkan pemrogram untuk menulis HTML yang menyertakan elemen yang dapat berubah. Karena ERB memproses halaman sebelum ditampilkan kepada pengguna, ERB dapat mengubah teks pada halaman setiap kali halaman dimuat, berdasarkan perintah Ruby dalam templat. Tombol Simpan pada layar pengeditan mengirimkan permintaan POST ke aplikasi yang memperbarui catatan Halaman.

Menghapus Halaman

Menghapus Halaman memerlukan kehati-hatian: ingat, kehilangan data yang tidak disengaja tidak dapat diterima. Jika Anda akan menghapus Halaman, Anda ingin benar-benar yakin bahwa pengguna benar-benar berniat untuk menghapus Halaman itu. Cara yang salah untuk melakukannya adalah dengan menautkan tombol hapus langsung ke rute HAPUS di aplikasi. Pendekatan itu akan menghapus Halaman bahkan jika pengguna mengklik tombol secara tidak sengaja. Strategi yang lebih baik adalah menggunakan proses dua fase. Mengklik tombol Hapus pada halaman akan membawa pengguna ke layar konfirmasi yang menampilkan judul Halaman yang ingin dihapus pengguna. Jika pengguna ingin melanjutkan, mereka dapat mengklik tombol konfirmasi yang mengirimkan permintaan HAPUS. Jika pengguna mengklik tombol Hapus secara tidak sengaja, mereka dapat menekan tombol Batal atau Kembali di browser mereka.

Mencantumkan Semua Halaman

"Daftar semua Halaman" mengarahkan pengguna ke slug `/all/`, yang berbeda dari halaman biasa. Alih-alih mengambil satu rekaman, DataMapper mengambil semua rekaman Halaman dalam database. Templat ERB untuk Halaman berisi perulangan bersyarat yang membuat item daftar untuk setiap Halaman. Setiap item berisi judul Halaman, yaitu

ditampilkan sebagai tautan yang berisi slug terkait Halaman tersebut. Mengklik tautan akan mengarahkan pengguna ke rekaman Halaman individual.

Meluncurkan Aplikasi untuk Pertama Kali

Saya memiliki fitur dasar, tetapi saya mengalami masalah: ketika aplikasi dengan mengunjungi halaman beranda, saya langsung mendapatkan pesan kesalahan. Program ini mencoba mencari catatan Home di database, tetapi tidak ada, karena saya baru saja memulai aplikasi! Solusinya adalah dengan membuat "proses administrasi satu kali" menggunakan program yang disebut Rake. Program Rake disimpan di aRakele, yang terletak di folder root aplikasi. Rakeles bekerja seperti aplikasi Ruby biasa, dengan satu pengecualian: mereka ada di luar program inti Anda, dan Anda harus memotong perintah secara manual. Itu membuat Rake sangat berguna untuk melakukan hal-hal seperti menambahkan informasi default ke database sebelum kita secara resmi menjalankan program aktual. Saya menyalin bit penting dari application.rb ke dalam Rakele, lalu membuat perintah yang membuat Halaman "Beranda" baru di database. Kemudian, yang perlu saya lakukan adalah menjalankan perintah ini sekali:

```
$ pengaturan penggaruk
```

Rake membuat catatan Halaman "Beranda", dan aplikasi saya berhenti menampilkan kesalahan saat startup. Ketika saya mendorong aplikasi ini ke Heroku, saya akan menjalankan perintah Rake dari jarak jauh untuk menyiapkan database sebelum saya mencoba menggunakan aplikasi. Pada titik ini, kami memiliki semua fitur utama. Sekarang, saatnya untuk menambahkan beberapa hal menyenangkan.

Menambahkan Dukungan Bilah Sisi

Saya baru menyadari bahwa saya tidak menyertakan cara untuk menambahkan halaman ke bilah sisi, seperti yang saya maksudkan semula. Fitur semacam ini membutuhkan

Boolean, karena hanya memiliki dua nilai: halaman seharusnya ditampilkan di bilah samping, atau tidak. Saya menambahkan ini ke kelas `DataMapper`:

```
properti :sidebar, Boolean, :d default => false
```

Saya juga menambahkan kotak centang ke layar "edit" di samping "Sertakan bilah dalam?" yang saya tautkan ke bilah sisi `eld` di database. Saya kemudian menulis loop sederhana ke catatan `nd` di database di mana `:sidebar = true`, lalu menampilkannya sebagai daftar, mirip dengan halaman "Daftar semua". Saya memulai ulang aplikasi, mengedit catatan, dan seluruh aplikasi mogok. Yikes! Saya mencoba dan mencoba untuk mengungkapkan apa yang salah, tetapi saya tidak beruntung. Setelah menyalin dokumentasi `DataMapper` dan mencari Stack Overflow, saya menemukan bahwa menggunakan variabel Boolean dengan cara ini tidak cocok dengan formulir HTML. Pendekatan lain diperlukan:

```
properti :sidebar, Enum[ :yes, :no ], :d default => :no
```

Ini adalah cara lain untuk melakukan hal yang pada dasarnya sama. Enum, yang merupakan singkatan dari "enumerate", membuat daftar opsi, dan formulir mengatur opsi mana yang akan disimpan dalam database.

Menambahkan Dukungan Penurunan Harga

Sekarang saya ingin memastikan halaman saya dapat berisi format yang mewah, seperti miring, teks tebal, dan judul. Saya telah memilih Markdown sebagai sintaks pemformatan, yang merupakan bahasa markup kecil yang populer dan sangat berguna yang dibuat oleh John Gruber.⁴⁵ Saya sudah akrab dengan cara kerja Markdown sebagai pengguna, karena beberapa aplikasi yang saya gunakan di komputer saya menyertakannya. Saya harus mencari tahu bagaimana membuat program saya memahaminya. Sedikit pencarian memberi tahu saya ada beberapa permata Markdown tersedia. Saya memilih perpustakaan `rdiscount`, yang saya sertakan in `application.rb`:

```
Membutuhkan 'Rdiscount'
```

Rdiscount mengubah teks yang ditulis dalam format Markdown menjadi HTML: browser web pengguna kemudian menampilkan teks tersebut dengan pemformatan yang tepat. Markdown les bukanlah sesuatu yang istimewa dalam diri mereka sendiri: mereka hanya teks les yang ditulis dengan cara tertentu. Itu berarti bahwa saya tidak perlu mengubah informasi halaman saya menjadi Markdown sebelum saya menemukannya ke database. Bagaimanapun juga, itu hanya teks. Yang perlu saya lakukan adalah menelepon rdiscount kapan pun saya ingin menampilkan teks berformat mewah. Berikut adalah perintah yang melakukan pekerjaan, yang saya tambahkan ke templat ERB yang bertanggung jawab untuk menampilkan halaman:

```
<% penurunan harga (@page.konten) %>
```

Metode ini menyedot konten Halaman eld, mengubahnya menjadi HTML, lalu menampilkan hasil akhir kepada pengguna. Mudah.

Menambahkan Keamanan

Bagaimana dengan informasi login? Jika saya memasang aplikasi ini di Heroku tanpa memerlukan nama pengguna dan kata sandi, siapa pun akan dapat melihat apa yang telah saya simpan di database. Ternyata, browser web modern mendukung protokol keamanan yang disebut HTTP Basic Authentication,⁴⁶ yang merupakan cara sederhana untuk mengharuskan pengguna memasukkan nama pengguna dan kata sandi sebelum mereka dapat melanjutkan. Jika pengguna tidak dapat memberikan kredensial akses, mereka akan diarahkan ke halaman kesalahan. Berikut kode yang memungkinkan otentikasi dasar di Sinatra:

```
use Rack::Auth::Basic, "Restricted Area" do |username, password|[nama
pengguna, kata sandi] == [ENV['ADMIN_USER'], ENV['ADMIN_PASS']]end
```

Dalam hal ini, saya menyimpan nama pengguna dan kata sandi yang sebenarnya di Heroku sebagai variabel lingkungan, yang dapat saya atur dengan Terminalcommand. Itu memungkinkan saya untuk menggunakan kode yang sama untuk di-erent

aplikasi, serta menunjukkan kepada Anda kode ini tanpa juga memberi Anda kata sandi saya! Ini juga merupakan contoh yang baik mengapa penting untuk diketahui bahwa Sinatra dibangun di atas Rack. Ada banyak perpustakaan seperti `Rack::Auth::Basic` yang tersedia, dan saya dapat menggunakan salah satu dari mereka dengan Sinatra. Semakin sedikit saya harus menemukan kembali roda, semakin baik. Ada satu lagi bagian keamanan yang ingin saya tambahkan: enkripsi. Saya akan menambahkan enkripsi SSL—jenis keamanan yang sama yang digunakan bank untuk memastikan sesi perbankan online mereka bersifat pribadi—ke aplikasi saya dengan permata `rack-ssl-enforcer`:

```
membutuhkan 'rack-ssl-  
enforcer'gunakan Rak::SslEnforcer
```

Perpustakaan ini memaksa browser web untuk menggunakan koneksi SSL yang aman untuk mengakses situs. Heroku mengizinkan aplikasi yang menghosting di domain Heroku untuk menggunakan SSL secara default, jadi tidak ada lagi yang harus diatur.⁴⁷

Menambahkan Pesan "Flash"

Ada satu fitur terakhir yang ingin saya tambahkan: Saya telah melihat situs web yang menampilkan pesan kecil seperti "Halaman Anda telah dibuat/diedit/dihapus" setelah Anda melakukan sesuatu. Bagaimana Anda melakukannya? Saya mencari-cari sedikit, dan menemukan perpustakaan bernama `sinatra-ash` yang menangani ini.⁴⁸ Pesan-pesan ini disebut pesan "abu", dan mereka bekerja dengan menyimpan sedikit teks di cookie browser pengguna sebelum pergi ke halaman berikutnya. Saat halaman berikutnya dimuat, aplikasi membaca cookie dan menampilkan pesan kepada pengguna. Saya menambahkan perpustakaan ke `Gemfile` dan `application.rb`, mengatur pesan yang ingin saya tampilkan di rute yang sesuai, lalu menambahkan sedikit kode untuk benar-benar menampilkan pesan tersebut di templat ERB saya, dan saya selesai. Aplikasi saya sekarang selesai.

Kode Lengkap

Berikut adalah awal dari Codex Readme le:

Codex adalah aplikasi web referensi pengguna tunggal sederhana yang ditulis dalam Ruby. Codex menggunakan Sinatra dan DataMapper untuk membuat, menyimpan, memperbarui, dan menghapus catatan halaman dari database Postgres sederhana. Aplikasi ini siap untuk penerapan segera di Heroku. Pemformatan Markdown diaktifkan untuk semua halaman, yang memudahkan untuk menulis halaman yang kompleks dengan markup sederhana. Otentikasi HTTP dan SSL paksa untuk alltrac menjaga keamanan informasi Anda. Gaya bootstrap membuat halaman Anda terlihat bersih dan menarik.

Readme berlanjut dengan instruksi terperinci tentang cara menyiapkan aplikasi di Heroku. "Readme Driven Development" adalah pendekatan yang sangat baik. Secara total, Codex membutuhkan waktu sepuluh jam untuk membuatnya. Itu membuat total investasi saya dalam belajar pemrograman menjadi dua puluh jam. (Butuh waktu lebih lama untuk menulis bab ini daripada menulis aplikasi yang sebenarnya.) Setelah membuat Codex, saya menghadiri pertemuan pemrogram Ruby lokal, dan secara sukarela menjelaskan cara kerja program tersebut. Umpan baliknya sangat positif, dan saya dipuji karena betapa bersih, kompak, dan mudah dimengerti kode itu. Salah satu peserta berkomentar bahwa kualitas kode lebih baik daripada proyek yang dia lihat yang ditulis oleh pemrogram profesional. Misi tercapai.

Kemarahan Terhadap Mesin

Saya ingin mengklarifikasi sesuatu: proses yang saya jelaskan terdengar sangat linier dan lugas. Itu karena, sampai sekarang, saya telah menggambarkan apa yang berhasil, bukan apa yang tidak. Mengkodekan aplikasi web yang berguna dan berfungsi seperti menyusun teka-teki dengan beberapa tantangan tambahan: Anda tidak tahu potongan mana yang ada, Anda harus membuat beberapa bagian sendiri, dan jika Anda membuat kesalahan, teka-teki itu meledak.

Inilah proses pemrograman saya yang sebenarnya: Saya akan mendapatkan ide tentang bagaimana menurut saya bagian dari program akan bekerja. Saya menulis beberapa kode, mengujinya, dan itu merusak program. Saya akan mencoba memodifikasinya. Terkadang perubahan saya akan x program, terkadang tidak, dan terkadang akan merusak lebih banyak hal. Jika saya benar-benar terjebak, saya akan mencari pesan kesalahan atau perpustakaan di Stack Overow atau Google. Ketika Anda masih mempelajari apa yang dilakukan semuanya, aplikasi Anda lebih sering rusak daripada yang berfungsi. Anda juga mempelajari tentang nilai hal-hal seperti kontrol versi, yang memungkinkan Anda mengembalikan kode Anda ke versi kerja sebelumnya. Ingat ketika saya menyebutkan bahwa saya tidak benar-benar grok untuk apa git itu? Ini untuk ini: jika Anda mengedit les dan ada sesuatu yang rusak, Anda mungkin tidak dapat menemukan apa yang rusak. Bergulir kembali ke versi sebelumnya yang berhasil adalah anugerah dan kelegaan. Jika Anda tidak bisa, Anda panik. Sekitar titik di mana saya mulai mengkodekan fitur Add to Sidebar, saya merusak aplikasi. Saya mencoba mencari kesalahan apa yang saya lakukan, tetapi saya tidak bisa mengungkapkannya. Jika saya memiliki rambut di kepala saya, saya akan merobeknya. Saat itulah Anda mempelajari nilai git branch dan git merge: Anda dapat membuat salinan eksperimental program Anda, lalu membuat modikasi Anda. Jika kode Anda berfungsi, Anda dapat menggabungkannya kembali ke dalam salinan master. Jika Anda mengacaukan segalanya, Anda dapat menghapus cabang eksperimental Anda tanpa kehilangan semua pekerjaan Anda sebelumnya. Pemrograman itu sulit, dan ada jutaan cara untuk mengacaukannya. Komputer tidak kenal ampun, dan tidak menuntut perintah yang tidak terbentuk. Demikian juga, mudah untuk mengabaikan detail kecil yang menghasilkan hasil yang tidak terduga. Saya mengalami waktu yang buruk dengan bug yang menyimpan banyak catatan kosong ke database, yang muncul di halaman "Daftar semua". Setiap kali saya melihat halaman itu, lebih banyak halaman hantu akan muncul dalam daftar, dan saya tidak dapat mengetahui dari mana asalnya.

Itu akhirnya menjadi bug di rute "daftar semua": Saya menggunakan perintah yang salah untuk mengambil catatan dari database. Setiap kali saya merusak sesuatu, saya belajar sesuatu. Itu adalah salah satu manfaat tersembunyi dari pemrograman. Komputer adalah yang tercepat dari semua loop umpan balik. Jika Anda melakukan sesuatu yang salah, Anda langsung mengetahuinya. Jika Anda melakukannya dengan benar, Anda akan segera melihat hasil pekerjaan Anda. Jika Anda dapat menghindari dorongan untuk melemparkan komputer Anda ke seberang ruangan, umpan balik instan dapat membuat pemrograman cukup membuat ketagihan. Pada awal percobaan ini, saya tidak bisa memprogram sama sekali. Sekarang, saya bisa. Yang diperlukan hanyalah menghabiskan waktu yang diperlukan untuk mengalahkan beberapa kode menjadi tunduk, dan mendorong atau melakukan sedikit penelitian setiap kali saya terjebak. Apa yang saya peroleh untuk eorts saya? Ton. Saya belajar apa itu pemrograman, seperti apa bentuknya, dan mengapa itu berguna. Saya belajar cara membuat aplikasi web yang berfungsi nyata di Ruby dari awal, lalu mendorongnya ke server produksi. Saya mempelajari dasar-dasar Sinatra, Heroku, Jekyll, DataMapper, Rake, dan alat serbaguna lainnya yang dapat saya gunakan untuk membuat aplikasi baru yang berguna. Saya belajar cara memecahkan masalah kesalahan dan bug, lalu x.

Meninjau Metode

Mari kita tinjau inti dari metode yang saya gunakan untuk belajar pemrograman:

Saya menghabiskan beberapa waktu mempelajari apa itu pemrograman dan aplikasi web secara umum, kemudian mendekonstruksi keterampilan ini menjadi subketerampilan yang lebih kecil yang lebih mudah dipahami dan dipraktikkan. Saya menetapkan tingkat kinerja target saya dengan memilih dua proyek khusus yang ingin saya buat, lalu menyangkal seperti apa proyek tersebut saat dibuat. Saya mendekonstruksi proyek-proyek itu menjadi sublangkah yang lebih kecil kemudian mengidentifikasi sublangkah mana yang tampaknya paling penting. Saya memastikan bahwa saya memiliki alat yang saya butuhkan untuk bekerja (seperti Ruby versi terbaru), dan bahwa saya dapat menggunakan alat tambahan yang saya butuhkan. Saya menemukan beberapa sumber informasi pemrograman yang dapat diandalkan, tetapi saya melewatkan pengalengan tutorial demi melompat dan menulis program yang sebenarnya.

- Saya mengerjakan sublangkah terpenting, seperti mengungkapkan cara menguji program di komputer saya, mendorong aplikasi yang diluncurkan ke produksi, dan sebagainya. Saya menggunakan contoh referensi untuk memulai dan membangun kondensi, kemudian menguji berbagai pendekatan untuk mengetahui cara memprogram fitur yang saya inginkan. Ketika saya membuat kesalahan, program mogok dan memberi saya pesan kesalahan, menciptakan loop umpan balik yang cepat. Setelah mendapatkan kesalahan, saya bereksperimen dengan beberapa cara untuk menggantikannya. Jika saya tidak dapat mengatasi masalah sendiri, saya mencari bantuan. Saya terus menggunakan pendekatan build/test/x sampai program saya selesai.

Total waktu yang berlalu: sekitar dua puluh jam. Sepuluh jam dari jam itu adalah penelitian, dan sepuluh sisanya memprogram dua aplikasi, yang sekarang sedang diproduksi dan sedang diproduksi.

Ke Mana Saya Pergi Dari Sini

Saya terus memprogram aplikasi web sejak menyelesaikan proyek-proyek dasar ini, dengan fokus pada program yang membuat menjalankan bisnis saya lebih mudah. Saya bangga mengatakan bahwa seluruh bisnis saya sekarang berjalan pada perangkat lunak yang saya buat sendiri. Aplikasi saya mampu menagih kartu kredit, menyiapkan langganan, mengirim email, dan mengelola akses situs web untuk pelanggan saya. Dengan belajar kode, saya sekarang memiliki pasukan robot kecil saya sendiri yang diprogram untuk melakukan permintaan saya. Berapa lama waktu yang dibutuhkan untuk menulis semua program ini? Total sekitar sembilan puluh jam, termasuk dua puluh jam yang telah saya rinci dalam bab ini. Inilah manfaat tambahan: setiap kali saya menemukan area bisnis saya yang berulang atau membuat frustrasi, saya mulai berpikir dalam kode. Bagaimana cara kerja program yang memecahkan masalah ini? Lebih sering daripada tidak, ada cara untuk mensistematisasi proses dengan cara yang membuat kehidupan sehari-hari saya lebih mudah. Saya juga mengambil trik baru, seperti menyesuaikan komputer saya untuk membuat pemrograman lebih cepat. Saya sedang mempelajari pintasan keyboard di editor teks saya untuk menghemat waktu, dan saya telah meningkatkan Terminal ke iTerm2 dan Z-Shell untuk membuat pengkodean sedikit lebih mudah.

Saya masih berlatih, dan saya bukan ahli dengan standar apa pun. Saya harus meneliti semuanya, dan butuh beberapa saat untuk memecahkan masalah, kesalahan, dan bug. Ini sering membuat frustrasi. Namun, saya membuat program yang memecahkan masalah nyata dengan cara yang mudah dan andal. Itulah yang benar-benar penting. Saya mematikan komputer, dan saya menang.



6Mengetik Sentuh

Pelajaran: Kebiasaan Lama Tidak Selalu Mati Keras

Apa pun yang layak dilakukan dengan baik layak dilakukan dengan buruk di rst.—RAY CONGDON

■ ■ ■

Untuk gambar, video, dan komentar tambahan tentang bab ini, visit<http://first20hours.com/typing>.

Sampai saat ini, saya telah mempelajari keterampilan baru di bidang-bidang di mana saya memiliki sedikit pengalaman sebelumnya. Kurangnya pengalaman itu tentu saja menjadi penghalang pada saat pertama, tetapi setidaknya pikiran saya tidak secara aktif mengganggu proses pembelajaran. Apa yang terjadi ketika Anda mempraktikkan sesuatu yang baru dan otak Anda memberontak pada saat yang sama? Tentu saja ada contoh dramatis untuk memperoleh kembali keterampilan. TigerWoods telah terkenal melatih kembali golfswing yang sudah berkinerja tinggi tiga kali.¹ Terkadang ada baiknya belajar bagaimana melakukan sesuatu yang penting dengan cara yang baru dan lebih baik, bahkan jika itu mengorbankan ketidaksadaran atau frustrasi jangka pendek. Laju teknologi modern membuat pembelajaran kembali sangat umum: Apa yang terjadi jika program perangkat lunak yang Anda andalkan untuk melakukan pekerjaan Anda berubah, atau program baru menjadi cara paling ilmiah untuk menyelesaikan tugas? Bagaimana jika Anda mendapatkan pekerjaan baru yang membutuhkan pekerjaan

dengan alat baru? Mempelajari kembali keterampilan dengan cepat seringkali sama pentingnya dengan mempelajari sesuatu yang baru. Saya penasaran: Bagaimana rasanya melatih kembali diri Anda untuk melakukan sesuatu yang benar-benar penting? Saya mulai membuat daftar hal-hal yang sudah saya ketahui bagaimana melakukannya, kemudian menyisir daftar mencari keterampilan yang (1) sudah saya kuasai, dan (2) yang memiliki beberapa metode untuk mencapai hasil yang sama. Tidak butuh waktu lama untuk menemukan kandidat yang menjanjikan. Keterampilannya? Sentuh mengetik.

Kehidupan di Balik Keyboard

Menurut David Allen, penulis buku terlaris produktivitas *Getting Things Done* (2002), jika pekerjaan Anda memerlukan penggunaan komputer apa pun, mempelajari cara mengetik sentuh adalah satu-satunya hal paling penting yang dapat Anda lakukan untuk meningkatkan produktivitas Anda. Ambang batas kritis tampaknya sekitar enam puluh kata per menit (disingkat WPM). Jika Anda tidak dapat mengetik setidaknya enam puluh WPM dengan tingkat kesalahan yang rendah, Anda benar-benar melumpuhkan kemampuan Anda untuk melakukan pekerjaan yang produktif. Semakin sedikit semangat yang perlu Anda curahkan untuk mengetik, semakin banyak waktu dan energi yang Anda miliki untuk melakukan tugas-tugas bernilai lebih tinggi. Saya telah mengetik sentuh setidaknya selama tujuh belas tahun, mungkin lebih lama. Di sekolah, setiap siswa diharuskan mengambil kursus mengetik di kelas tujuh, dan saya ingat bosan. Saya sudah tahu cara mengetik sentuhan, jadi saya menyelesaikan tugas dalam beberapa menit, lalu menghabiskan sisa waktu kelas untuk mengungkapkan cara membuat pengolah kata memuntahkan karakter yang tampak aneh, seperti pilcrows (¶), tanda bagian (§), dan interrobangs (!). Saya belajar cara mengetik sentuh sebagian besar dengan menghabiskan banyak waktu bekerja dengan komputer di waktu luang saya. Latihan mengetik saya ambient, tidak disengaja: saya tidak secara sadar bekerja untuk meningkatkan kecepatan atau akurasi saya. Saya hanya menggunakan komputer, dan dalam prosesnya, saya belajar cara mengetik.

Itu tidak berarti teknik saya hebat. Tangan saya berputar di seluruh keyboard alih-alih menghabiskan sebagian besar waktu di baris rumah, tombol di tengah keyboard. Metode saya tidak benar buku teks, tetapi itu menyelesaikan pekerjaan, yang hanya saya pedulikan. Meskipun formulir pengetikan saya tidak bagus, itu fungsional. Dalam pekerjaan saya, saya sering berada di depan komputer, dan kecepatan dan akurasi mengetik saya menyelesaikan pekerjaan. Namun, sesekali, saya menemukan artikel tentang tata letak keyboard alternatif: pengaturan tombol yang diterik dari QWERTY, apa yang disebut tata letak default universal yang menghiasi sebagian besar keyboard berbahasa Inggris yang diproduksi setiap tahun:2



QWERTY, argumennya, sangat tidak kuno—desain yang mengerikan. Ada cara lain untuk mengatur tombol pada keyboard yang membantu pengguna mengetik lebih cepat, akurat, dan dengan lebih sedikit semangat. Eort kumulatif adalah masalah yang sangat besar: cedera stres berulang (RSI) dan sindrom terowongan karpal adalah gangguan tangan dan pergelangan tangan yang umum yang dapat disebabkan atau diperburuk oleh mengetik. Meskipun saya belum mengalami gejala yang nyata, saya memiliki beberapa teman dekat yang memilikinya, dan itu tidak menyenangkan. Saya ingin menghindari berurusan dengan gangguan ini jika saya bisa. Saya berencana menulis dan memprogram selama bertahun-tahun yang akan datang, dan kecuali pengenalan ucapan atau pemikiran ke teks menjadi metode utama penggunaan komputer, saya akan mengetik di masa mendatang. Mungkin demi kepentingan terbaik saya untuk belajar cara mengetik

cara yang paling kunyit, bahkan jika itu melibatkan kebingungan atau ketidaknyamanan jangka pendek. Selamat tinggal QWERTY: Saya akan belajar kembali cara mengetik sentuhan.

Bagaimana Tata Letak QWERTY Menjadi Standar

"Universal" Bertentangan dengan kepercayaan populer, format keyboard QWERTY tidak dirancang untuk memperlambat juru ketik: itu adalah solusi untuk masalah teknik mekanik. Di masa lalu, sebelum pengolahan kata dan komputer, mesin tik mekanis membuat karakter muncul di halaman dengan mengayunkan kunci logam kecil, yang disebut bilah huruf, ke arah selembur kertas, yang dililitkan erat di sekitar silinder. Pita tinta terletak di antara kertas dan kunci. Kuncinya mengenai pita, yang bersentuhan dengan kertas, meninggalkan karakter di halaman. Tombol "Return" memutar silinder, memindahkan kertas dan memungkinkan juru ketik untuk terus mengetik pada baris baru. C. L. Sholes, yang dikreditkan dengan penemuan tata letak mesin tik QWERTY, membangun prototipe pertamanya pada tahun 1868. Sholes bukanlah orang yang paling baik untuk membuat mesin tik: setidaknya ada tiga puluh satu penemu lain yang mencoba sebelum dia melakukannya, dan Sholes mempelajari pekerjaan mereka, memasukkan banyak fitur mereka ke dalam desainnya.³ Kunci pada prototipe pertama Sholes disusun dalam urutan abjad, yang masuk akal. Pada saat itu, tidak ada yang membayangkan bahwa orang akan (atau bisa) belajar mengetik sentuhan dengan kesepuluh nger. Menempatkan tombol dalam urutan abjad memastikan bahwa pengguna yang tidak terlatih dapat menggunakan huruf yang sesuai saat mereka mematuhi dengan dua indeks nger. Namun, ada masalah besar dengan prototipe: bilah huruf memiliki kebiasaan buruk untuk saling menempel ketika huruf yang berdekatan ditekan secara berurutan. Pertimbangkan huruf yang paling umum digunakan dalam kata-kata bahasa Inggris: vokal AOEUI, dan konsonan DHTNS. Pada keyboard mekanis abjad, S dan T duduk tepat di sebelah satu sama lain. Jika satu bilah huruf naik sementara yang lain

Turun, mereka akan macet, mengharuskan juru ketik untuk berhenti dan mengurai bilah huruf secara manual. Untuk memperbaiki gangguan kronis ini, Sholes meminta bantuan Amos Densmore, seorang guru. Densmore melakukan studi cepat dan kotor tentang frekuensi huruf dalam bahasa Inggris, yang kemudian digunakan Sholes untuk menempatkan kombinasi huruf umum, seperti TH, di sisi berlawanan dari bank bilah huruf, mencegah tabrakan. Strategi ini tidak sepenuhnya menyelesaikan masalah, tetapi memperbaikinya sedemikian rupa sehingga Sholes memimpin untuk paten pada desain pada tahun 1872.4 E.Remington & Sons, sebuah perusahaan yang saat itu terutama dikenal untuk pembuatan senjata ulang, membeli paten Sholes pada tahun 1873. Setelah membuat beberapa perbaikan mekanis tambahan, seperti menambahkan tombol Shift untuk memungkinkan juru ketik beralih antara huruf kapital dan huruf kecil, Remington mulai memproduksi massal mesin tik yang menampilkan tata letak QWERTY, dengan maksud menjualnya ke pasar bisnis, pada tahun 1874. Remington bukan satu-satunya perusahaan yang menjual mesin tik. Yang lain, seperti Hammond dan Blickensderfer, memiliki perangkat yang bersaing, yang masing-masing memiliki tata letak keyboard yang berbeda. Pada saat itu, bisnis terutama mengandalkan memo tertulis untuk catatan dan korespondensi. Mesin tik berpotensi menghemat banyak peralatan manual, tetapi hanya jika operator tahu cara menggunakannya. Untuk mendapatkan penjualan, perusahaan harus mengatasi kebutuhan untuk melatih juru ketik untuk menggunakan alat aneh. Itu menyebabkan dinamika pasar yang menarik: perusahaan mesin tik merekrut dan melatih juru ketik sendiri, secara efektif mengoperasikan agen penempatan. Jika seorang pebisnis ingin mempekerjakan seseorang yang bisa mengetik, mereka menelepon Remington, yang akan menjual bisnis itu baik mesin tik maupun layanan seseorang yang dapat menggunakannya. Siring waktu, karena semakin banyak bisnis yang mengadopsi mesin tik, QWERTY mulai muncul sebagai standar. Tidak ada den

momen, tidak ada komite penetapan standar hukum atau birokrasi, hanya gerakan pasar yang halus menuju satu solusi yang cukup baik. Bisnis membutuhkan juru ketik dan mesin tik; Remington pandai memasok keduanya. Dan ketika sebuah bisnis membutuhkan mesin tik baru atau juru ketik baru, yang paling bijak adalah membeli mesin QWERTY baru dan menyewa juru ketik yang tahu QWERTY. Selama periode enam puluh tahun, QWERTY menjadi standar de facto, dan tata letak yang bersaing perlahan menghilang. Seiring berjalannya waktu, QWERTY diam-diam mengambil alih dunia.

Kompetisi Muncul: Dvorak

Pada tahun 1932, August Dvorak, seorang profesor di University of Washington, diberi hibah \$ 130.000 oleh Komisi Carnegie untuk Pendidikan untuk meneliti desain keyboard. Salah satu faktor pendorong penelitian ini adalah kesadaran bahwa QWERTY dirancang untuk memecahkan masalah teknik mesin yang sudah tidak ada lagi. Apakah ada cara yang lebih baik untuk mendesain keyboard? Empat tahun kemudian, pada tahun 1936, Dvorak memimpin paten untuk Keyboard Dvorak Simplified, yang dia klaim jauh lebih unggul daripada tata letak lainnya. Dasar untuk klaim ini sederhana: tata letak Dvorak menempatkan karakter yang paling umum digunakan tepat di bawah ngerson pengguna di baris beranda.



Konsonan yang paling umum ditempatkan di sisi kanan baris rumah, dan semua vokal ve ditempatkan di sisi kiri. Pemisahan antara vokal dan konsonan ini menyeimbangkan beban pengetikan secara merata antara tangan kiri dan kanan juru ketik, secara teoritis mengurangi kelelahan dan meningkatkan kecepatan. Dvorak juga mengklaim tata letaknya lebih mudah dipelajari, dan mengejar studi penelitian dengan organisasi yang melatih sejumlah besar juru ketik, terutama, mereka yang berada di militer AS. Hasilnya beragam: karena Dr. Dvorak menjalankan penelitian ini sendiri, banyak hasil yang menguntungkan diragukan, mengingat kepentingan nansialnya yang jelas dalam hasilnya. Sebuah studi independen yang dikendalikan oleh General ServicesAdmission, yang bertanggung jawab untuk melatih juru ketik pemerintah AS, menemukan bahwa dibutuhkan lebih dari seratus jam untuk mendapatkan kembali kecepatan mengetik sebelumnya jika mereka berlatih kembali di Dvorak. Itu terlalu lama. Oleh karena itu, penelitian ini merekomendasikan standarisasi juru ketik pemerintah di QWERTY. Produsen mesin tik dan bisnis mengikutinya. Dvorak, sebagai tata letak, terus ada sebagai opsi pinggiran, tetapi sebagai standar, itu adalah kegagalan. Meskipun memiliki banyak klaim yang valid atas QWERTY, itu tidak dapat menggantikan status quo. Dvorak mendekam selama beberapa dekade, nomor dua yang sangat jauh.

Penantang Baru Muncul: Colemak

QWERTY dan Dvorak bukan satu-satunya dua tata letak keyboard bahasa Inggris yang ada. Meskipun sebagian besar juru ketik belajar QWERTY, penemu dan penghobi telah mengejar inovasi dalam desain keyboard selama beberapa dekade. Sebagian besar tata letak alternatif ini tidak pernah mendapatkan daya tarik luas, tetapi pada tahun 2006, tata letak keyboard baru mulai menarik perhatian juru ketik yang suka berpetualang, sebagian besar berkat Internet.

Inilah ceritanya. Shai Coleman, seorang pemrogram komputer, memutuskan untuk mencoba desain keyboard. Tujuannya sederhana: untuk membuat tata letak yang memiliki keahlian relatif Dvorak, tetapi lebih mudah dipelajari. Salah satu masalah utama dengan Dvorak adalah bahwa itu mengubah segalanya: setiap kunci huruf didirikan dalam Dvorak versus QWERTY. Jika Anda awalnya belajar mengetik di QWERTY, besarnya perubahannya membuat belajar Dvorak menjadi mimpi buruk mutlak. Lebih buruk lagi, sekarang komputer adalah alat pengetikan utama, biasanya mengandalkan pintasan keyboard untuk menyelesaikan tugas berulang, seperti menyimpan le, memotong dan menempelkan teks, dan sebagainya. Jika Anda terbiasa dengan kombinasi keyboard tertentu yang bekerja dengan cara tertentu, Dvorak sangat menggelegar: tidak hanya semua huruf bergerak, tetapi semua pintasan juga berbeda. Strategi Coleman adalah menggabungkan analisis komputer dari kumpulan data dokumen bahasa Inggris yang sangat besar dengan gagasan melestarikan tombol sebanyak mungkin untuk mempertahankan pintasan keyboard yang sama. Sisi kiri keyboard, serta sebagian besar baris bawah, sebagian besar dikiri. Hanya kunci yang paling signifikan yang diubah. Algoritma Coleman merekomendasikan sebagian besar perubahan, hanya menyisakan beberapa kunci ambigu yang memerlukan panggilan penilaian. Pada akhirnya, tata letak baru Coleman, yang dia beri nama Colemak, mengubah tujuh belas tombol dibandingkan dengan tata letak QWERTY standar. Berdasarkan angka-angkanya, itu jauh lebih tua daripada QWERTY, dan sedikit lebih tua daripada Dvorak. Bahkan lebih menjanjikan, karena Colemak mengganti lebih sedikit tombol, kemungkinan akan lebih mudah bagi pengguna QWERTY untuk belajar. Coleman membuat situs web, colemak.com, yang berisi detail tentang tata letak baru, serta instruksi tentang cara menginstal dan mempelajarinya. Dibandingkan dengan eorts mahal Dvorak untuk mempopulerkan tata letaknya dalam pertemuan tatap muka dengan organisasi besar,

Internet membuat penyebaran berita tentang keberadaan Colemak benar-benar murah. Akibatnya, Colemak sekarang menjadi tata letak keyboard bahasa Inggris terpopuler ketiga, di belakang QWERTY dan Dvorak. Tata letak sekarang dibundel di sebagian besar sistem operasi komputer baru secara default dan memiliki ribuan pengguna di seluruh dunia. Tidak buruk sama sekali untuk tata letak yang dikembangkan oleh satu penghobi yang sudah ada kurang dari satu dekade.

Jika ragu, uji

Dengan semua tata letak keyboard alternatif ini, bagaimana Anda memutuskan mana yang terbaik? Sederhana: Anda mengujinya. Teknologi pemrograman modern telah membuatnya lebih mudah untuk mengumpulkan data keras tentang berbagai pendekatan desain keyboard. Alih-alih menguji dan mengumpulkan data secara manual, dimungkinkan untuk menggunakan program yang menganalisis tata letak keyboard untuk Anda. Salah satu program tersebut disebut carpalx,5 ciptaan Martin Krzywinski, seorang programmer yang bekerja untuk British Columbia Genome Sciences Centre. Carpalx dirancang untuk "melakukan simulasi stokastik untuk tata letak yang meminimalkan skor eort untuk serangkaian parameter model tertentu." Dengan kata lain, carpalx dapat secara otomatis menguji ketepatan berbagai tata letak keyboard dengan menjalankannya terhadap teks sampel. Krzywinski menggunakan carpalx untuk merancang tata letaknya sendiri, tetapi dia juga menggunakannya untuk membandingkan Colemak dengan QWERTY dan Dvorak, menghasilkan serangkaian data komparatif yang sangat besar dan menyeluruh. Inilah yang dia temukan:6

QWERTY keluar sebagai pecundang besar di sini, dengan peningkatan besar atas Colemak di base eort (+193 persen), peningkatan jalur stroke besar (+36 persen) dan peningkatan penalti asignant (+16 persen). Dvorak sudah menjadi peningkatan QWERTY, jadi perbedaan antara QWERTY dan Colemak lebih kecil.

Colemak bahkan lebih banyak menggunakan barisan kandang (74 persen) daripada Dvorak (71 persen). Ini membuat QWERTY 34 persen jauh di belakang. Dan penggunaan baris bawah Colemak rendah pada 9 persen, seperti Dvorak. Colemak lebih seimbang dalam penggunaan tangan, dengan preferensi 6 persen untuk tangan kanan (Dvorak memiliki 14 persen untuk kanan dan QWERTY memiliki 15 persen untuk kiri). Colemak sangat pandai mempertahankan pergantian tangan untuk kedua tangan. Keduanya Colemak dan Dvorak lebih banyak menggunakan kelingking. Colemak menggunakan kelingking 16 persen dari waktu (18 persen untuk Dvorak dan 10 persen untuk QWERTY). Colemak melakukan pekerjaan dengan baik dalam memuat nger yang lebih kuat (indeks dan tengah) dan menggunakannya 67 persen dari waktu. Ini lebih baik daripada Dvorak, yang menggunakan ini 60 persen dari waktu, tetapi tidak sebagus QWERTY, yang menggunakannya 69 persen dari waktu.

Berdasarkan data Krzywinski, dibutuhkan hampir dua kali lebih banyak eort fisik untuk mengetik pada keyboard QWERTY, dibandingkan dengan keyboard Colemak. Colemak juga mengalahkan Dvorak sedikit, yang mengesankan, mengingat Colemak hanya mengubah tujuh belas tombol pada tata letak QWERTY standar, versus dua puluh empat Dvorak. Mengingat datanya, saya akan belajar Colemak. Saya suka itu menyelamatkan tanpa sepenuhnya mengatur ulang pintasan keyboard umum, yang sering saya gunakan. Jadi apa yang harus saya lakukan selanjutnya?

Seperti Apa Colemak Itu?

Berikut adalah diagram seperti apa tata letak keyboard Colemak:7



Selain mengubah posisi tombol karakter yang paling sering digunakan, Colemak memetakan ulang tombol Caps Lock sebagai tombol Hapus kedua. Perubahan ini merupakan salah satu inovasi terbesar di Colemak. Tidak ada seorang pun (selain troll Internet) yang menggunakan tombol Caps Lock secara teratur, tetapi menempati real estat utama: ini mudah dijangkau oleh kelingking kiri, dan tepat di sebelah tombol A di baris rumah. Mengubah Caps Lock menjadi Hapus memungkinkan juru ketik Colemak untuk mengoreksi kesalahan tanpa menggerakkan tangan kanan mereka di baris beranda untuk menekan Hapus tombol di ujung kanan atas keyboard, menghemat banyak eort. Perubahan tunggal itu menghasilkan pengurangan jarak nger sebesar 15 hingga 20 persen dibandingkan dengan QWERTY. Semakin banyak kesalahan yang Anda buat yang Anda perbaiki dengan kelingking kiri Anda, semakin tua Colemakbe.

Bagaimana Anda Mengaktifkan Mode Colemak?

Dengan komputer, mengubah tata letak keyboard itu mudah: hanya masalah memberi tahu komputer untuk menggunakan letak letak yang berbeda, mekanisme yang sama yang digunakan untuk beralih ke keyboard dalam bahasa lain. Pengaturan ini biasanya terletak di panel "Sistem dan Preferensi" utama komputer. Colemak disertakan dalam sistem operasi Mac OS X pada versi 10.5, jadi tidak ada yang perlu dipasang.⁸Selain itu, beberapa keyboard, seperti TypeMatrix 2030,⁹ mendukung Colemak di perangkat keras itu sendiri. Alih-alih mengarahkan komputer untuk mengenali Colemak, keyboard menerjemahkan penekanan tombol Colemak menjadi bit QWERTY, sehingga karakter yang benar muncul di layar. Hasilnya, Anda dapat mengetik Colemak tanpa mengubah apa pun di komputer sama sekali, asalkan Anda bersedia menghabiskan sekitar seratus dolar untuk keyboard ergonomis yang bagus.¹⁰Mengubah pengaturan di komputer adalah bagian yang mudah, tetapi itu bukan satu-satunya hambatan. Karena sebagian besar keyboard berbahasa Inggris hadir

dengan QWERTY secara default, beralih ke Colemak berarti huruf yang tercetak pada tombol tidak akan cocok dengan karakter yang akan muncul di layar. Jika Anda mengandalkan pengetikan string karakter yang aneh, seperti kata sandi yang rumit, itu adalah resep untuk kebingungan. Bagaimana cara menggunakan keyboard Colemak fisik? Ada dua pendekatan umum: Anda dapat membeli keyboard kosong, atau mengubah keyboard QWERTY ke format Colemak. Beberapa keyboard, seperti TypeMatrix 2030, tersedia diversifikasi Colemak, yang merupakan solusi mudah jika Anda bersedia membeli keyboard baru. Itu tidak membantu saya: Saya mengetik di laptop, jadi keyboardnya sudah terpasang.

Memodifikasi Mesin Saya

Untungnya, mudah untuk memunculkan tombol o keyboard Apple dan mengatur ulang tanpa merusak komputer. Karena Apple menggunakan tombol gaya "Chiclet" yang semuanya memiliki dimensi yang sama, mengubah keyboard menjadi Colemak adalah proyek menit ve.11



Dengan menggunakan obeng berlubang kecil, saya dengan lembut mengangkat sisi kanan atas setiap kunci, memasukkan ujung obeng, lalu memindahkannya ke bawah

di sepanjang tepi kanan kunci. Setelah obeng sampai ke tengah tombol, tombol muncul dari mekanisme "gunting" bawaan keyboard, yang membuat tombol memantul kembali setelah ditekan. Setelah Anda menguasainya, sangat mudah untuk membuka tombol yang perlu diubah. Saya kemudian menggunakan diagram tata letak Colemak untuk meletakkan kembali tombol pada keyboard di tempat yang benar. Yang diperlukan hanyalah menekan lembut, dan tombol-tombol berbunyi klik secara permanen pada tempatnya. Ini adalah prosedur yang sederhana, tetapi perlu dicatat bahwa ini kemungkinan akan membatalkan garansi pada laptop saya. Demi keamanan, saya menguji ini dengan keyboard nirkabel Apple cadangan, lalu mengganti tombol di MacBook Air saya setelah saya merasa nyaman dengan prosesnya. Sekarang saya memiliki keyboard Colemak asli di laptop saya. Betapa kerennya itu? Ini adalah contoh menghabiskan sedikit waktu untuk mengubah lingkungan untuk mendukung latihan. Sekarang keyboard saya ada di Colemak, akan lebih mudah untuk beralih. Jika saya kesulitan mengingat kunci mana yang mana, saya dapat melihat jika perlu.

Seberapa Cepat Saya Mengetik?

Sekarang keyboard saya sudah siap, saya lebih dekat untuk membuat peralihan besar. Namun, sebelum saya melanjutkan, saya ingin mendapatkan gambaran tentang seberapa cepat saya saat ini mengetik, yang akan membantu saya menentukan tingkat kinerja target saya. Dalam hal ini, tingkat kinerja target saya sederhana: Saya ingin dapat mendapatkan kembali kecepatan mengetik QWERTY saya di Colemak secepat mungkin. Saya tidak ingin melebihinya, karena kecepatan mengetik bukanlah faktor pembatas dalam pekerjaan saya. Saya hanya ingin bisa mengetik sebaik yang biasa saya lakukan dengan lebih sedikit semangat. Satu-satunya informasi yang benar-benar saya butuhkan sebelum sakelar saya adalah kecepatan pengetikan saya saat ini, jadi saya mencari tes kecepatan mengetik online dasar.¹²

Ujiannya mudah: ketika Anda menekan tombol Mulai, program menunjukkan kepada Anda seratus kata acak dari buku lama.¹³ Tugas Anda adalah mengetik sampel secepat mungkin, dengan sesedikit mungkin kesalahan. Setelah selesai, Anda menekan tombol Berhenti, dan program memberi Anda kecepatan mengetik dan tingkat kesalahan Anda. Rencana saya adalah mengikuti tes pertama menggunakan QWERTY, lalu ulangi tes setelah saya beralih ke Colemak, hanya untuk melihat dari mana saya memulai. Saya tekan tombol Mulai, dan mulai mengetik. Ketika saya dihapus, saya menekan tombol Stop dan menerima hasil saya: enam puluh satu WPM, akurasi 100 persen, nol kesalahan. Tidak buruk: Saya mengetik tepat di ambang batas yang direkomendasikan David Allen. Saya bukan iblis kecepatan, tetapi saya bisa mengetik dengan cukup baik untuk menyelesaikan pekerjaan. Saya memiliki semua yang saya butuhkan: Saya tahu seberapa cepat saya mengetik, keyboard saya dalam mode Colemak, dan saya tahu cara mengaktifkan tata letak di sistem operasi. Tidak ada yang tersisa untuk saya persiapkan. Ini dia. Apakah saya siap untuk meninggalkan QWERTY?

Membalik Sakelar

Sekarang atau tidak sama sekali. Saya mengalihkan komputer saya ke mode Colemak, lalu tutup panel pengaturan. Mulai sekarang, saya tidak akan dapat mengetik apa pun di QWERTY sampai saya menyelesaikan percobaan. Browser web saya masih memiliki tes pengetikan yang dimuat. Saya memulai pengatur waktu dengan mouse saya dan mulai mengetik. Inilah kesan pertama saya: #%&@. Dengan risiko terdengar melodramatis, rasanya seperti bagian dari otak saya telah dihilangkan. Saya terbiasa dengan kata-kata yang berutang tanpa henti dari otak saya ke komputer. Sekarang, saya tidak tahu di mana kuncinya. Saya telah mengerti sebagian besar dari mereka, bahkan tombol yang belum berubah dari QWERTY, yang tidak masuk akal. Saya melihat sesuatu di layar, dan nger saya bergerak tanpa sadar, jadi saya mengetik omong kosong, yang kemudian harus saya hapus.

Saya membutuhkan beberapa detik untuk mengetik kata-kata sederhana, dan saya khawatir saya memakai tombol Hapus. Setiap kata adalah perjuangan baru. Saya melihat ke jam, dan saya membutuhkan beberapa menit penuh untuk mengetik dua kalimat. Saya bahkan belum setengah jalan. Saya serius mempertimbangkan untuk berhenti, tetapi memilih untuk melanjutkan. Seiring berjalannya waktu, saya hampir berhenti setidaknya sepuluh kali. Pada akhirnya, saya membutuhkan waktu hampir dua puluh menit untuk mengetik seratus kata. Kecepatan mengetik baru saya: ve kata per menit. Bunuh aku sekarang. Saya menulis untuk mencari nafkah, dan saya baru belajar cara memprogram. Sekarang, saya tidak bisa melakukan keduanya. Bagaimana saya akan menjawab email? Bagaimana cara kerjanya? Apa yang telah saya lakukan?

Saya telah melihat musuh, dan itu adalah saya

Ini adalah penghalang utama yang mencegah kebanyakan orang mempelajari tata letak keyboard baru. Aspek teknis switching sepele dibandingkan dengan aspek emosional. Ketika Anda terbiasa dengan tingkat kecepatan atau kemudahan tertentu dalam menyelesaikan tugas, apa pun yang kurang tampak mengerikan. Ini terutama benar dengan mengetik: jika Anda terbiasa mengetik tanpa eort, dan tiba-tiba membutuhkan banyak semangat, terus menekan terasa benar-benar menyakitkan. Yang lebih buruk lagi adalah pengetahuan bahwa jika Anda kembali ke cara Anda dulu melakukan sesuatu, semuanya akan lebih baik lagi. Pikiran kita tidak membantu kita di sini: otak kita memiliki kecenderungan keras kepala untuk berasumsi bahwa apa yang kita alami saat ini akan terus menjadi benar di masa depan. Saat ini, pikiranku panik. Jika saya hanya bisa mengetik kata-kata per menit, saya tidak akan pernah bisa bekerja lagi! Karier saya akan berakhir! Keluarga saya akan kelaparan! Itu tidak benar tentu saja, tetapi terasa benar pada saat itu. Pengalaman emosional itu adalah penghalang terbesar untuk belajar.

Saya menutup komputer dan mengeluarkan buku catatan dan pena. Saya membutuhkan rencana untuk mencapai kecepatan pengetikan fungsional di Colemak, dan saya membutuhkannya sekarang.

Memetakan Ulang Otak Saya

Urutan bisnis pertama dan paling mendesak adalah saya tidak tahu di mana kunci baru berada. Tentu, saya memiliki lembar referensi yang dicetak, dan tombol fisik pada keyboard dalam format Colemak, tetapi otak saya saat ini tidak mampu memetakan keinginan untuk mengetik huruf ke gerakan nger tertentu. Membantu otak saya memetakan tata letak baru ke dalam keterampilan motorik adalah prioritas utama saya. Saya harus berfungsi secepat mungkin. Untungnya, orang telah belajar mengetik sentuhan selama beberapa dekadesekarang, jadi ada alat komersial yang dikembangkan dengan baik yang dapat membantu. Tutorial mengetik, seperti Mavis Beacon¹⁴ dan Typing Trainer,¹⁵ telah tersedia selama beberapa dekade, dan tidak terlalu mahal. Sayangnya, program-program ini mengasumsikan bahwa Anda ingin mempelajari cara mengetik sentuh di QWERTY, karena ini adalah taruhan yang aman sebagian besar pelanggan ingin belajar cara mengetik sentuh menggunakan tata letak standar. Program-program ini biasanya dimulai dengan tombol di baris beranda. Baris rumah QWERTY dan baris rumah Colemak berbeda, jadi tutorial QWERTY tidak akan membantu saya. Dvorak telah ada cukup lama sehingga beberapa program mendukungnya, tetapi Colemak relatif baru. Mungkin ide yang baik untuk menggunakan semacam tutorial mengetik, tetapi saya perlu mencari satu yang mampu mendukung Colemak. Untungnya, saya memiliki petunjuk. Saat saya menelusuri Hacker News selama penelitian pemrograman saya, saya menemukan posting di Keyzen,¹⁶ pelatih pengetikan sumber terbuka yang dibuat oleh Rye Terrell. Terrell mengembangkan Keyzen untuk membantu pemrogram belajar cara mengetik lebih cepat. Program ini berjalan di browser web standar, dan menyertakan tanda baca yang tidak biasa seperti tanda kurung, tanda kurung, dan garis miring:

karakter yang cukup sering digunakan programmer, tetapi sebagian besar typingtrainer melewatkan. Terrell memposting kode sumber lengkap untuk Keyzen di GitHub, dan mengundang pemrogram lain untuk menggunakan atau memodifikasinya. Program itu sendiri cukup sederhana: menampilkan satu set tujuh karakter, dimulai dengan huruf di baris beranda. Tugas Anda adalah mengetik karakter dalam urutan itu. Saat Anda mengetik, program memutar eects suara. Karakter yang benar menciptakan dentingan mesin tik klasik, sedangkan karakter yang salah menghasilkan pukulan! dan mengubah karakter menjadi merah. Setelah Anda menyelesaikan satu set, program menyajikan yang baru. Jika Anda salah mengetik karakter, program akan secara otomatis memperkenalkan karakter tersebut ke set selanjutnya. Jika Anda mengetikkan semua karakter dengan benar selama tiga set berturut-turut, Anda akan mendengar bunyi yang sangat memuaskan!, yang berarti Anda telah naik level. Keyzen kemudian memperkenalkan karakter baru, dan pelatihan berlanjut.

Memodifikasi Keyzen

Keyzen, seperti kebanyakan pelatih mengetik, mendukung QWERTY secara default. Namun, karena program ini open source, saya mungkin dapat memodifikasinya agar cocok untuk belajar Colemak. My praktik pemrograman akan sangat berguna. Saya membuat salinan kode sumber Keyzen, lalu membukaprogram les di komputer saya. Program itu sendiri sederhana, jadi mudah untuk menemukan bagian program yang mengontrol karakter mana yang ditampilkan, dan sama mudahnya untuk mengatur ulang mereka sesuai keinginan saya. Program asli memperkenalkan karakter di homerow QWERTY rst, dimulai dengan index ngers, kemudian menambahkan karakter hingga baris beranda selesai. Berikutnya datang baris atas, maju dari tengah keyboard ke pinggiran, lalu baris bawah, mengikuti pola yang sama.

Dengan menggunakan pola pelatihan Keyzen sebagai model, saya mengedit program untuk menggunakan set karakter Colemak. Pola kunci fisik tetap sama, tetapi urutannya sekarang mengajarkan Colemak alih-alih QWERTY. Ketika saya menutup program, saya memiliki tutorial mengetik Colemak saya sendiri. Keberhasilan! Jika Anda tertarik, Anda dapat mencoba program ini sendiri: ini <http://rst20hours.com/keyzen-colemak>.

Keterampilan Motorik Halus

Saya memiliki tutorial mengetik pertama saya, dan tujuan pertama saya sederhana: pelajari di mana setiap tombol berada di keyboard. Saya kembali Keyzen dan mulai mengetik (mewakili spasi):

Ahh

—

Jenis latihan ini sama sekali tidak glamor, tetapi perlu. Dengan mengebor lokasi setiap karakter dalam urutan semi-acak, saya membantu otak saya menerjemahkan melihat (atau berpikir) karakter menjadi gerakan ane-motorik di nger saya. Selain itu, Keyzen membantu saya belajar secara kuantis dalam beberapa cara. Pertama, Colemak dirancang untuk menempatkan huruf yang paling umum digunakan di baris rumah, di bawah huruf terkuat. Keyzen memperkenalkan karakter ini rst, jadi saya menguasai ARSTDHNEIO sebelum masuk ke QWZXYM. Ketika saya membuat kesalahan, Keyzen memperkenalkan kembali karakter ke dalam urutan latihan. Akibatnya, saya menghabiskan sebagian besar waktu saya untuk berlatih huruf yang paling sulit bagi saya. Ini adalah bentuk pengulangan spasi, yang kebetulan dikombinasikan dengan umpan balik instan, membuat praktik ini sangat kuno.

Saya mengatur jadwal latihan: setidaknya dua sesi dua puluh menit, dengan istirahat singkat di antaranya, di penghujung hari, tepat sebelum tidur. Ada metode untuk kegilaan saya: memperoleh semua jenis keterampilan motorik membutuhkan perubahan fisik struktur otak Anda, dan ternyata, tidur berperan penting dalam proses itu.

Belajar sambil tidur!

Selama empat dekade terakhir, akuisisi keterampilan motorik telah menjadi bidang penelitian psikologis yang sangat aktif. Jika Anda pergi ke perpustakaan akademik yang disediakan dengan baik, Anda dapat menemukan rak demi rak penelitian tentang perolehan keterampilan secara umum, dan akuisisi keterampilan motorik pada khususnya. Saya melakukan hal itu. Saya menjelajahi tumpukan psikologi kognitif di Colorado State University, mencari informasi berguna tentang perolehan keterampilan. Saya tidak kesulitan sama sekali membaca buku dan studi. Sayangnya, sebagian besar penelitian ini sangat aneh, penuh dengan jargon akademis. Untungnya, itu tidak semua omong kosong. Berikut adalah kutipan dari sebuah penelitian yang menarik perhatian saya:

Tahapan Pembelajaran Keterampilan Motorik (2005)¹⁷Pembelajaran keterampilan motorik yang sukses membutuhkan pelatihan berulang ... Artikel ini mencakup bukti yang berkembang bahwa pembelajaran keterampilan motorik maju melalui tahapan, di mana mekanisme penyimpanan yang berbeda mendominasi. Fase akuisisi ditandai dengan pembelajaran cepat (dalam sesi) dan lambat (antar sesi). Untuk waktu singkat setelah sesi pelatihan awal, keterampilan ini dapat diganggu oleh keterampilan lain dan oleh penghambatan sintesis protein, menunjukkan bahwa proses konsolidasi terjadi selama periode istirahat di antara sesi pelatihan. Selama pelatihan serta periode istirahat, aktivasi di daerah otak berubah secara dinamis.

"Gangguan" dan "konsolidasi" adalah kata kunci di sini. Saat Anda berlatih keterampilan, otak Anda sangat aktif, bekerja untuk mengubah pola dan menyimpannya dalam ingatan. Dalam kasus keterampilan motorik, mereka

Pola melibatkan mengasosiasikan apa yang Anda lihat, dengar, rasakan, rasakan, dan pikirkan dengan cincin neuron yang mengontrol otot-otot dalam tubuh Anda. Semakin kuat koneksi saraf ini, semakin baik kinerja Anda. Selama latihan, otak Anda sibuk membuat koneksi dan asosiasi ini, tetapi itu tidak berarti mereka disimpan secara instan dalam struktur neuron Anda. Butuh beberapa saat agar pola-pola ini dapat dilakukan, yang terjadi selama proses yang disebut konsolidasi. Konsolidasi terjadi sepanjang waktu, tetapi itu sangat efektif saat Anda tidur. Berikut adalah studi pertama yang saya temukan yang menarik hubungan langsung antara tidur dan perolehan keterampilan:

*It's Practice, with Sleep, That Makes Perfect: Implications of Sleep-Dependent Learning and Plasticity for Skill Performance (2005)*¹⁸ Latihan sering diyakini sebagai satu-satunya penentu peningkatan. Meskipun berulang kali melakukan tugas baru sering menghasilkan manfaat belajar, yang mengarah pada pepatah "praktik membuat sempurna," kumpulan penelitian selama dekade terakhir mulai mengubah konsep ini. Sebaliknya, laporan ini menunjukkan bahwa setelah latihan awal, otak manusia terus belajar tanpa adanya latihan lebih lanjut, dan bahwa peningkatan yang tertunda ini berkembang selama tidur.

Berikut kesimpulan dari penelitian itu:

Meskipun fungsi otak tidur masih belum pasti, literatur yang meningkat pesat sekarang mendukung peran tidur dalam memodifikasi dan meningkatkan daya ingat. Laporan ini memberikan banyak bukti konvergensi yang menunjukkan bahwa mekanisme plastisitas saraf yang bergantung pada tidur mengarah pada konsolidasi memori keterampilan dan akibatnya peningkatan kinerja yang tertunda. Bentuk memori keterampilan yang sederhana dan kompleks tampaknya membutuhkan jenis tidur yang halus untuk peningkatan memori semalaman, dan beberapa penelitian menunjukkan bahwa tidur dalam 24 jam pertama setelah latihan awal sangat penting untuk konsolidasi untuk berkembang.

Akuisisi keterampilan efektif, terutama akuisisi keterampilan motorik, tampaknya membutuhkan tidur, yang memainkan peran utama dalam mengkonsolidasikan keterampilan ke dalam memori jangka panjang. Penelitian terbaru menunjukkan bahwa, karena, yang terbaik adalah tidur dalam waktu empat jam setelah latihan keterampilan motorik: bahkan tidur siang singkat lebih baik daripada tidak sama sekali. Sepanjang waktu,

dan kemampuan otak Anda untuk mengkonsolidasikan informasi yang dikumpulkannya selama latihan terganggu. Itu sebabnya saya berlatih mengetik tepat sebelum tidur. Jika saya tidur dalam waktu satu jam atau lebih setelah latihan, saya dapat membantu otak sayamengkonsolidasikan gerakan motorik dengan lebih efektif. Anehnya adalah saya bisa melihat ini bekerja. Sesi latihan pertama saya sangat mengerikan: Saya tidak bisa melakukan sesuatu dengan benar, membuat kesalahan terus-menerus, dan hampir tidak maju melewati karakter di barisan rumah. Setelah tidur malam penuh, ketika saya duduk di depan komputer, saya menyadari bahwa saya membuat lebih sedikit kesalahan. Tidur telah mengkonsolidasikan apa yang telah saya pelajari malam sebelumnya. Otak kita benar-benar keren.

Interferensi Kognitif

Interferensi adalah kebalikan dari konsolidasi: ini adalah gangguan proses konsolidasi. Jika Anda berlatih atau menggunakan keterampilan serupa kedua segera setelah mempraktikkan keterampilan baru, latihan itu dapat mengganggu kemampuan otak Anda untuk mengkonsolidasikan informasi baru. Periode kritis untuk interferensi juga tampaknya sekitar empat jam. Jika Anda menunggu untuk melatih keterampilan konstruksi setelah konsolidasi terjadi, Anda cenderung tidak mengganggu peningkatan yang Anda peroleh dalam keterampilan utama. Itu sebabnya saya tidak berlatih QWERTY segera setelah berlatih Colemak. Ini akan mengganggu kemampuan otak saya untuk mengkonsolidasikan latihan Colemak saya, memperlambat tingkat perolehan keterampilan saya. Menarik juga untuk dicatat bahwa, setelah tujuh jam total latihan Colemak, saya tiba-tiba kesulitan mengetik di QWERTY, meskipun saya telah mengetik dengan sentuhan di QWERTY untuk waktu yang sangat lama. Otak saya memetakan gerakan motorik mengetik ke Colemak, yang tampaknya membuatnya lebih sulit untuk mengakses QWERTY, setidaknya untuk saat ini.

Setelah otak saya menyesuaikan diri dengan Colemak, saya dapat kembali dan memperoleh kembali QWERTY jika saya mau: berdasarkan apa yang telah saya baca dari Colemaktypist lainnya, dimungkinkan untuk menjadi "dwibahasa keyboard", dan beralih bolak-balik sesuai permintaan. Namun, untuk saat ini, saya fokus secara eksklusif pada Colemak untuk meminimalkan gangguan. Mengaktifkan kembali QWERTY bisa menunggu.

Menghentikan Kebiasaan Melihat

Saya sekarang memiliki tujuh jam latihan yang disengaja di bawah ikat pinggang saya, rata-rata empat puluh menit setiap malam. Saya menyelesaikan seluruh urutan Keyzen tanpa banyak kesusahan. Saya masih membuat kesalahan, tetapi kesalahan itu lebih jarang datang. Ketika saya mengikuti tes kecepatan mengetik, hasil saya jauh lebih baik: dua puluh kata per menit. Pada siang hari, saya mendapatkan beberapa latihan sekitar dalam bentuk email penting yang membutuhkan balasan mendesak. Ketika saya duduk untuk mengetik, itu tidak menyakitkan. Saya lambat, tetapi saya bisa mengekspresikan diri. Itu kemajuan! Namun, saya perhatikan bahwa saya sering melihat keyboard. Memiliki tombol yang benar pada keyboard sangat berguna untuk mengetik string karakter yang aneh, seperti kata sandi, tetapi juga merupakan kruk: setiap kali saya merasa tidak yakin, saya melihat ke bawah. Jika saya ingin mengetik sentuhan, saya harus menghentikan kebiasaan itu secepat mungkin, tetapi itu tidak menyenangkan, karena dibutuhkan sedikit semangat untuk melihat ke bawah.

Papan ketik

Untuk melepaskan diri dari kebiasaan terlihat, saya memutuskan untuk mengambil alat pembelajaran baru: keyboard yang benar-benar kosong. Das Keyboard adalah keyboard paling badass yang pernah Anda gunakan. "Ultimate Model S"19 tidak memiliki tanda apa pun di atasnya. Jika Anda tidak dapat mengetik sentuh, Anda tidak dapat menggunakan Das Keyboard ... masa. Lucu

Untuk melihat reaksi ketika orang melihatnya: Bahkan juru ketik sentuhan yang terampil pun terintimidasi. Saya mencolokkan Das Keyboard dan menutupi keyboard laptop saya dengan selembar kertas, jadi saya tidak dapat melihat tanda tombol sama sekali. Sensasinya mirip dengan ketika saya beralih ke Colemak untuk pertama kali: Saya bingung dan frustrasi, tetapi hanya untuk sesaat. Keterampilan motorik yang saya pelajari dalam tujuh jam latihan pertama dimulai, dan saya menemukan bahwa saya bisa mengetik dengan cukup baik. Setiap kali saya lupa surat, saya harus memburunya selama beberapa detik dengan coba-coba, tetapi saya bisa berfungsi. Tanpa tempat lain untuk mencari, saya fokus pada layar. Das Keyboard melayani tujuannya. Dengan mengubah keyboard, saya mengubah perilaku saya secara otomatis. Bersama dengan Das Keyboard, saya memodifikasi metode pelatihan saya. Mengetik karakter acak menjadi tua, jadi saya beralih ke program bernama TypeFu.20 Selain karakter dan kata acak, Type Fu berisi database peribahasa dan kutipan, yang membuat latihan sedikit lebih menghibur. Program ini juga melacak karakter mana yang paling sering Anda lewatkan, yang berguna. Saya sedang mempelajari J, U, V, dan B saat ini. Setiap malam, saya berlatih selama empat puluh menit. Setelah total empat belas jam latihan yang disengaja, saya sekarang mengetik dengan kecepatan empat puluh WPM. At titik ini, saya sangat fungsional: Saya dapat menggunakan email dan menjelajahi web secara relatif normal. Saya bahkan bisa mengetik ve-pageproposal tanpa terlalu banyak kesulitan. Butuh waktu lebih lama dari biasanya, tetapi itu bukan hal paling membuat frustrasi yang pernah saya lakukan.

Latihan yang Disengaja vs. Latihan Sekitar

Karena saya dapat berfungsi, saya ingin menguji hipotesis: Seberapa penting praktik yang disengaja?

Saat ini, saya berlatih dengan dua cara: Keyzen dan Type Fusions. Saya adalah latihan yang disengaja, karena saya fokus pada tugas dan bekerja secara aktif untuk meningkatkan. Pengetikan saya di siang hari adalah praktik ambient: setiap kali saya menulis email atau esai, saya mengetik di Colemak, meskipun saya lebih fokus pada isi pesan daripada teknik saya. Saya bertanya-tanya: Bagaimana jika saya menghentikan latihan yang disengaja untuk sementara waktu dan terus mengetik email dan menjelajahi web? Saya dua pertiga dari jalan menuju tingkat kinerja target saya enam puluh WPM setelah hanya empat belas jam latihan yang disengaja. Bisakah latihan ambient membawa saya ke sisa jalan, tanpa semangat terfokus tambahan? Saya memutuskan untuk melakukan percobaan: Saya akan menanggung latihan yang disengaja selama tiga puluh hari dan melihat apa yang terjadi. Saya akan terus mengetik secara normal di Colemak, tanpa beralih kembali ke QWERTY. Dengan waktu sebanyak yang saya habiskan di komputer, saya seharusnya bisa mendapatkan latihan sekitar yang cukup untuk mencapai enam puluh WPM, bukan? Setelah tiga puluh hari, saya mengikuti kembali tes mengetik. Ingin menebak kecepatan mengetik? Empat puluh WPM. Nol perbaikan. Meskipun saya cukup banyak mengetik, saya tidak secara aktif fokus untuk meningkatkan keterampilan saya. Latihan ambient tidak cukup untuk ditingkatkan. Jika Anda ingin meningkatkan keterampilan, Anda memerlukan latihan yang disengaja, setidaknya pada tahap awal perolehan keterampilan. Pelajaran yang dipetik.

Dorongan Terakhir

Kembali ke latihan yang disengaja: ada tes lain yang ingin saya coba. Bahasa manusia, termasuk bahasa Inggris, mengikuti kurva hukum kekuasaan yang disebut hukum Zipf: sekumpulan kata yang sangat kecil merupakan sebagian besar penggunaan aktual. Berdasarkan analisis The Brown Corpus (1964), koleksi 1 juta kata dari 500 dokumen bahasa Inggris modern, hanya 135 kata yang menyumbang 50 persen dari semua bahasa Inggris

penggunaan.²¹ Kata "the" itu sendiri menyumbang 7,5 persen, sedangkan "dari" menyumbang 3,5 persen. Anda dapat membawa ide ini lebih jauh: dalam sebagian besar kata, ada kumpulan umum pengelompokan dua dan tiga karakter yang muncul berulang kali, seperti TH, AN, ING, dan NCE. Pengelompokan ini disebut n-gram (atau terkadang n-grafik): "n" adalah variabel yang merupakan singkatan dari jumlah karakter yang Anda kelompokkan. Saya dapat menemukan daftar n-gram yang paling umum dalam sebuah buku berjudul Matematika Kriptologi oleh Robert Edward Lewand (2000). N-gram adalah bidang studi utama di elds seperti kriptografi. Jika Anda dapat mengidentifikasi pola dalam pesan terenkripsi, itu memberi Anda petunjuk tentang kontennya. Dengan membandingkan n-gram dalam pesan yang dikodekan dengan n-gram yang paling umum digunakan dalam bahasa target, kriptografi dapat memecahkan sandi yang kompleks. Mari kita gunakan teori ini. Berikut daftar Lewa dari 2 gram (digram) bahasa Inggris yang paling umum, dalam urutan frekuensi penggunaan:

th, dia, di, en, nt, re, er, an, ti, es, on, at, se, nd, atau, ar, al, te, co, de, to, ra, et, ed, it, sa, em, ro

Dan berikut adalah 3 gram (trigram) yang paling umum, dalam urutan frekuensi penggunaan:

the, dan, tha, ent, ing, ion, tio, for, nde, memiliki, nce, edt, tis, sering, sth, pria

Daftar ini sangat berguna. Semakin baik saya dapat mengetik urutan karakter ini, semakin cepat saya dapat mengetik secara umum. Otak kita sangat pandai dalam hal semacam ini: memori prosedural adalah istilah yang digunakan ilmuwan kognitif untuk keterampilan motorik yang terjadi dalam urutan tertentu. Dengan berlatih n-gram yang paling umum, saya dapat melatih memori prosedural yang terlibat dalam mengetik secara langsung. Untuk melakukan ini, saya mengunduh program gratis bernama Amphetamine,²² yang dirancang untuk latihan semacam ini. Program ini memungkinkan Anda untuk membuat set pelatihan khusus, serta menetapkan ambang batas kinerja tertentu, seperti kata per menit dan tingkat kesalahan.

Saat Anda memulai sesi di Amphetamine, Anda dapat memintaprogrammenghasilkan set latihan dengan berbagai cara. Saya mengatur program untuk menampilkan setiap n-gram tiga kali, untuk menampilkan tiga set item, dan untuk mengulangi urutan tiga kali. Akibatnya, set latihan digram pertama terlihat seperti ini:

Dia berada dalam posisi yang sama seperti dia berada di

Tujuan saya adalah mengetik seluruh urutan pada lebih dari enam puluh WPM dengan akurasi setidaknya 95 persen. Jika saya gagal memenuhi kriteria itu, saya harus mengulangi seluruh urutan. Berlatih dengan cara ini tidak menghibur, tetapi sangat, sangat menarik. Saya mengebor setiap urutan berulang kali sampai saya mendapatkannya, lalu fokus untuk menguasai urutan berikutnya. Setiap hari, ketika saya duduk untuk berlatih, saya mulai dari awal. Peningkatan dari hari ke hari terlihat: urutan yang memakan saya percobaan sehari sebelumnya mengambil satu atau dua percobaan berikutnya. Dalam waktu singkat, saya bisa menyelesaikan seluruh set. Setelah saya menguasai digram, saya beralih ke trigram. Dari sana, saya menemukan daftar kata-kata bahasa Inggris yang paling umum, yang disusun oleh Dr. Peter Norvig. Norvig adalah direktur penelitian di Google. Beberapa tahun yang lalu, dia menerbitkan satu set kata bahasa Inggris yang paling umum berdasarkan "triliun word corpus" Google, yang berisi setiap kata unik yang pernah diindeks oleh laba-laba mesin pencari Google.²³ Niat Norvig dalam menerbitkan data adalah untuk membantu pemrogram membangun utilitas yang berguna seperti alat pemeriksa ejaan, tetapi dalam kasus saya, itu adalah set pelatihan yang sempurna. Saya menarik seratus kata teratas yang paling umum digunakan, menambahkannya ke Amphetamine, dan terus mengebor. Setelah delapan jam latihan tambahan yang disengaja, saya mengikuti tes kecepatan mengetik. Hasilnya, setelah beberapa tes berturut-turut untuk memastikan itu bukan uke: enam puluh WPM, dengan semburan tujuh puluh hingga delapan puluh WPM, dengan akurasi 98 persen. Total waktu yang dihabiskan dalam latihan yang disengaja: dua puluh dua jam.

Misi tercapai!

Tayangan dari Sixty WPM

Saya sangat suka mengetik Colemak. Saya tidak lagi frustrasi. Sebaliknya, saya bertanya-tanya bagaimana saya mengatasi QWERTY begitu lama. Mengetik di QWERTY terasa seperti tangan Anda berputar di seluruh keyboard: banyak gerakan ke segala arah. Tata letaknya sama sekali tidak masuk akal: karakter yang Anda gunakan sepanjang waktu adalah beberapa area keyboard yang paling sulit dijangkau. Colemak, sebaliknya, terasa seperti Anda sedang memutar keinginan Anda untuk membuat kata-kata muncul di layar. Tangan Anda terlihat bergerak lebih sedikit, dan Anda menghabiskan lebih sedikit waktu untuk meraih karakter di baris atas dan bawah. Ini adalah perubahan yang bagus dari QWERTY, dan saya tidak bisa membayangkan untuk kembali.

Meninjau Metode

Mari kita tinjau inti dari metode yang saya gunakan untuk mempelajari kembali cara mengetik sentuhan:

Saya belajar cara mengubah tata letak keyboard saya ke Colemak. Saya membuat loop umpan balik cepat dengan mengatur ulang tombol fisik pada keyboard, jadi jika saya lupa di mana karakter berada, saya dapat melakukannya dengan mudah. Saya menggunakan tutorial mengetik Keyzen untuk mempelajari penempatan karakter dengan sentuhan, mempelajari karakter yang paling banyak digunakan rst. Keyzen memperkenalkan kembali karakter ke set pelatihan saat saya membuat kesalahan, jadi saya menghabiskan sebagian besar waktu saya untuk berlatih karakter dicult sampai akurasi saya meningkat. Saya berlatih selama empat puluh menit setiap malam, tepat sebelum tidur, sehingga otak saya dapat mengkonsolidasikan keterampilan motorik ke dalam memori jangka panjang dengan paling efektif. Setelah saya mencapai kecepatan fungsional (dua puluh WPM), saya beralih ke Tipe Fu, berfokus pada pengetikan kalimat secepat mungkin dengan akurasi lebih dari 99 persen. Ketika saya mencapai empat puluh WPM, saya menggunakan Amphetamine untuk berlatih menggunakan digram dan trigram bahasa Inggris yang paling umum, semakin meningkatkan kecepatan dan akurasi saya. Setelah saya menguasai set n-gram, saya beralih ke pelatihan yang menggunakan kata-kata bahasa Inggris yang paling umum di Amphetamine sampai saya mencapai enam puluh WPM yang dipertahankan dengan akurasi 98 persen, yang terjadi pada tanda dua puluh dua jam.

Ke Mana Saya Pergi Dari Sini

Sekarang saya mengetik enam puluh WPM yang solid di Colemak, saya tidak perlu terus berlatih untuk kecepatan mentah. Pada tingkat ini, saya dapat menulis secepat yang saya butuhkan. Karena kecepatan mengetik saya tidak lagi menjadi faktor pembatas dalam output saya, latihan kecepatan bukan lagi prioritas terbesar saya. Mengetik cepat adalah keterampilan tersendiri. Beberapa juru ketik tercepat di dunia dapat mencatat lebih dari 180 WPM pada keyboard normal, tetapi peningkatan kecepatan pengujian tidak selalu sama dengan peningkatan kecepatan menulis atau pengkodean. Tes ini selalu melibatkan pengetikan apa yang Anda lihat di layar, jadi keterampilan utama yang dipraktikkan oleh juru ketik kecepatan melibatkan melihat lebih jauh ke depan dalam teks dan menyimpannya dalam memori jangka pendek cukup lama bagi pengetikan saat itu. Meskipun saya ingin bisa menulis prosa nished di lebih dari 180 WPM, itu bahkan melampaui juru ketik tercepat sekalipun. Mengetik lebih sulit dan lebih lambat ketika Anda harus membuat apa pun yang Anda letakkan di halaman. Namun, saya dapat memanfaatkan untuk terus menurunkan tingkat kesalahan saya. Dengan semangat itu, saya terus berlatih kata-kata umum, digram, dan trigram dengan Amphetamine, dan berlatih menggunakan pinkienger kiri saya untuk koreksi kesalahan. Dengan menurunkan tingkat kesalahan saya, myeciency akan meningkat, dan kecepatan saya kemungkinan besar akan meningkat bersamanya. Yang paling mengejutkan saya tentang belajar Colemak adalah betapa mudahnya untuk menimpa hampir dua puluh tahun pengalaman mengetik sentuh sebelumnya menggunakan QWERTY. Saya berasumsi bahwa dua dekade memori otot akan memakan waktu lebih dari dua puluh jam untuk menggantinya. Saya salah. Otak kita lebih mudah diubah daripada yang kita pikirkan.



7Pe rgi

Pelajaran: Jelajahi, Lalu Putuskan

Go menggunakan bahan dan konsep yang paling mendasar: garis dan lingkaran, kayu dan batu, hitam dan putih, menggabungkannya dengan aturan sederhana untuk menghasilkan strategi halus dan taktik kompleks yang mengejutkan imajinasi.—
IWAMOTO KAORU, PEMAIN GO PROFESIONAL 9-DAN

• • •

Untuk gambar, video, dan komentar tambahan tentang bab ini,
visithttp://first20hours.com/go.

Aku adalah pedang dalam kegelapan, penjaga di dinding. Malam dingin: salju turun. Di belakang saya, obor menerangi cahaya di atas rekan senegaraku. Kami berdiri di benteng kastil, menonton. Di sisi jauh jembatan kastil, sekelompok tentara musuh berkumpul tepat di luar jangkauan ketapel kami, bersiap untuk menyerang. Prajurit, pencuri, dan penjaga hutan mengasah senjata mematikan. Ahli nجوم memanggil pasukan makhluk mayat hidup yang menjijikkan. Para insinyur sedang menyiapkan mortir dan domba jantan: senjata ampuh yang dirancang untuk menerobos gerbang kita. Gerombolan ini bermaksud untuk mencuri harta benda paling berharga di alamku: bola bercahaya dengan kekuatan besar. Mereka tidak akan menerimanya. Tidak di mywatch.

Kami sudah siap dengan baik. Di kedua sisi tempat saya berdiri, tentara mengawaki balista dan gerobak panah, siap untuk melepaskan muatan mematikan mereka pada pasukan penyerang. Di antara pertahanan elemental sedang bersiap untuk menghujani kembali dan petir pada pasukan musuh. Penjaga dan pesona kami telah memasang rune dan perisai reaktif di sepanjang perimeter, memberi kami perlindungan dari panah dan mantra, setidaknya untuk sementara waktu. Kami menunggu. Salju turun, dan cahaya obor menari di baju besi saya yang dipoles. Kami tidak akan jatuh. Kami tidak akan putus. Kami akan bersemangat, dan kami akan menang. Dari kegelapan, sebuah batu besar muncul, ying tinggi di atas pasukan musuh. Batu itu menabrak gerbang kastil dan hancur, menghancurkan kayu yang diperkuat. Musuh mengaum, lalu menyerang melintasi jembatan. Dalam sekejap, pedang dan perisaiku ada di tangan. Pertempuranku bergema di setiap batu, membawa kekuatan dan kemarahan kepada setiap bek di dinding." **UNTUK KEADILAN YANG BESAR!"**

Singkirkan senjatamu, maksudku kamu tidak ada salahnya

Memimpin pasukan dalam perang pengepungan skala besar adalah cara yang menyenangkan untuk menghabiskan Sabtu malam. Saya bukan penggemar menonton televisi, film, atau olahraga. Sebaliknya, Anda mungkin akan mendatangkan saya bermain video game seperti World of Warcraft atau Guild Wars 2 selama satu atau dua jam. Saya menikmati permainan: sebaiknya tantangan yang melibatkan merapal mantra, melawan monster, dan mengecoh pemain musuh. Sejak usia muda, saya menyukai cerita epik tentang penyihir, prajurit, dan pencarian heroik, dan video game memungkinkan untuk menjadi karakter dalam petualangan ini, setidaknya untuk sementara waktu.¹

Generasi saya adalah yang pertama tumbuh dengan video dan game komputer yang imersif. Dimulai dengan Atari dan Nintendoconsole awal, game telah tumbuh dalam detail dan kompleksitas. Sekarang, dimungkinkan untuk mengalahkan naga Internet dan pemain lain secara real time dengan sekutu dari seluruh dunia. Bagian dari apa yang menarik bagi saya tentang game seperti ini adalah keterampilan yang diperlukan untuk bermain dengan baik. Siapa pun dapat membuat karakter, tetapi menarik ke dalam pertempuran dan muncul sebagai pemenang, Anda harus tahu apa yang Anda lakukan. Kami telah menempuh perjalanan yang sangat jauh dari Mario melompat ke musuh dan menembakkan bola ulang sesekali. Sekarang, karakter video game biasanya memiliki hampir seratus kemampuan potensial. Untuk bermain dengan baik, Anda harus tahu apa kemampuan itu, kapan menggunakannya, dan bagaimana menyesuaikan karakter Anda untuk kemampuan terbaik. Ada banyak yang harus dipelajari. Kemampuan mana yang paling merusak, atau melindungi Anda dari bahaya? Apa yang bisa dilakukan monster musuh? Apa strategi terbaik saat melawan pemain manusia lain? Semakin banyak saya bermain, semakin baik saya menjadi. Itu sebabnya menyenangkan. Video game, secara historis, benar-benar baru. Permainan keterampilan dan peluang, bagaimanapun, telah menjadi bagian dari pengalaman manusia selama ribuan tahun.

Permainan Papan Strategis Tertua di Dunia

Beberapa waktu yang lalu, saya menemukan permainan yang menarik. Dibandingkan dengan apa yang biasa saya mainkan, permainannya tenang, hampir tenang. Namun, di bawah permukaan, ada drama yang tinggi. Papan adalah peta perang besar, dan para pemainnya adalah jenderal, bertarung satu sama lain untuk memenangkan supremasi tertinggi. Go adalah game tertua di dunia yang masih dimainkan dalam bentuk aslinya. Berdasarkan catatan sejarah, Go berasal dari Tiongkok kuno, dan telah ada di bawah aturan yang ditetapkan saat ini setidaknya selama tiga ribu tahun,

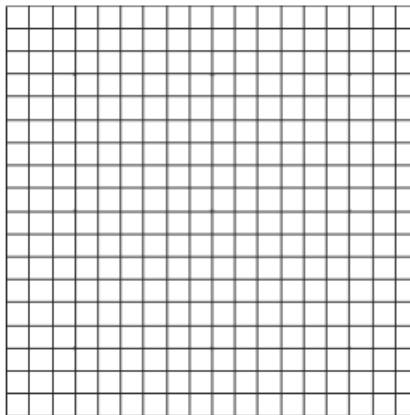
dengan beberapa sejarawan memperkirakan lebih dari empat ribu tahun. Jika ada indikasi kualitas, Go memiliki banyak hal untuk itu. Nama Cina untuk Go adalah weiqi. Wei () berarti "mengelilingi" dan qi () berarti "permainan papan." Bersama-sama, karakter-karakter ini menjadi weiqi (, sederhana:), yang berarti "permainan mengelilinginya." Itu adalah deskripsi yang bagus dan sederhana tentang kondisi kemenangan permainan: mengelilingi pemain lawan. Go diperkenalkan di Eropa dan Amerika melalui Jepang, jadi bahasa Inggris Go adalah implikasi dari kata Jepang untuk permainan, igo(). Apakah Anda menyebut game sebagai Go, weiqi, igo, baduk, atau istilah lainnya, permainannya sama.

Seni Perang

Di Barat, catur adalah permainan papan strategis paling populer, jadi kita dapat menggunakannya sebagai kontras. Di permukaan, game tersebut memiliki banyak kesamaan. Setiap permainan dimainkan oleh dua pemain, Hitam dan Putih. Permainan berlangsung di papan persegi. Hitam bergerak rst. Para pemain bergiliran bergerak sampai permainan selesai. Anda dapat menganggappermainan sebagai kerucut militer, di mana para pemainnya adalah jenderal lawan. Di situlah kesamaan berakhir. Catur dimainkan di papan yang terdiri dari delapan kotak kali delapan kotak, dengan total enam puluh empat kotak. Setiap bidak catur menempati satu kotak. Papan itu menyerupai battleeld, dan potongan-potongannya adalah tentara. Inilah tampilan papan catur di awal permainan:



Go, di sisi lain, dimainkan di papan yang terdiri dari sembilan belas garis vertikal yang berpotongan dengan sembilan belas garis horizontal. Batu-batu dimainkan di persimpangan, bukan di alun-alun. Akibatnya, ada 360 persimpangan yang tersedia untuk dimainkan di papan Go, 5.625 kali lebih banyak daripada di papan catur. Inilah tampilan papan Go di awal permainan:



Arah pemberitahuan? Catatan hanya papan Go IS Permainan yang jauh lebih besar dan bodoh dimulai tanpa batu di papan. Di Go, Batu Adalah

umumnya ditambahkan selama permainan. Dalam catur, bidak dihapus saat ditangkap. Perhatikan bahwa goban (papan) berbentuk persegi. Titik-titik kecil di papan tulis (disebut "titik bintang") sangat simetris. Poin-poin itu penting, dan kami akan kembali sebentar lagi. Dalam catur, ada enam jenis bidak yang berbeda, dan setiap bidak memiliki aturan dan kemampuan khusus. Pion selalu bergerak lurus ke depan, kecuali ketika mereka bergerak untuk menangkap bidak lain. Uskup dapat bergerak secara diagonal, tetapi tidak dapat bergerak secara vertikal atau horizontal. Benteng dapat bergerak secara horizontal atau vertikal, tetapi tidak secara diagonal. Ksatria dapat melompati bidak lain, tetapi harus memindahkan dua kotak secara horizontal dan satu persegi secara vertikal, atau dua kotak secara vertikal dan satu persegi secara horizontal. Ratu dapat bergerak secara horizontal, vertikal, atau diagonal, tetapi tidak dapat melompat bidak lainnya. Raja dapat bergerak ke segala arah, tetapi hanya dapat bergerak satu kotak pada satu waktu, kecuali jika ia melompat menggunakan gerakan khusus yang disebut "kastil". Ada banyak hal yang perlu diingat. Setiap gerakan Go, sebaliknya, adalah sama: sebuah batu ditempatkan di persimpangan. Kecuali batu "ditangkap" nanti dalam permainan, stonelaid adalah batu yang dimainkan. Bidak catur ditangkap oleh salah satu bidak lawan, seperti pertarungan tunggal pada battleeld. Benteng itu memukul kepala uskup dengan gada besar, dan uskup keluar dari permainan. Batu Go ditangkap ketika dikelilingi di semua sisi oleh batu lawan. Ketika pasukan Black dikepung di semua sisi oleh White, batu-batu Black menyerah, dan disandera. Permainan catur profesional biasanya berisi 30 hingga 40 gerakan. Dalam permainan Go, 30 langkah pertama atau lebih dianggap sebagai gerakan pembuka: permainan akhir dimulai sekitar langkah 100. Selesaikan permainan Goregularly 250 gerakan teratas. Dalam segala hal, skala Go jauh lebih besar daripada catur. Jika setiap permainan catur adalah pertempuran, goban adalah peta perang besar-besaran. Jadi bagaimana Anda memainkan Go, tepatnya? Mari kita gali masuk.

Aturan Permainan

Percaya atau tidak, Go hanya memiliki tujuh aturan utama, dan kami telah membahas dua di antaranya:

1. Batu dimainkan di persimpangan.
2. Hitam dan Putih bergiliran meletakkan batu di goban.

Aturan ve berikutnya menyangkal perkembangan permainan dan kondisi kemenangan:

3. Batu "ditangkap" ketika dikelilingi di semua sisi oleh batu lawan.
4. Memainkan batu yang segera ditangkap ("bunuh diri") dilarang.
5. Mengulangi urutan gerakan yang sama berulang kali dalam lingkaran innite (situasi yang disebut ko) dilarang.
6. Permainan berakhir ketika para pemain kehabisan batu, satu pemain kebobolan permainan, atau kedua pemain mengoper.
7. Pemain yang mengelilingi wilayah terbanyak di papan di akhir permainan menang.

Dalam permainan kompetisi, beberapa aturan lagi diperkenalkan untuk menghilangkan ambiguitas (seperti "apa yang dimaksud dengan loop berulang?"), menentukan metode penilaian, dan mencegah seri. Jika tidak, itu adalah aturannya. Cukup sederhana, bukan? Mempelajari aturan Go itu mudah: hanya membutuhkan beberapa menit. Aturan itu sendiri tidak rumit. Namun, ada pepatah Go yang sangat lama: "beberapa saat untuk belajar, seumur hidup untuk dikuasai."

Gabungkan aturan sederhana ini dengan papan sederhana dan batu sederhana, dan Anda mendapatkan kerumitan yang menakjubkan.

Ukuran Alam Semesta

Sementara aturan barok Catur hanya bisa diciptakan oleh manusia, aturan Goare begitu elegan, organik, dan sangat logis sehingga jika bentuk kehidupan cerdas ada di tempat lain di alam semesta, mereka hampir pasti memainkan Go.—EDWARD LASKER, GRANDMASTER CATUR DAN PENULIS GO AND GO MOKU

Katakanlah kita ingin membuat program komputer yang memainkan Gointelligently, mirip dengan Deep Blue, intelijen artisial yang terkenal

program yang mengalahkan Garry Kasparov, juara dunia bertahan, pada tahun 1996. Biasanya, komputer mengecoh pemain manusia melalui kekuatan komputasi belaka: mereka menghitung semua kemungkinan gerakan legal di papan, kemudian memilih gerakan yang memiliki probabilitas keberhasilan matematis tertinggi, berdasarkan perpustakaan data pastgame yang luas. Di papan catur, jenis komputasi ini tidak mudah, tetapi itu mungkin. Ada enam puluh empat kotak, dan gerakan setiap bagian dibatasi oleh aturan khusus. Karena setiap bagian hanya dapat bergerak dengan cara tertentu, program hanya perlu mempertimbangkan sejumlah kecil pilihan. Dalam permainan Go, pemain aktif dapat meletakkan batu di persimpangan terbuka mana pun di papan. Gim ini dimulai dengan 360 kemungkinan pilihan, jadi sejak awal, program AI edgling kami memiliki lebih banyak analisis yang harus dilakukan. Mari kita lakukan perhitungan cepat. Berapa banyak kemungkinan urutan moves yang dapat dimainkan di papan Go, dengan asumsi itu adalah awal permainan dan tidak ada pemain yang menangkap batu lawan? Berikut perhitungannya:

$$360 * 359 * 358 * 357 * 356 = 5.880.282.488.640$$

Itu lebih dari 5,8 triliun kemungkinan urutan, dan itu hanya gerakan rstve. Matematika menjadi gila sangat, sangat cepat. Ingat ketika saya menyebutkan bahwa game Go biasanya bertahan 250 gerakan? Tergantung pada asumsi Anda, ada sekitar 2,08 kali 10170 urutan gerakan legal pada Goboard berukuran sembilan belas kali sembilan belas. Jika matematika itu benar, ada lebih banyak kemungkinan permainan hukum Godaripada ada partikel subatomik di alam semesta yang diketahui. Ini adalah kepastian matematis bahwa setiap pertandingan Go yang pernah dimainkan belum pernah dimainkan sebelumnya dalam sejarah

alam semesta, bahkan jika Anda bersedia melemparkan kemungkinan bahwa ada miliaran peradaban alien canggih di suatu tempat di luar sana yang juga kebetulan bermain Go.² Mengingat teknologi saat ini, bahkan komputer paling canggih yang menjalankan algoritma brute-force terbaik akan membutuhkan waktu sekitar empat ratus tahun untuk menghitung satu gerakan optimal, dengan asumsi program menyelesaikan perhitungan setiap beberapa milidetik. Pikiran = tertiu. Game ini sangat besar.

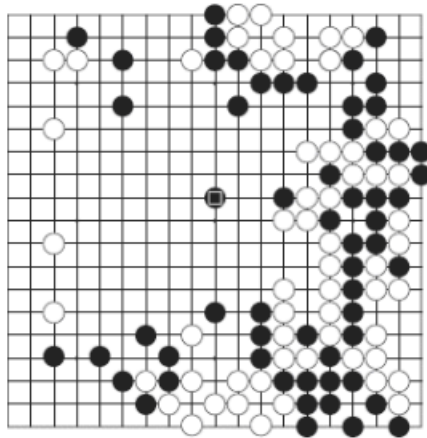
Jadi bagaimana pemain (dan komputer) bermain?

Jika pemain manusia mengandalkan analisis brute-force untuk memainkan Go, mereka akan menjadi gila. Mereka jelas tidak: pemain manusia yang terampil dapat mengidentifikasi gerakan terbaik di papan dalam beberapa detik. Bagaimana mereka melakukannya? Pemain Go mengandalkan pengenalan pola untuk mengidentifikasi gerakan bernilai tinggi. Pemain menggunakan banyak kata yang memunculkan intuisi, seperti "bentuk" dan sente (inisiatif). Pemain Go terbaik tampaknya mengandalkan geometri, keindahan estetika, dan emosi seperti halnya pada logika dan analisis yang ketat. Itu masuk akal: otak manusia tidak dilengkapi dengan baik untuk mengolah angka brute-force, tetapi luar biasa dalam pengenalan pola. Dengan memperhatikan pola pada batu di papan, serta pola bagaimana batu dimainkan, pemain Go yang terampil mampu membaca situasi saat ini, kemudian melakukan langkah terbaik: semuanya dalam waktu kurang dari empat ratus tahun. Bahkan yang lebih mengesankan: pemain terbaik dapat mengantisipasi bagaimana batu akan dimainkan di masa depan, seringkali tiga puluh hingga empat puluh langkah ke depan. Jika Anda bermain dengan seorang profesional, mungkin akan terasa seperti mereka membaca pikiran Anda.

Permainan Aktual

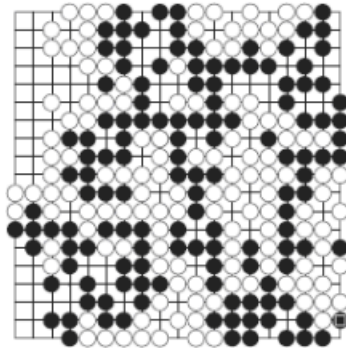
Arah antara batu yang dimainkan di satu persimpangan daripada di adjésentnigbor tidak penting bagi United. Master of the Master of Go, Thog, Cis IT SL Thedeerance Between a Over and a Cinder Block.—Dave Laurie, Penulis The Challenge of the OF Go: Esoteric Granddaddy Offboard Games

Inilah tampilan permainan Go yang sedang berlangsung:



Ini adalah diagram permainan nyata: salah satu permainan paling terkenal dalam sejarah Go.³Pada tahun 1846, Shusaku dari House Honinbo, seorang 4-dan berusia tujuh belas tahun, diundang untuk memainkan Gennan Inseki, seorang 8-dan. Inseki adalah kepala House Inoue, salah satu dari empat sekolah Go profesional utama di pertengahan abad kesembilan belas Jepang. Shusaku menerima kehormatan itu, dan pertandingan itu menarik penonton. Tidak ada yang menyangka Shusaku akan menang, tetapi dia cukup kuat untuk memberikan kekuatan. Setelah memainkan permainan pendek di mana Inseki memberi Shusaku handicap dua batu, jelas Shusaku tidak membutuhkannya. Inseki mengakui untuk memulai game kedua tanpa batu handicap: even odds.

Shusaku, sebagai penantang, mengambil Black. Permainan awal tidak berhasil, kecuali kesalahan kecil oleh Shusaku dalam pertukaran di sisi kanan bawah papan. Seratus dua puluh enam giliran ke dalam permainan, Inseki memimpin, seperti yang diharapkan. Lagi pula, siapa yang bisa terbaik tuannya? Permainan Shusaku berikutnya mengubah permainan. Anda dapat melihat sendiri batunya: itu adalah batu Hitam tepat di atas tengah papan yang ditandai dengan persegi. Setelah gerakan Shusaku, seorang penonton melihat sesuatu yang aneh: Inseki'sears berwarna merah. Tuannya marah. Langkah Shusaku di 127 bagus: sangat, sangat bagus. Batu tunggal tengah secara bersamaan memberikan dukungan pada batu Shusaku di bagian atas, kanan, dan bawah papan sambil membangun inuence baru di sebelah kiri. Langkah itu adalah keseimbangan sempurna antara oense dan pertahanan. Inseki dalam masalah, dan dia tahu itu. Dengan satu batu, Shusaku mampu menghancurkan seluruh papan. Permainan berlanjut, dengan Inseki tiba-tiba menggigit untuk mengontrol. Inilah permainan yang terlihat di langkah 325:

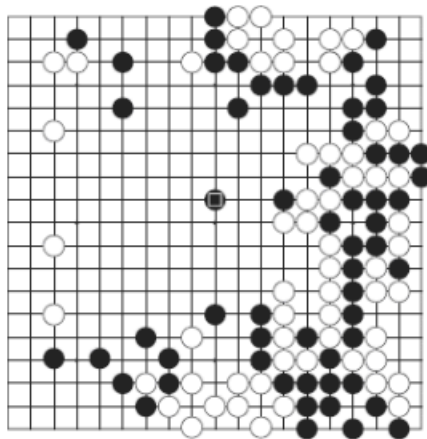


Luangkan waktu sejenak untuk menelaah papan tulis. Area mana yang dikelilingi oleh Hitam, dan mana yang dikelilingi oleh Putih? Siapa yang mengelilingi lebih banyak wilayah di goban?

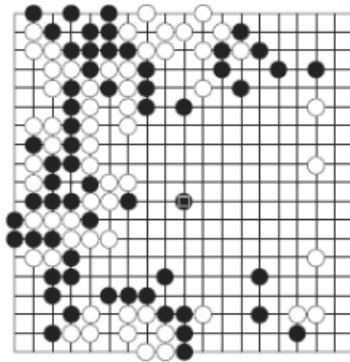
Inseki mengaku kalah, dan Shusaku menang dengan selisih dua poin. Permainan ini akan tercatat dalam sejarah sebagai Ear Reddening Game, dan memindahkan 127 Ear Reddening Move." Invincible Shusaku" kemudian menjadi salah satu pemain Go paling terkenal dalam sejarah, terkenal karena sembilan belas kemenangan permainan kastil berturut-turut, yang diselenggarakan setiap tahun oleh Shogun. Dia meninggal pada tanggal 7 September 1862, pada usia tiga puluh tiga tahun, setelah merawat korban wabah kolera.

Pengenalan Pola

Ada sesuatu yang bisa kita pelajari dari diagram ini. Inilah Langkah Ear Reddening lagi:



Game Go biasanya disajikan dari sudut pandang Black. Kami melihat papan seperti yang dilihat Shusaku selama pertandingan. Berikut adalah bagaimana Inseki (Putih) memandang papan, rotasi 180 derajat:

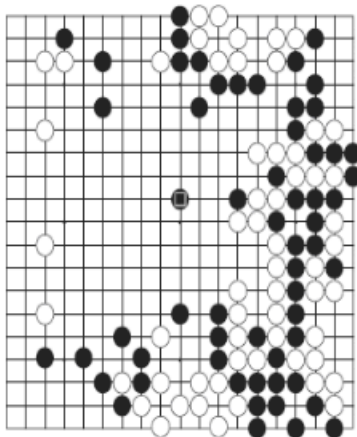


Dalam catur, hanya ada dua perspektif utama: Hitam duduk di seberang dari Putih, dan kedua pemain memulai permainan dengan semua bidak mereka di sisi papan masing-masing. Catur tidak pernah dimainkan dengan bidak Putih dan Hitam yang disusun di sisi kiri dan kanan papan. Selain itu, gerakan catur memiliki pola umum: jauh dari pemain, ke arah sisi papan lawan. Akibatnya, pemain catur dapat belajar mengenali pola spesifik gerakan di papan. Berikut adalah pola terkenal, yang disebut King's Gambit:

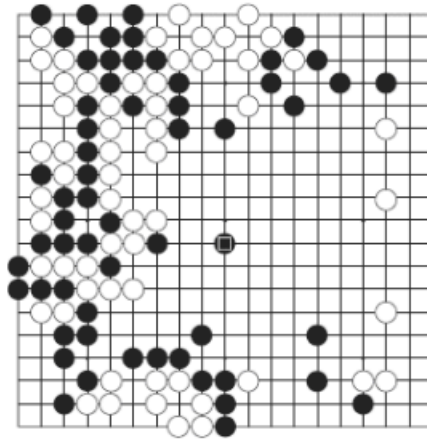


Sebagian besar pelatihan untuk menjadi grandmaster catur melibatkan menghafal pola umum yang muncul dalam permainan. Si

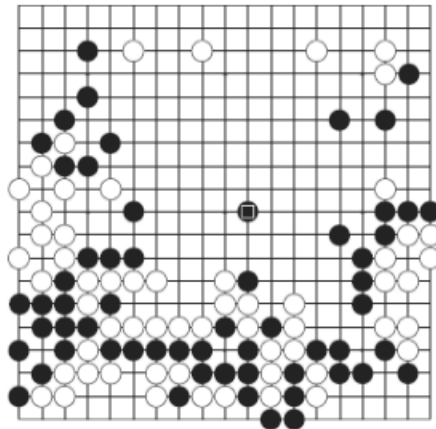
Pemain catur terbaik dapat melihat pola berkembang di papan dalam sekejap, karena polanya selalu terlihat sama. Jika Anda mencoba memainkan aKing's Gambit, seorang grandmaster akan segera menyadarinya, dan akan tahu persis bagaimana meresponsnya. Meskipun pengenalan pola dalam catur tidak mudah, ini membantu bahwa pemain tidak perlu melihat papan yang diputar sembilan puluh derajat ke kiri atau kanan, yang membuat pola jauh lebih mudah dipelajari. Sebaliknya, karena goban sangat simetris, tidak ada sisi papan yang memiliki arti khusus. Hasilnya, Anda dapat melihat goban dari salah satu dari empat sisi papan. Di awal permainan Go, setiap batu handicap ditempatkan oleh Hitam pada titik bintang, titik-titik hitam kecil itu, dalam pola simetris. Jika tidak, papan kosong. Batu Go dapat ditempatkan di persimpangan kosong kapan saja. Apemain dapat bermain di bagian atas papan satu putaran, dan bagian bawah papan berikutnya. Mereka bisa bermain di sisi kiri, lalu bermain di sebelah kanan. Tidak ada arah permainan yang universal dan dapat diprediksi seperti yang ada dalam catur. Mari kita kembali ke Gerakan Kemerahan Telinga. Berikut perspektif Shusaku:



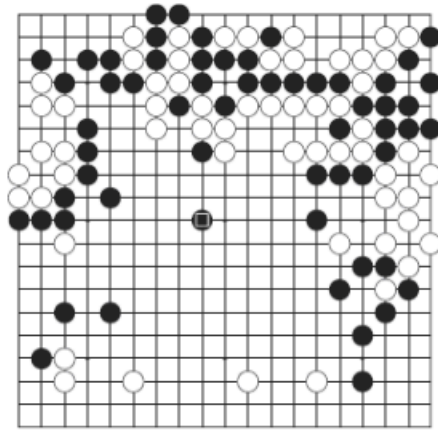
Sekali lagi, inilah perspektif Inseki:



Sekarang, inilah perspektif dari kanan Shusaku, atau kiri Inseki:



Dan kiri Shusaku, atau kanan Inseki:

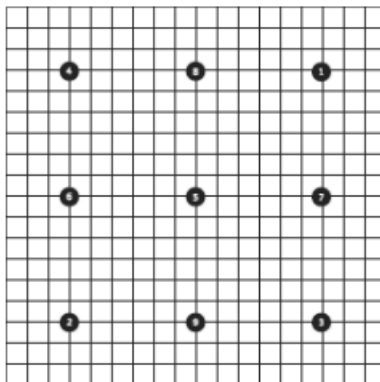


Sifat simetris goban membuat pengenalan pola menjadi lebih sulit daripada biasanya. Keempat diagram dari Gerakan Kemerahan Telinga ini persis sama dari sudut pandang strategis, meskipun mereka terlihat benar-benar berbeda bagi mata yang tidak terlatih. Jika mau, Anda dapat memutar papan Go sembilan puluh derajat setelah setiap gerakan tanpa secara dramatis merusak permainan. Pemain Go tidak dapat mengandalkan teknik menghafal permainan yang lebih mudah yang diandalkan pemain catur untuk mengidentifikasi pola yang berkembang selama permainan. Setiap pola yang dipelajari pemain Go perlu dikuasai dari empat perspektif, bukan hanya satu. Itu sebabnya pemain Go sangat mengandalkan intuisi. Gim ini terlalu besar untuk dihafal atau dihitung, sehingga pemain yang terampil melatih diri mereka untuk melihat pola yang berkembang pada tingkat yang jauh lebih tinggi: bentuk umum, arah gerakan, dan kesan kekuatan dan kelemahan.

Gunakan perasaan Anda ...

[Go membutuhkan] taktik prajurit, ketepatan matematikawan, imajinasi seniman, inspirasi penyair, ketenangan filsuf, dan kecerdasan terbesar.

Salah satu artikel awal yang saya baca tentang Go membuat argumen yang menarik: menguasai permainan tidak benar-benar tentang bersaing dengan atau mendominasi lawan. Ini tentang menguasai diri sendiri. Permainan peluang telah ada sejak awal peradaban manusia. Saat Anda melempar dadu, Lady Luck (atau lebih tepatnya, MamaPhysics) memutuskan siapa yang menang. Keterampilan bukanlah faktor. Permainan keterampilan, sebaliknya, biasanya berfokus pada mengecoh atau mengungguli pemain lain. Siapa yang menunjukkan bakat dan kontrol paling banyak? Siapa yang bisa mencari peluang terbesar rst? Siapa yang lebih baik mengeksploitasi kelemahan lawannya? Catur termasuk dalam kategori ini: menjadi hebat melibatkan penguasaan taktik dan membaca lawan Anda. Go adalah permainan unik dalam hal ini: pengenalan handicapstones adalah cara bawaan untuk sengaja membuat pemain kuat lebih lemah. Jika lawan Anda memulai permainan dengan beberapa batu di papan di posisi strategis kunci, itu secara dramatis mempengaruhi cara Anda bermain. Di sinilah batu handicap ditempatkan:



Dalam permainan Go yang cacat dengan baik antara dua pemain dengan keterampilan yang kira-kira sama, setiap pemain harus menang sekitar setengah waktu. Kalau

Satu pemain mendominasi yang lain, di permainan berikutnya, pemain yang kalah dimulai dengan satu atau dua batu tambahan di papan. Karena fitur handicapping ini, Anda dapat menganggap Go sebagai permainan yang Anda mainkan melawan diri sendiri. Tentu, Anda membuat keputusan berdasarkan tindakan lawan Anda, tetapi memenangkan permainan bukanlah satu-satunya tujuan. Saat keterampilan Anda meningkat, jumlah batu handicap yang Anda butuhkan menurun. Saat Anda memainkan lawan yang kuat, Anda akan dapat melakukan pertempuran dengan cara yang seimbang. Akhirnya, Anda akan memberikan batu kepada lawan yang lebih lemah. Go history membawa aspek penguasaan diri dari permainan menjadi ekstrem: belajar bermain dengan baik membutuhkan penguasaan pikiran dan emosi Anda. Pemain tingkat lanjut dapat belajar melihat tanda-tanda di batu yang mengungkapkan kondisi mental dan emosional lawan mereka. Menurut cerita, master kuno dapat membaca transkrip permainan dan mengidentifikasi kapan setiap pemain merasakan kemarahan, kebingungan, iri, atau keserakahan, serta menentukan saat yang tepat "ketika pelayan masuk untuk menyajikan teh."

Ninja Batu

Selama berabad-abad, sistem handicapping Go berkembang menjadi sistem peringkat: perbedaan peringkat menentukan berapa banyak batu yang dapat ditempatkan pemain peringkat lebih rendah di awal permainan. Percaya atau tidak, sistem peringkat "sabuk" dalam seni bela diri berasal dari peringkat Go. Saat pemain meningkat dalam keterampilan, dia naik peringkat. Pemula mulai dari peringkat 35 kyu (mirip dengan sabuk putih). Saat pemain semakin kuat, peringkatnya menurun hingga mencapai 1 kyu. Peringkat berikutnya dari 1 kyu adalah 1 dan, setara dengan sabuk hitam. Dari sana, peringkat meningkat hingga pemain mencapai 9 dan, peringkat resmi tertinggi; 10 dan adalah gelar kehormatan yang disediakan untuk pemenang turnamen dunia teratas.

Menentukan peringkat pemain dapat dilakukan dengan beberapa cara. Kompetisi rστιs: jika seorang pemain dapat menang sebagian besar waktu dengan cara yang seimbang melawan 12 pemain kyu, tetapi kalah sebagian besar waktu melawan 8 pemain kyu, kekuatannya mungkin berada di kisaran 10 kyu. Semakin banyak permainan yang diselesaikan pemain dalam kompetisi peringkat, semakin akurat peringkatnya. Cara lain untuk memperkirakan peringkat adalah dengan menyelesaikan masalah Go: teka-teki terstruktur yang menyajikan situasi, lalu meminta pemain untuk menentukan langkah terbaik yang mencapai hasil tertentu, seperti "menangkap kelompok Hitam" atau "Putih untuk menyelamatkan." Saat pemain naik peringkat, kemampuannya untuk memecahkan masalah dengan benar juga meningkat. Karena Go telah ada begitu lama, perpustakaan besar masalah Go tersedia untuk dipelajari. Masalah-masalah ini diberi peringkat berdasarkan fakultas: pemain 20 kyu akan berjuang dengan masalah 10 kyu, sedangkan pemain 1 dan akan berpikir masalah 10 kyu mudah dan jelas. Kiseido, penerbit khusus buku Go, menerbitkan serangkaian masalah peringkat yang sangat praktis, berjudul *Graded Go Problems for Beginners*, yang sangat berguna untuk tujuan ini. Keberadaan soal peringkat membuat Go menjadi game yang bisa kamu pelajari sebanyak yang kamu mainkan. Menyelesaikan masalah dalam buku-buku ini adalah cara yang baik untuk melatih keterampilan Go sendiri, serta memperkirakan kekuatan relatif Anda. Saya siap untuk memulai. Apa yang perlu saya praktikkan?

Bersiap

It's difficult to play a game of Go without a board and stones, so I picked up a nice set from Yellow Mountain Imports,⁴ a company that distributes Go products from all over the world in the United States. Go boards and stones vary widely in price and quality. Inexpensive boards and simple glass stones can be acquired for a few dollars, so a basic game set is easy to obtain.

Di ujung lain spektrum, papan berkualitas unggul yang terbuat dari kayu kayane, bahan tradisional yang digunakan di Jepang, dijual seharga puluhan ribu dolar. Demikian juga, Anda dapat membeli batu yang terbuat dari batu tulis dan cangkang asli, tetapi harganya mahal. Setelah melakukan beberapa penelitian, saya memilih papan shin-kaya (tiruan, biasanya cemara putih) yang bagus dan satu set batu Yunzi, yang dibuat oleh perusahaan dengan nama yang sama di Cina. Perusahaan menganggap komposisi batu mereka sebagai rahasia dagang: batu-batu itu terasa bagus dan kokoh, menciptakan suara yang sangat memuaskan saat Anda meletakkannya di papan, dan harganya terjangkau. Batu-batu itu dilengkapi dengan mangkuk kayu tradisional, yang ditempatkan di samping goban saat bermain. Selain membeli goban dan batu, saya juga mengambil beberapa buku Go untuk pemula, antara lain:

Go: Pengantar Lengkap untuk Game oleh Cho Chikun (2010) Buku Kedua Go: Apa yang Perlu Anda Ketahui Setelah Anda Mempelajari Aturan oleh Richard Bozulich (1998) Cara Tidak Bermain Go oleh Yuan Zhou (2009) Pelajaran dalam Dasar-dasar Go oleh Toshiro Kageyama (1996) Teori Pembukaan Menjadi Mudah oleh Otake Hideo (1992)

Saya menemukan buku-buku ini melalui situs web yang sangat berguna bernama Sensei's Library,⁵ yang memiliki ratusan halaman informasi dan komentar tentang sejarah dan teknik Go. Satu halaman adalah daftar besar opini tentang buku Go terbaik yang tersedia, disusun oleh pemain tingkat lanjut, yang merupakan emas padat pada tahap proses pembelajaran ini. Saya membaca seluruh halaman, yang mencantumkan hampir seratus buku. Inilah cara saya memilih buku mana yang akan dibaca rst: Go: A Complete Introduction to the Game dengan mudah menjadi rekomendasi teratas untuk pemula, jadi itu tidak perlu dipikirkan lagi. Demikian juga, The Second Book of Go was sangat direkomendasikan sebagai teks awal tentang strategi. Buku

mengasumsikan Anda sudah mengetahui aturan dasar, sehingga menghabiskan lebih banyak waktu pada teknik dasar. *How Not to Play Go* adalah contoh inversi, yang saya senang melihatnya: Anda dapat belajar banyak tentang cara melakukan sesuatu dengan baik dengan mempelajari kesalahan umum. Untuk sebagian besar keterampilan, Anda harus meneliti inversi sendiri, tetapi dalam hal ini, ada seluruh buku yang disusun oleh seorang ahli. Itu bagus! *Lessons in the Fundamentals of Go*, sering disebut "The Yellow Book," adalah buku yang dikutip oleh sebagian besar pemain tingkat lanjut sebagai teks tunggal yang membantu mereka meningkatkan permainan mereka secara dramatis. Kelihatannya terlalu maju sebagai teks awal, jadi saya akan membacanya setelah saya mempelajari dasar-dasarnya. Akhirnya, saya memilih *Opening Theory Made Easy* karena jelas, bahkan dari analisis dasar saya, bahwa pembukaan permainan sangat penting. Karena goban kosong di awal permainan selain dari batu handicap apa pun, tiga puluh hingga empat puluh batu pertama yang dimainkan menciptakan struktur yang sangat merusak sisa permainan. Jika Anda tidak tahu cara memainkan pembukaan dengan benar, dan lawan Anda tahu, Anda mungkin akan kalah, membuat pembukaan menjadi mata pelajaran yang layak dipelajari sejak awal.

Ikuti Kelinci Putih

Selain itu, teknologi modern telah membuat latihan Go sedikit lebih mudah. *SmartGo*, sebuah program yang tersedia untuk iPhone dan iPad, mencakup program AI bawaan yang sangat bagus, database masalah Go berperingkat, dan game sejarah beranotasi untuk dipelajari. *SmartGo* dirancang untuk memanfaatkan layar sentuh perangkat, sehingga Anda dapat meletakkan batu di "papan" dengan menyentuh persimpangan yang sesuai. Ini membuat menyelesaikan masalah Go menjadi lebih mudah. Alih-alih membayangkan solusinya, lalu mencari solusi dengan tangan di bagian belakang teks cetak, program ini memberi Anda umpan balik instan. Loop umpan balik yang cepat ini membuatnya banyak, banyak

Masalah yang lebih mudah dipraktikkan, terutama masalah yang membutuhkan lebih dari satu langkah untuk dipecahkan. Saya sudah siap. Saya memiliki semua yang saya butuhkan untuk belajar cara bermain. Hanya ada satu hal lagi yang harus saya lakukan ... pastikan bahwa saya menghabiskan waktu saya bermain Go alih-alih memainkan yang lain.

Menghilangkan Gangguan

Go tidak semenarik video game penuh aksi. Belajar Go akan membutuhkan waktu dan konsentrasi. Saya sudah memainkan game lain, tetapi saya memiliki waktu luang yang sangat terbatas. Jika saya ingin maju di Go secepat yang saya bisa, saya harus fokus. Itu berarti saya perlu menghilangkan potensi gangguan. Ancaman langsung terbesar adalah game lain: waktu yang dihabiskan untuk mengelola monster digital adalah waktu yang tidak dihabiskan untuk belajar Go. Jika saya terus bermain video game, saya tidak akan punya waktu untuk belajar. Ingat, waktu tidak pernah ditemukan: itu dibuat. Oleh karena itu, saya memilih untuk tidak memainkan game lain sampai saya menginvestasikan setidaknya dua puluh jam di Go. Saya tidak dapat mengalihkan perhatian jika saya ingin belajar dengan cepat. Berikut adalah taktik yang berguna dalam situasi semacam ini: cara terbaik untuk mengubah perilaku Anda adalah dengan mengubah struktur lingkungan langsung Anda. Jika Anda tidak ingin melakukan sesuatu yang sedang Anda lakukan, buat itu tidak mungkin untuk dilakukan. Jika Anda tidak dapat membuat perilaku itu mustahil, buatlah sesulit mungkin, mahal, atau semahal mungkin. Semakin banyak effort yang dibutuhkan, semakin kecil kemungkinan Anda untuk kembali ke perilaku Anda sebelumnya. Selamat tinggal, World of Warcraft ... Senang sekali mengenalmu. Sebelum memulai petualangan Go saya, saya membatalkan akun World of Warcraft saya dan menghapus game dari komputer saya. Jika game tidak diinstal, saya tidak dapat memainkannya bahkan jika saya mau. Saya tidak akan bermain

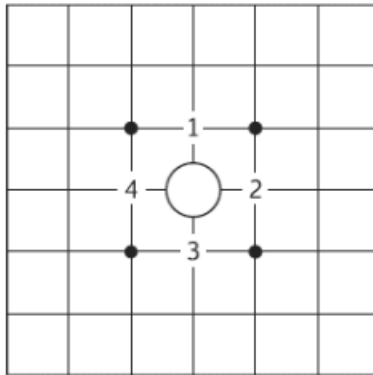
video game dalam keadaan apa pun sampai eksperimen Go saya selesai secara sosial.

Aturan Main, Diperiksa Kembali

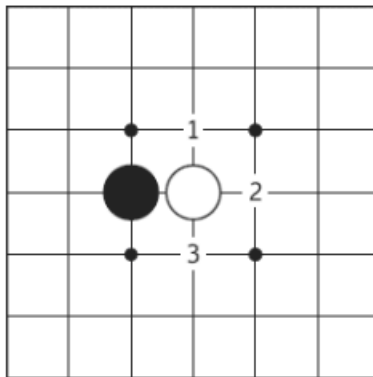
Tempat terbaik untuk memulai adalah dengan memeriksa kembali aturan. Saya sudah membacanya, tetapi saya perlu melihat seperti apa bentuknya di goban. Ada fitur Go bagus yang membuatnya lebih mudah dipelajari: Anda dapat mengubah ukuran papan. Karena goban simetris, Anda dapat membuat papan lebih besar atau lebih kecil, selama Anda memilih garis nomor ganjil. Aturan dan taktik yang sama masih berlaku. Papan kompetisi berumur sembilan belas kali sembilan belas, tetapi untuk tujuan pembelajaran, yang terbaik adalah memulai dengan papan tujuh kali tujuh. Itu cukup besar untuk memiliki ruang untuk mempelajari ide-ide kunci, tetapi cukup kecil sehingga papan tidak berlebihan. Mari kita lihat seperti apa aturannya di papan yang sebenarnya.

Beri aku kebebasan, atau beri aku kematian

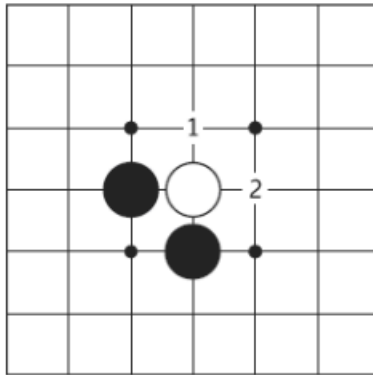
Mari kita mulai dengan aturan ketiga Go: Batu "ditangkap" ketika mereka dikelilingi di semua sisi oleh batu lawan. Saat Anda memainkan batu di persimpangan, persimpangan yang berdekatan dengan batu itu disebut kebebasan. Bayangkan tentara di *abattleeld*: jika ada tempat yang aman untuk mundur di dekatnya, para prajurit tidak dapat ditangkap. Jumlah maksimum kebebasan yang dapat dimiliki batu adalah empat:



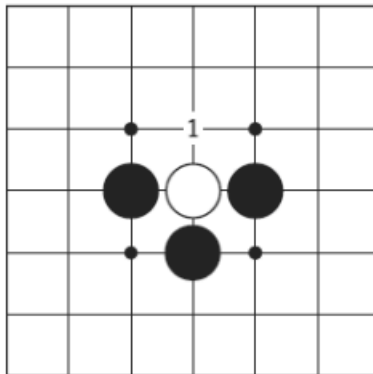
Kebebasan berkurang ketika lawan menyerang dengan menempatkan batu yang berdekatan langsung dengan milik Anda. Serangan ini mengurangi Whitesstone menjadi tiga kebebasan:



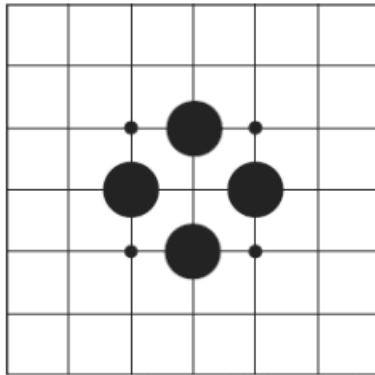
Serangan lain mengurangi batu Putih menjadi dua kebebasan:



Hati-hati, White! Hanya satu kebebasan yang tersisa:



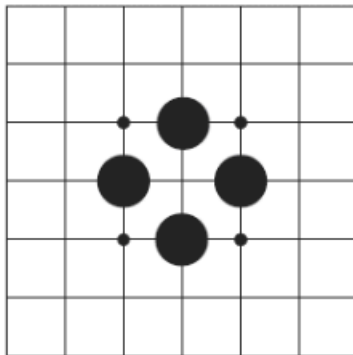
Situasi ini disebut atari.6 Jika Hitam menghilangkan kebebasan terakhir Putih, batu Putih ditawan, dan dikeluarkan dari papan:



Sejauh ini cukup sederhana. Jika Anda tidak ingin batu Anda ditangkap, Anda perlu memastikan mereka memiliki kebebasan sebanyak mungkin. Jika Anda ingin menangkap batu lawan Anda, singkirkan kebebasan mereka.

Tolong Jangan Bunuh Diri

Ini adalah kesempatan bagus untuk melihat aturan keempat: Memainkan batuyang segera ditangkap ("bunuh diri") dilarang. Mari kita lihat situasi terakhir itu lagi:

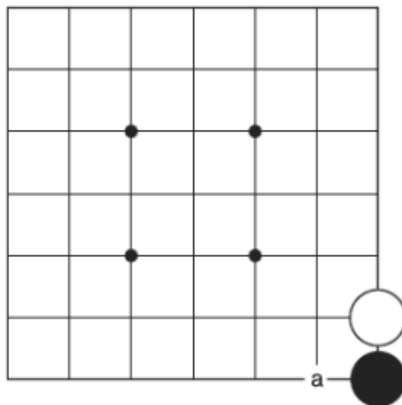


Bentuk ini disebut ponnuki, dan sangat umum di Go. Ruang di tengah batu Hitam tidak memiliki kebebasan, jadi jika Putih

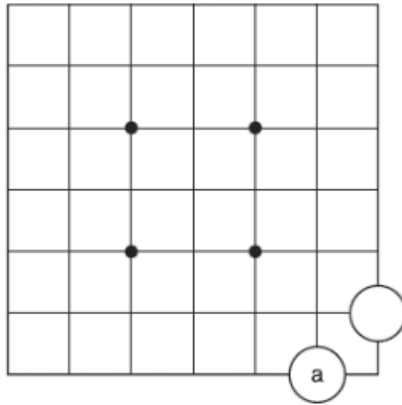
memainkan batu di sana, itu akan segera ditangkap. Aturan keempat melarang gerakan "bunuh diri" seperti ini. Aturan ini menjadi sangat penting ketika kita mulai melihat kelompok batu yang lebih besar, jadi kita akan kembali sebentar lagi.

Mundur ke Sudut

Di tengah goban, batu biasanya memiliki banyak kebebasansampai diserang. Di sudut dan sisi papan, ada lebih sedikit kebebasan, karena tidak ada banyak persimpangan di sekitarnya. Di sini, Hitam hanya memiliki satu kebebasan yang tersisa: persimpangan berlabel "a":



Hitam ada di atari. Jika White bermain di "a", Black ditawan:

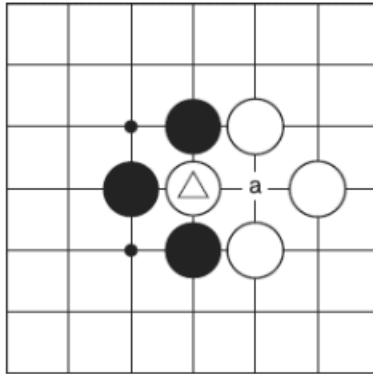


Contoh ini menggambarkan perbedaan antara sudut dan tengah papan. Di sudut, hanya dibutuhkan dua batu untuk menangkap. Di tengah, dibutuhkan setidaknya empat. Di samping, dibutuhkan tiga batu untuk ditangkap. Akibatnya, biasanya paling mudah untuk membangun dan mempertahankan wilayah di sepanjang sudut papan. Sisinya sedikit lebih sulit, dan tengahnya yang paling sulit dari semuanya, karena dibutuhkan begitu banyak batu untuk menjebak lawan. Itu sebabnya, jika Anda melihat awal game Go, pemain terampil mengikuti pola umum. Pertama, mereka akan menetapkan posisi di sudut papan. Setelah wilayah awal itu aman, mereka akan bercabang ke samping. Bagian tengah papan adalah prioritas terakhir, dan disediakan untuk akhir permainan. Itu sebabnya Shusaku menunggu sampai langkah 127 untuk memainkan Gerakan Kemerahan Telinga.

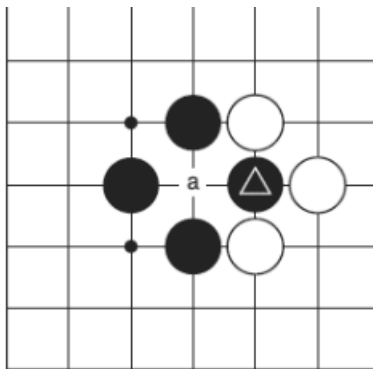
Ke Innity, tetapi Tidak Melampaui

Sekarang setelah kita mempelajari aturan penangkapan, mari kita periksa aturan kelima: Mengulangi urutan gerakan yang sama berulang kali dalam loop innite dilarang.

Ini adalah situasi yang disebut ko, yang merupakan kata Jepang untuk "innity". Loop Innite dimungkinkan di Go, dan jika diizinkan, mereka akan merusak sebagian besar game. Begini penampilan ko. Hitam dapat menangkap batu Putih, yang ditandai dengan segitiga, jika mereka bermain pada titik "a":



Perhatikan apa yang terjadi? Kami memiliki pola yang sama, hanya terbalik. Sekarang, jika White bermain di "a", mereka dapat mengambil kembali batu itu:

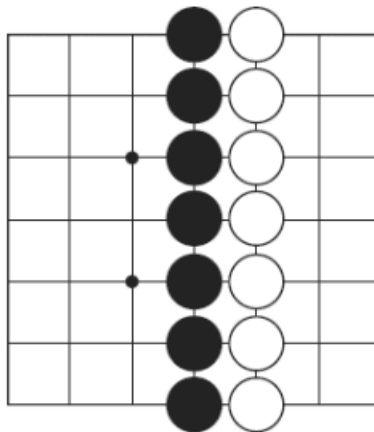


Tanpa aturan melawan ko, Putih dan Hitam bisa bertarung memperebutkan batu ini sampai akhir zaman. Itu sebabnya aturan itu ada: dalam ko ght,

Pemain yang kehilangan Stone RST mereka tidak dapat mengambil kembali KO sampai setidaknya satu giliran berlalu, mencegah loop Innite. Dalam turnamen, biasanya ada aturan tambahan yang disebut "superko". Bayangkan ada dua atau lebih ko ghts di papan. Pemain bisa berputar melalui ko ghts dalam lingkaran innite yang lebih besar. Setelah pola dibuat, super ko dimulai, dan pemain harus melakukan sesuatu yang lain, atau permainan berakhir.

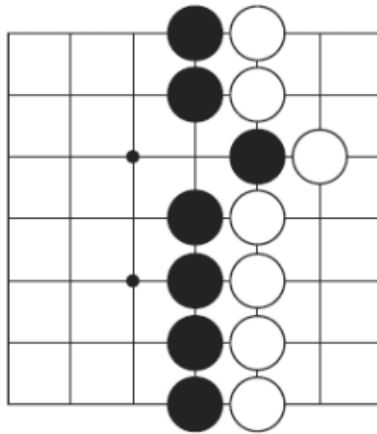
Kondisi Kemenangan

Sekarang, mari kita lihat bagaimana permainan berakhir. Berikut aturan enam: Permainan berakhir ketika para pemain kehabisan batu, satu pemain kebobolan permainan, atau kedua pemain mengoper. Cukup sederhana. Bagaimana kita menentukan siapa yang menang? Itu aturan tujuh: Pemain yang mengelilingi wilayah paling banyak di papan di akhir permainan menang. Mari kita lihat contohnya. Pada papan tujuh kali tujuh, ada empat puluh sembilan persimpangan. Mari kita asumsikan papan dibagi seperti ini:

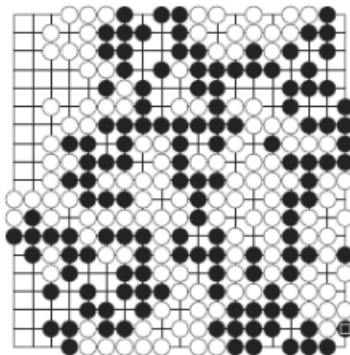


Hitam mengontrol sisi kiri papan, yang menyumbang dua puluh delapan ruang wilayah. Putih mengontrol sisi kanan, yang

sama dengan dua puluh satu spasi. Hitam menang. Berikut variasinya: Hitam telah berhasil menyerang sedikit ke wilayah White. Hitam sekarang memiliki dua puluh sembilan spasi, dan Putih memiliki dua puluh:



Penghitungan menjadi lebih rumit ketika papan lebih besar, dan grup tersebar di mana-mana, seperti akhir Ear ReddeningGame:

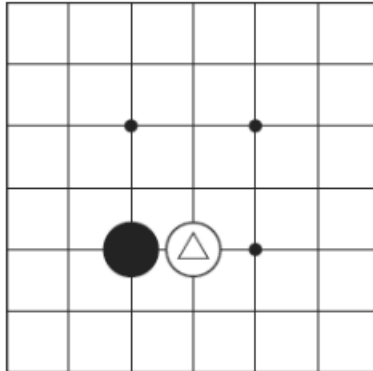


Prinsip umum menghitung wilayah adalah ini: jika wilayah diperebutkan, tidak ada pihak yang mendapat poin untuk itu. Terkadang maju

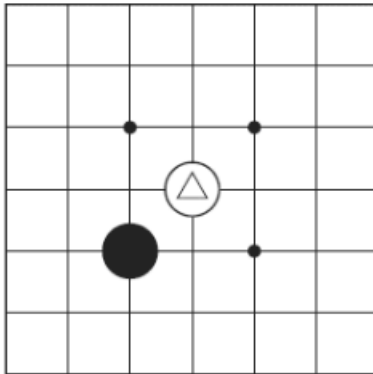
Pemain akan menyerahkan wilayah kepada lawan tanpa memainkan resolusi penuh, sehingga wilayah itu dihitung sebagai yang ditangkap. Jika ada perselisihan tentang siapa yang mengendalikan wilayah apa di akhir permainan, permainan dilanjutkan sampai perselisihan diselesaikan. Memperkirakan wilayah adalah keterampilan, dan itu membutuhkan latihan.

Berjaga-jaga!

Itu mencakup aturan dasar. Sekarang, bagaimana dengan taktik umum? Hal pertama yang perlu diketahui adalah seperti apa serangan itu. Serangan langsung menghapus kebebasan dari salah satu batu lawan Anda. Di sini, Black menyerang batu putih yang ditandai dengan segitiga:

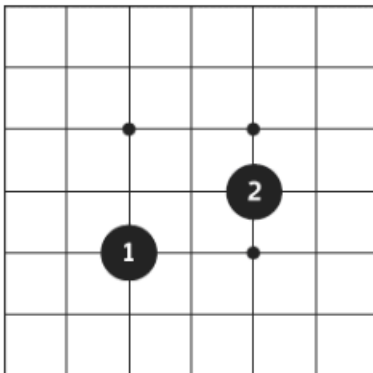


Anda juga dapat menyerang secara tidak langsung, dengan menempatkan batu di dekat lawan Anda. Serangan tidak langsung adalah cara untuk memperkuat posisi Anda di area sebelum serangan langsung Anda dimulai. Di sini, Hitam mengancam Batu putih yang ditandai dengan segitiga:



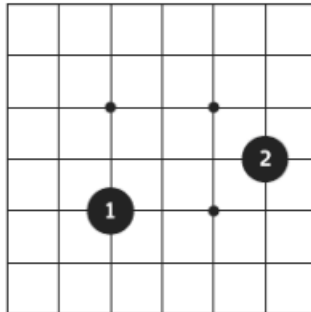
Tuduhan Ksatria

Serangan adalah elemen dasar oense. Elemen dasar pertahanan disebut koneksi dan sendi. Gerakan yang menyeimbangkan oense dan pertahanan disebut pendekatan. Salah satu pendekatan umum disebut "gerakan ksatria", yang mendapatkan namanya dari gerakan ksatria dalam catur:



Gerakan ksatria seimbang antara oense dan pertahanan: dekat dengan batu lain, jadi mudah untuk terhubung jika perlu, tetapi itu

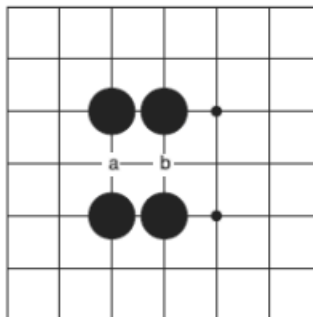
memperpanjang Inuence Anda lebih dari satu langkah bergerak.
Ada juga gerakan ksatria besar yang bahkan lebih agresif:



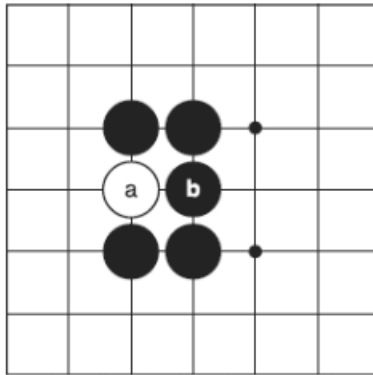
Gerakan ini menukar pertahanan dengan inuence: memperluas jangkauan pemain, tetapi batu akan lebih sulit untuk dipertahankan jika diserang.

Bambu Lebih Kuat Dari Baja

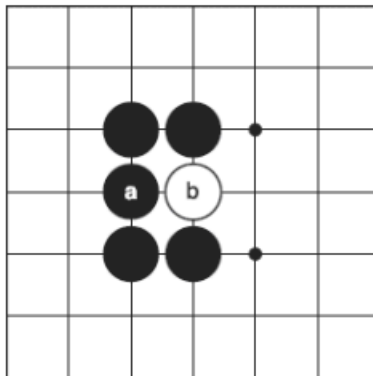
Di sisi pertahanan strategi, ada bentuk yang disebut bamboojoint yang sangat kokoh. Begini tampilannya:



Sendi bambu sangat kuat karena, tidak peduli bagaimana lawan Anda mencoba menyerangnya, mudah untuk menghubungkan batu Anda menjadi bentuk yang kuat. Dalam hal ini, jika Putih menyerang "a", Hitam dapat terhubung dengan "b":



Demikian juga, jika Putih menyerang "b", Hitam dapat terhubung dengan "a":



Setelah Black menghubungkan sambungan bambu, akan dibutuhkan sembilan batu tambahan untuk menangkap kelompok tersebut. Tingkat perlindungan itu sulit dikalahkan. Cukup mudah: menyerang, bertahan, dan menangkap tampak relatif mudah, dan gerakan ksatria dan sendi bambu adalah taktik yang mudah diingat dan dipahami.

Ditempa oleh seratus pertempuran?

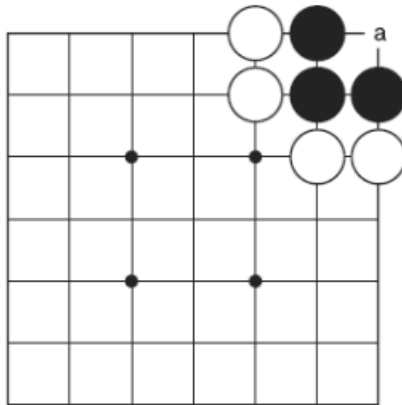
Sekarang saya tahu aturannya, serta dasar-dasar strategi, saya memiliki pilihan untuk dibuat. Haruskah saya mencoba meningkatkan permainan saya dengan menghadapi lawan manusia, atau apakah jalan yang paling mudah untuk mempelajari permainan dengan membaca buku dan memecahkan masalah peringkat? Pemain ahli di Perpustakaan Sensei memiliki pendapat yang beragam. Beberapa pemain terampil memanggil Sun Tzu, penulis *The Art of War*, berpendapat bahwa yang terbaik adalah "ditempa oleh seratus pertempuran." Yang lain percaya mempelajari masalah Go dan membaca buku tentang strategi dan taktik adalah cara terbaik untuk meningkatkan diri. Saya tidak yakin harus berpikir apa. Ada klub Go lokal yang bertemu setiap hari Rabu, tetapi mereka bertemu di sore hari, yang sesuai dengan waktu keluarga. Saya ingin bermain, tetapi itu tidak berhasil saat ini. Ada opsi lain: bermain di server Internet Go. Pemain dari seluruh dunia berkumpul secara online untuk bermain, dan server naik 24–7. Lebih baik lagi, game-game ini diberi peringkat, sehingga Anda dapat memperkirakan kekuatan relatif Anda jika Anda memainkan game yang cukup. Ini bukan peringkat turnamen, tapi itu sesuatu. Saya memutuskan untuk memeriksa dua Godestination online paling populer: IGS (Internet Go Server) dan KGS (Kseido Go Server). Bermain itu sederhana: Anda membuat akun dengan alamat email Anda, mengunduh program Go, dan masuk. Dari sana, Anda dapat meminta pemain lain untuk permainan, atau menonton pemain lain bertarung secara real time. Saya membuat akun di kedua sistem, lalu mencari lawan nyata pertama saya. Saya tidak memiliki peringkat, karena saya baru saja membuat akun, jadi saya mengundang pemain lain yang tidak berperingkat. Dia menerimanya, dan permainan dimulai. Karena kami berdua tidak berperingkat, tidak ada batu cacat. Saya bermain Black, dan berpikir permainan awal saya solid. Kami bergiliran

menetapkan wilayah di sudut rst, lalu di samping. Tiga puluh langkah ke dalam permainan, saya merasa cukup baik. Kemudian, White menyerang. Saya tidak khawatir: wilayah saya terlihat kuat, jadi saya menghubungkan batu saya untuk memperkuat posisi saya. Studi saya sampai saat ini tidak mempersiapkan saya untuk apa yang akan terjadi. Ganti demi belokan, White menghancurkanku. Wilayah yang saya pikir aman diserang secara brutal. Batu-batu yang kupikir aman disandera. Saya mencoba melakukan serangan balik, dan kemajuan saya diblokir. Saya mencoba membela, dan perlindungan saya gagal. Saya terus mencoba untuk mendapatkan pijakan, tetapi satu jam setelah pertandingan, jelas tujuan saya tidak ada harapan. Saya mengakui. Setelah pertandingan selesai, saya bertanya kepada lawan saya berapa lama dia bermain." Bertahun-tahun," katanya. "Saya baru saja membuat akun di server ini, jadi saya belum memiliki peringkat." Tidak heran dia menghancurkanku! Saya mengatakan kepadanya bahwa saya adalah pemain baru, dan meminta saran tentang cara meningkatkannya." Tonton pertandingan 10 hingga 20 kyu. Anda akan belajar banyak." Saya berterima kasih kepada lawan saya untuk permainannya, lalu masuk o.Apa yang salah? Melihat kembali permainan, saya pikir saya sedang membangun kelompok batu yang dapat dipertahankan, tetapi saya jelas tidak. Memahami mengapa kelompok-kelompok saya ditangkap membutuhkan melihat bagaimana kelompok besar dapat diserang.

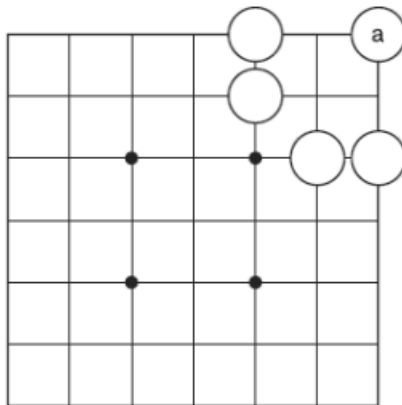
Satu Mata Buruk!

Ingat aturan yang melarang gerakan bunuh diri? Ada kerutan penting dalam aturan itu: jika gerakan yang seharusnya bunuh diri menangkap satu atau lebih batu lawan dengan cara yang menciptakan kebebasan baru, itu diperbolehkan.

Berikut contohnya. Persimpangan di "a" tidak memiliki kebebasan untuk Kulit Putih, karena dikelilingi oleh Hitam:

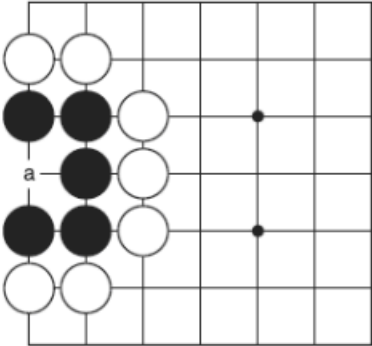


Biasanya, Putih tidak akan diizinkan bermain di "a" di bawah aturan bunuh diri, tetapi dalam hal ini, Putih memiliki Hitam yang dikepung. Kebebasan terakhir Black yang tersisa kebetulan adalah "a." Ketika White memainkan "a", kelompok Black ditangkap:

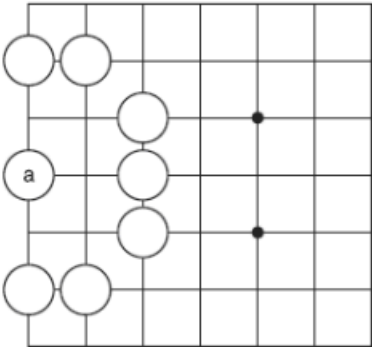


Di Go, jenis ruang ini disebut mata. Tergantung pada situasinya, mereka bisa menjadi sumber kekuatan atau kelemahan. Jika

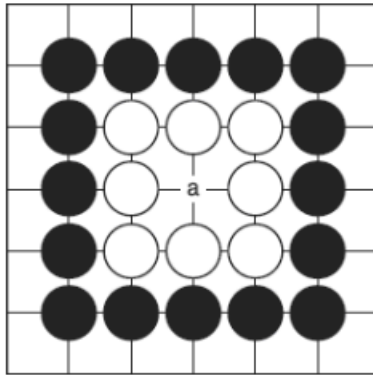
kelompok hanya memiliki satu mata, itu lemah. Di sini, Black dalam masalah:



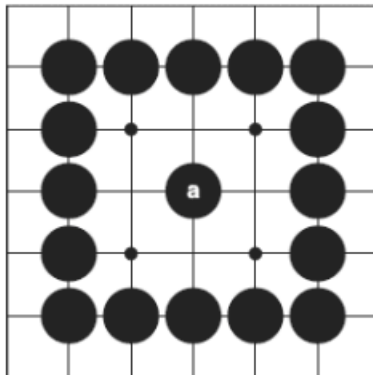
White menangkap di "a":



White dalam masalah sekarang:



Tangkapan hitam pada "a":

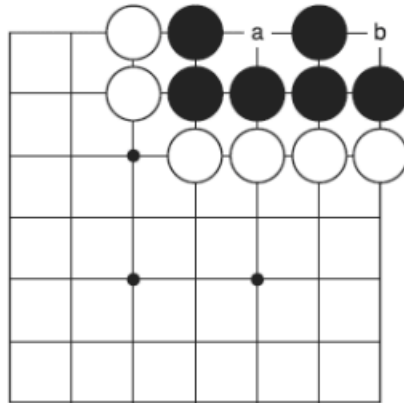


Ini membutuhkan sedikit waktu untuk membiasakan diri. Grup dengan hanya satu mata mudah ditangkap, jadi mereka pada dasarnya tidak aman.

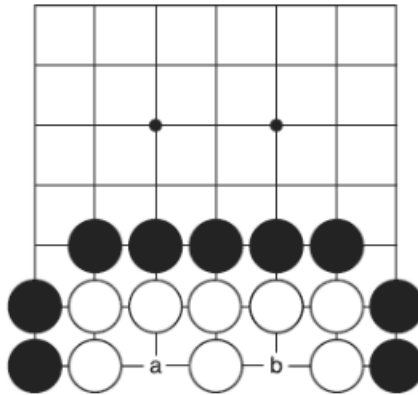
Dua mata bagus!

Itu tidak berarti mata itu buruk: mereka perlu. Itu tampak aneh sampai kita memeriksa aturan bunuh diri lagi. Pengecualian untuk aturan melarang bunuh diri hanya berfungsi jika batu menghasilkan penangkapan kelompok lawan. Jika tidak,

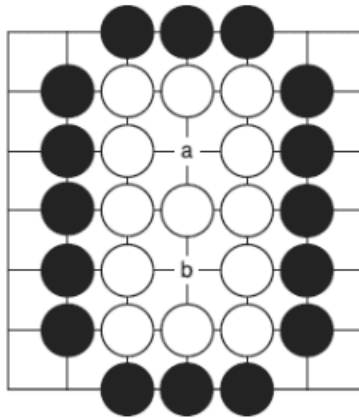
Pindah tidak diperbolehkan di bawah aturan bunuh diri. Itu berarti bahwa kelompok dengan dua mata tidak terkalahkan. Ini seperti membangun benteng di tengah wilayah musuh: tidak peduli berapa banyak batu yang digunakan lawan untuk menyerang, mereka tidak akan dapat menangkap kelompok. Di sini, kelompok Black memiliki dua mata:



Jika White bermain "a", itu adalah bunuh diri. Jika White bermain di "b", itu adalah bunuh diri. Putih tidak bisa bermain di "a" dan "b" pada saat yang sama, jadi Hitam aman, meskipun kelompok itu dikelilingi di semua sisi. Di sini, sekelompok batu Putih memiliki dua mata di satu sisi:



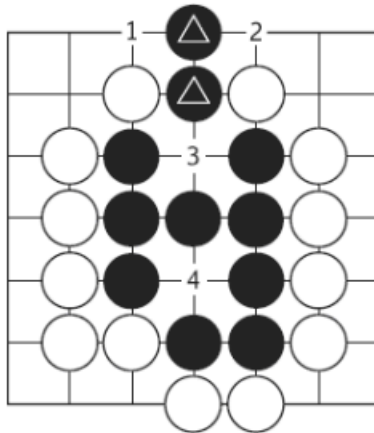
Di sini, sekelompok batu Putih memiliki dua mata di tengah:



Perhatikan bagaimana dibutuhkan enam batu untuk membentuk dua mata di sudut, delapan batu untuk membentuk dua mata di samping, dan tiga belas batu untuk membentuk dua mata di tengah? Perbedaan dalam kemampuan itu adalah mengapa sudut paling mudah untuk dipertahankan, dan bagian tengah selalu sulit untuk diamankan.

Mata Palsu (dari Sauron?)

Jika memiliki dua mata atau lebih dalam satu kelompok membuatnya tak terkalahkan, membentuk mata secepat mungkin menjadi prioritas besar. Sayangnya, tidak selalu mudah untuk mengetahui apakah bentuk yang terlihat seperti mata sebenarnya adalah mata atau bukan. Seringkali, kelompok akan membentuk mata yang terlihat padat, tetapi rentan terhadap serangan. Bentuk-bentuk ini disebut mata palsu, dan mereka mematikan. Berikut contohnya. Kelompok Black terlihat seperti memiliki dua mata, tetapi batu di bagian atas yang ditandai dengan segitiga rentan:

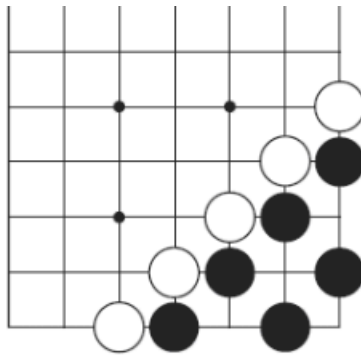


Katakanlah giliran White. Putih dapat menyerang pada 1 atau 2, menghilangkan kebebasan. Hitam mungkin mencoba melarikan diri di sepanjang sisi atas, tetapi dalam langkah berikutnya, Putih akan memotong rute pelarian. Tidak peduli bagaimana Blacks merespons, Putih kemudian akan menangkap batu yang ditandai dengan bermain di 3. Itu membuat sisa kelompok Hitam rentan, dan seluruh kelompok akan ditangkap ketika Putih bermain di 4. Aduh! Bagian utama dari strategi Go melibatkan pembuatan grup dengan mata dua, menghindari mata palsu, mencegah lawan Anda membangun mata yang benar, dan menghancurkan lawan Anda saat mereka membangun mata palsu.

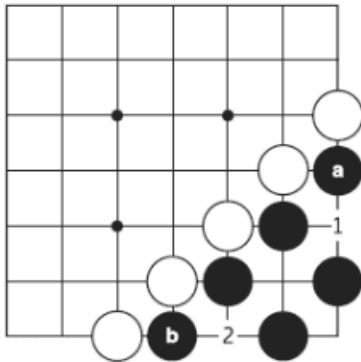
Dibutuhkan banyak pengalaman dan keterampilan untuk menciptakan mata sejati, dan untuk memperhatikan ketika mata palsu.

"Pembongkaran"

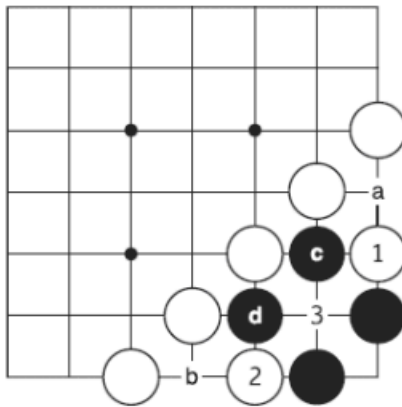
Itu membawa kita lebih dekat untuk memahami penyebab kerugian besar pertama saya. Saya pikir saya menciptakan wilayah yang aman, tetapi pada kenyataannya, saya rentan terhadap serangan, dan saya tidak menyadarinya tepat waktu. Berikut adalah contoh sederhana: lihat semua mata yang telah dibuat Black. Dengan empat mata, kelompok itu harus tak terkalahkan, bukan?



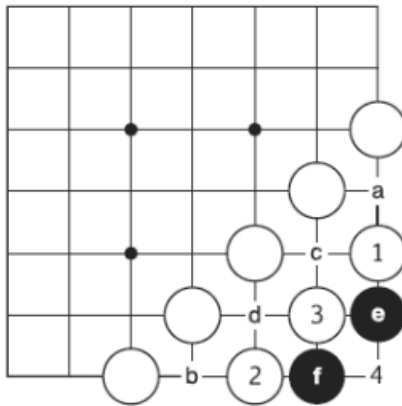
Sayangnya bagi Black, tidak demikian. Jika Putih bermain pada 1 atau 2, "a" atau "b" akan ditangkap:



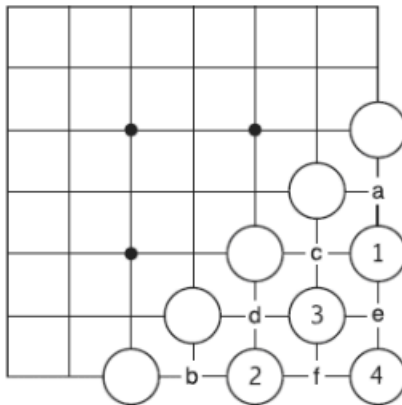
Kemudian, jika Putih bermain pada 3, "c" dan "d" akan ditangkap:



Terakhir, jika Putih bermain di 4, "e" dan "f" akan ditangkap:



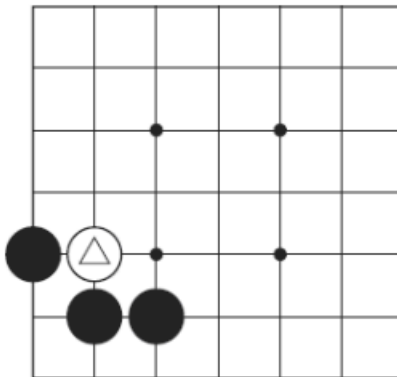
Akibatnya, White mampu sepenuhnya membongkar kelompok "tak terkalahkan" Black. Mereka tidak pernah menjadi mata sejati, dan Black membayar harganya:



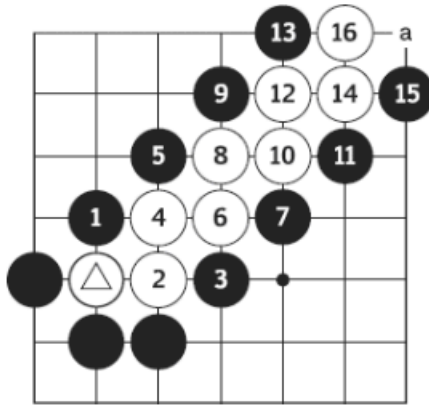
Itulah esensi dari apa yang terjadi pada saya. Saya pikir saya sedang membangun wilayah yang aman di sudut dan di samping, tetapi saya tidak membangun dua mata yang sebenarnya. Akibatnya, lawan saya mampu membongkar semua yang saya bangun. Pelajaran yang dipetik.

Parasut dan Tangga

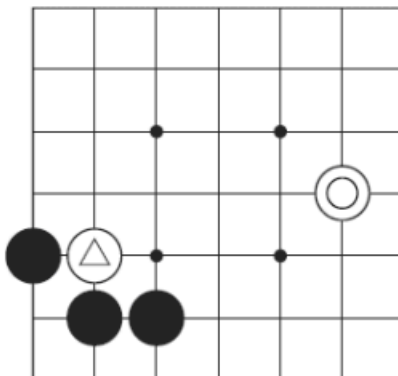
Ada beberapa konsep strategis penting lainnya yang tidak sepenuhnya saya hargai sampai saya melihatnya beraksi. Yang pertama disebut tangga. Setiap kali serangan terjadi, pihak lain memiliki pilihan: mereka dapat melakukan serangan balik, atau mereka dapat lari. Seringkali, mencoba melarikan diri adalah pilihan yang lebih baik. Jika lawan mengejar, a capture race dimulai. Mari kita lihat contohnya. Di sini, batu Putih (ditandai dengan segitiga) akan diserang:



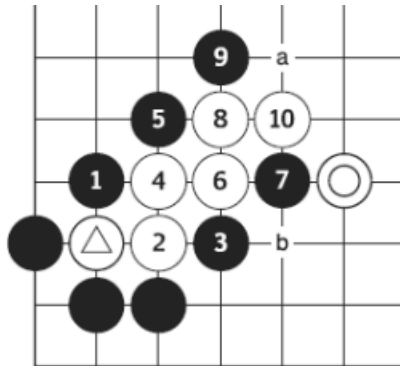
Hitam menyerang di 1, dan Putih melarikan diri di 2. Hitam menyerang pada 3, dan Putih melarikan diri pada pukul 4. Hitam menyerang pada pukul 5, Putih melarikan diri pada pukul 6, dan seterusnya. Akhirnya, kami menemukan diri kami di sini:



Giliran Hitam, dan Putih tidak punya tempat untuk lari. Ketika Hitam bermain duduk "a", semua batu Putih ditangkap, dan Hitam mengendalikan seluruh papan. Tamat! Ada pepatah Go lama: "Jika Anda tidak tahu apa itu tangga, jangan bermain." Situasi ini adalah ilustrasi yang baik tentang mengapa itu adalah posisi yang valid. Sekarang, mari kita periksa kembali situasi yang sama dengan sedikit twist: Whitememiliki batu (ditandai dengan lingkaran) di sisi paling kanan papan. Apa yang terjadi?



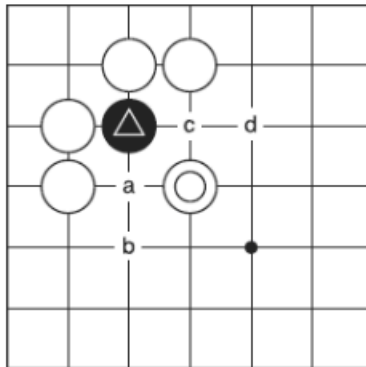
Bagian pertama adalah sama: Hitam menyerang di 1, dan Putih melarikan diri di 2. Hitam menyerang pada pukul 3, dan Putih melarikan diri pada pukul 4. Pola berlanjut sampai Putih melarikan diri pada 10. Pada titik itu, kita melihat ini:



Jika Hitam menyerang "a", melanjutkan pola, Putih akan menangkap Hitam di 7 dengan bermain di "b". Jika Black mencoba bertahan di "b", Whitescape dari kemarahan Black dengan bermain di "a." Satu batu di sisi berlawanan papan menyelamatkan White dari pemusnahan total. Itulah yang membuat Go begitu menarik: batu yang terlihat terputus dan jauh dari aksi utama mampu mengubah jalannya permainan. Membaca tangga adalah keterampilan. Di papan sembilan belas kali sembilan belas, ada banyak cara untuk memulai balapan menangkap, atau mengarahkan diri Anda pada serangan yang ingin Anda hindari, membuat tangga. Dalam kasus ini, Anda harus dapat membaca ke depan untuk memprediksi siapa yang akan memenangkan perlombaan. Jika Anda tidak berpikir Anda bisa menang, lebih baik mengsucikan satu atau dua batu daripada kehilangan seluruh kelompok. Jika Anda memprediksi kemenangan, itu adalah kepentingan Anda untuk memprovokasi lawan Anda untuk memainkan permainan yang tidak bisa mereka kembar, yang pada akhirnya memberi Anda keuntungan besar.

Melempar Jaring

Melempar batu ke jalur tangga potensial bisa menjadi permainan yang sangat bagus. Demikian juga, menjebak lawan Anda dengan menempatkan batu di posisi sekitarnya bisa menjadi strategi yang baik. Di sini, Putih ingin menangkap batu Hitam (ditandai dengan segitiga). Alih-alih menyerang secara langsung, Putih melemparkan jaring yang mengepung Hitam secara tidak langsung:



Jika Hitam mencoba melarikan diri di "a", Putih memblokir di "b." Jika Hitam mencoba melarikan diri di "c", Putih memblokir di "d." Tidak peduli seberapa banyak Blackstruggles, dia tidak bisa lepas dari jaring White. Selain masalah mata, lawan saya jauh lebih baik memprovokasi saya untuk terlalu banyak. Saya menemukan diri saya di tangga yang tidak bisa saya kembar, atau terjebak dalam jaring yang tidak saya sadari sampai terlambat. Semua hal inilah yang membuat praktisi terampil pandai dalam permainan. Jika Anda tidak tahu bentuk dasar seperti mata, tangga, dan jaring, Anda akan kalah dari pemain yang tahu cara membuat dan memanfaatkannya.

Memaksimalkan Waktu Latihan

Setelah kekalahan saya, saya mengambil saran lawan saya ke dalam hati, dan mencoba mencari permainan dengan pemain dengan peringkat antara 10 dan 20 kyu.

Sayangnya, tidak banyak yang bisa ditonton. Server Internet Go bukanlah tempat yang paling ramah pemula di internet. Server dan program utama telah ada sejak sekian lama, jadi antarmukanya langsung dari tahun 1990-an, dan butuh waktu untuk mengetahui cara menggunakannya. Sepertinya juga tidak banyak pemain berperingkat rendah di sekitar. Dan jika Anda tidak tahu cara memainkan game dengan baik, menginjak berulang kali bukanlah pengalaman yang paling menyenangkan. Sebagian besar pertandingan aktif yang saya temukan adalah antara pemain 8 kyu ke atas. Saya mencoba menonton beberapa pertandingan, tetapi tingkat pembelajaran saya sangat lambat. Saya bisa mengikuti gerakan hanya ne, tetapi bukan alasan mengapa para pemain membuatnya. Terkadang urutan gerakan terjadi terlalu cepat, dan saya tidak dapat mengikuti analisisnya. Saya yakin ada motif di balik setiap batu, tetapi saya terlalu tidak berpengalaman untuk mengungkapkan apa itu. Permainan Go penuh dapat memakan waktu lebih dari satu jam untuk diselesaikan, yang merupakan penghalang lain. Beberapa kali, saya mulai menonton pertandingan, hanya harus menjauh untuk mempertengkar Lela atau membantu Kelsey dengan tugas. Karena game berlangsung secara real time, itu berarti saya tidak dapat mengikuti game hingga resolusi. Saya mencoba pendekatan "seratus pertempuran", dan berdasarkan eksperimen awal saya, itu bukan cara paling pintar untuk meningkatkan diri sebagai pemula. Saya memutuskan untuk mengubah strategi saya, mengganti ounding yang tidak tercerahkan dengan studi terstruktur.

Manusia Melawan Mesin

Setiap malam, saya menghabiskan empat puluh menit mempelajari masalah Go, menggunakan program SmartGo di iPad saya, dan kemudian dengan mengerjakan seri buku Graded Go Problems Kiseido. Setelah mempelajari masalah, saya kemudian memainkan game melawan program AI bawaan SmartGo di papan kecil sembilan kali sembilan. Di rst, SmartGo

menginjak saya, bahkan dengan cacat delapan batu. Memalukan! Namun, saat saya belajar, permainan saya mulai meningkat dengan cepat. Saya belajar memanfaatkan batu cacat saya dengan melakukan theoense, menyerang White dengan cepat dan membentuk koneksi yang kuat bila memungkinkan. Tidak butuh waktu lama untuk mengurangi batu handicap saya menjadi enam, lalu ve, saat saya memenangkan pertandingan melawan AI. Setelah tiga puluh pertandingan, saya berada di sekitar tanda handicap empat batu. Memperkirakan skor itu menantang: terkadang saya pikir saya akan melakukannya dengan baik, tetapi ketika saya mengintip skor, saya tertinggal dengan margin yang signifikan. Karena memenangkan permainan membutuhkan gerakan protable yang meningkatkan jumlah wilayah yang Anda kendalikan, itu masalah. Seringkali, gerakan yang saya pikir meningkatkan wilayah saya tidak, menyia-nyiakan gerakan dan menciptakan celah bagi lawan.

Mode Kegagalan

Sehubungan dengan mempelajari masalah Go, saya membaca buku-buku yang saya ambil dari daftar bacaan Perpustakaan Sensei. Apa yang saya pelajari sangat membantu saya. Mempelajari How Not to Play Go adalah penggunaan waktu yang sangat baik. Menurut Yuan Zhou, penulis, pemula membuat beberapakesalahan umum yang merugikan mereka:

Pemain yang tidak berpengalaman mengikuti lawan mereka secara membabi buta. Misalnya, saat diserang, pemain yang buruk akan secara otomatis merespons dengan bertahan, melarikan diri, atau melakukan serangan balik di area lokal yang sama, alih-alih mempertimbangkan semua opsi yang tersedia. Pemain yang tidak berpengalaman tidak memperhatikan seluruh papan. Terkadang, langkah terbaik bukanlah di mana aksi berlangsung; itu berada di sisi lain papan, jauh dari segalanya. Pertempuran itu mengasyikkan, dan pemain yang malang bisa buta terhadap peluang yang ada di daerah goban yang lebih tenang. Pemain yang tidak berpengalaman tidak membuat langkah yang paling menguntungkan. Go membutuhkan pemikiran dalam hal prot dan biaya: setiap gerakan membutuhkan biaya batu, dan Anda selalu ingin mendapatkan keuntungan terbaik untuk uang Anda. Seringkali, ini melibatkan pensucian satu atau dua batu, asalkan Anda akan menuai manfaat yang lebih besar di tempat lain.

- Pemain yang tidak berpengalaman tidak menghargai sente. Di Go, "inisiatif" adalah masalah besar: Anda ingin mendikte tanggapan lawan Anda sebanyak mungkin. Alih-alih fokus pada agenda mereka, Anda ingin lawan Anda sangat khawatir tentang mencegah potensi kerugian sehingga mereka mengabaikan mengamankan keuntungan. Siapa pun yang membangun dan menjaga sente hampir selalu menang. Pemain yang tidak berpengalaman tidak pandai memperkirakan wilayah. Akibatnya, mereka akan menghabiskan banyak waktu untuk melakukan pertempuran lokal atau mengamankan sudut kecil papan sementara lawan mereka mengambil alih sejumlah besar wilayah tanpa tertandingi. Pemain yang tidak berpengalaman cemburu dengan posisi lawan. Zhou menyebut ini "masalah mata merah", dan biasanya mengakibatkan seorang pemain membuat gerakan yang sangat tidak dapat diproporsihanya karena mereka merasa wilayah lawan mereka semakin besar. Pemain yang tidak berpengalaman menyerah pada angan-angan. Sangat mudah untuk terpicak oleh sekelompok batu yang dapat Anda tangkap hanya dengan dua batu, dan gagal memperhitungkan bahwa Anda tidak dapat melakukan dua gerakan berturut-turut: lawan Anda dapat merespons apa yang Anda lakukan. Akibatnya, Anda membuang-buang batu mulia, semua dengan harapan-bahwa lawan Anda terlalu bodoh untuk memperhatikan apa yang Anda lakukan.

Kualitas-kualitas ini menyentuh rumah bagi saya, terutama angan-angan. Berapa kali saya menyerang kelompok, berharap lawan mengabaikan rencana saya? Berapa kali saya ditarik ke dalam pertempuran lokal, mengabaikan gambaran yang lebih besar? Setelah membaca How Not to Play Go, game saya meningkat pesat. Salah satu hal terbesar yang harus saya perbaiki adalah perasaan mendalam bahwa menangkap batu lawan adalah jalan terpendek menuju kemenangan. Dalam catur dan dam, itu benar: di Go, tidak. Menangkap batu sangat membantu, tetapi itu bukan kondisi kemenangan. Mengamankan wilayah adalah tujuan Go, dan sangat sering, Anda dapat melakukannya tanpa menangkap batu lawan sama sekali. Pelajaran itu membutuhkan beberapa saat untuk meresap: menangkap batu terasa menyenangkan, bahkan jika itu sebenarnya kontraproduktif.

Kebijaksanaan Pepatah

Aphorisms adalah alat pengajaran yang sangat umum di Go. Karena gim ini sangat rumit, aturan praktis membantu pemain mengingat cara memainkan situasi yang tidak biasa.

Salah satu contoh paling awal dari pendekatan pengajaran ini adalah "The Ten Golden Rules of Go," yang dikaitkan dengan Wang Jixin, yang bermain sekitar masa dinasti Tang, yang memerintah di Tiongkok dari sekitar 600 hingga 900 M:7

1. "Yang serakah tidak mendapatkan kesuksesan." 2. "Jangan tergesa-gesa untuk memasuki wilayah lawan." 3. "Jaga diri sendiri saat menyerang orang lain." 4. "Buang batu untuk mendapatkan sente." 5. "Tinggalkan kecil untuk menabung besar." 6. "Saat dalam bahaya, sakral." 7. "Buat bentuk yang tebal, hindari gerakan tergesa-gesa." 8. "Sebuah gerakan harus menanggapi lawan." 9. "Melawan posisi kuat, bermain dengan aman." 10. "Carilah kedamaian, hindari ghting dalam situasi terisolasi atau lemah."

Banyak kebijaksanaan Go seperti itu. Ada banyak pepatah Go:

Poin kunci musuh adalah poin kunci Anda. Jika ragu, tenuki (bermain di tempat lain). Ada kematian di hane (batu yang menjangkau untuk menyerang). Pemula bermain atari. Ponnuki bernilai tiga puluh poin. Jaring lebih baik daripada tangga. Buat st sebelum memukul. Plum sakral untuk buah persik. Orang kaya tidak boleh bertengkar. Jika Anda mengendarai harimau, sulit untuk mendapatkan o. (Jangan memulai apa yang tidak ingin Anda tonish.)

Ada banyak hal yang harus dipelajari: tidak hanya bijaksana untuk mempelajari peribahasa dan mengapa mereka ada, Anda perlu memahami kapan harus mengabaikannya. Lagi pula, salah satu peribahasa adalah "Jangan mengikuti amsal secara membabi buta." Ada juga peribahasa yang berlaku untuk pemula. Favorit saya adalah "Hanya setelah pukulan kesepuluh Anda akan melihat st," yang biasanya diikuti oleh "Hanya setelah pukulan kedua puluh Anda akan dapat memblokirnya." Dalam pengalaman saya sejauh ini, itu sangat benar: pemain terampil dapat menyerang saya dengan cara yang bahkan tidak dapat saya kenali. Saya harus banyak belajar.

Pertanyaan "Lima Batu"

Hampir sebulan setelah belajar Go, saya menemukan sebuah esai oleh Sam Bleckley yang menggambarkan pengalamannya mempelajari permainan.8 Dalam esai itu, Samrelay saran dari seorang teman, yang menyarankan agar dia bertanya pada dirinya sendiri empat pertanyaan spesifik tentang situasi di papan tulis sebelum meletakkan batu:

1. Bisakah saya memastikan salah satu kelompok saya hidup? (Dengan membentuk dua mata, menghubungkan, dll.)
2. Bisakah saya membunuh salah satu kelompok lawan saya?
3. Bisakah saya mengamankan wilayah saya?
4. Bisakah saya menyerang wilayah lawan saya?

Keempat pertanyaan itu menangkap esensi dari empat prioritas utama dalam Go: hidup, membunuh, mengamankan, menyerang. Dengan mengerjakan pertanyaan dalam urutan itu sebelum Anda bergerak, Anda akan segera mendapatkan kekuatan "batu besar" dengan membuat keputusan yang lebih baik dan menghindari perdagangan yang buruk. Sangat mudah untuk menghafal dan mengingat daftar empat prioritas. Dengan bertanya pada diri sendiri beberapa pertanyaan sederhana sebelum Anda memainkan batu, dan dengan mempertimbangkan gerakan bernilai tertinggi sebelum gerakan bernilai lebih rendah, Anda memastikan bahwa keputusan Anda saat ini sesuai dengan prioritas Anda yang sebenarnya. Go membutuhkan pengambilan keputusan dengan informasi yang tidak lengkap di lingkungan yang berubah, jadi teknik seperti ini sangat berguna. Pertanyaan-pertanyaan seperti ini memberikan struktur dan logika dalam keadaan yang tidak pasti. Lebih baik lagi, dimungkinkan untuk mengidentifikasi dan menggunakan pertanyaan "vestone" di bidang lain dalam hidup Anda. Setiap kali Anda berada dalam situasi yang mengharuskan membuat keputusan dalam informasi yang tidak lengkap, menetapkan prioritas, dan menerima perdagangan, kemungkinan besar ada serangkaian pertanyaan "ve stone" yang sederhana dan jelas yang dapat membantu Anda membuat keputusan yang lebih baik.

Mencapai Tanda Dua Puluh Jam

Berdasarkan catatan saya, setelah sekitar satu bulan belajar, saya telah mencapai tanda dua puluh jam. Di mana saya, dan apa yang telah saya capai? Menurut database di SmartGo, saya telah memecahkan sekitar 150 masalah peringkat, dan saya telah maju ke set 18 kyu. Saya telah memainkan tiga puluh tiga pertandingan, dan saat ini saya memiliki handicap empat batu melawan AI. Saya juga telah menyelesaikan volume pertama dalam seri Kseido Graded GoProblems, yang diberi peringkat 30 hingga 25 kyu. Saya berada di tengah-tengah buku kedua, yang diberi peringkat 25 hingga 15 kyu, jadi aman untuk memperkirakan saya berada di suatu tempat di kisaran 18 hingga 20 kyu. Tidak buruk.

Meninjau Metode

Mari kita tinjau inti dari metode yang saya gunakan untuk mempelajari cara bermain Go:

Saya mempelajari aturan dasar permainan. Saya membeli goban, batu, dan beberapa buku tentang strategi Go yang ditujukan untuk pemula. Saya menemukan beberapa cara untuk mempelajari prinsip-prinsip strategis yang paling penting dengan mengerjakan masalah Go berperingkat, baik dalam program komputer maupun dalam buku. Saya memfokuskan latihan awal saya untuk memecahkan masalah peringkat, mengulangi latihan yang tidak dapat saya jawab atau jelaskan pada percobaan pertama. Saya berlatih masalah dan bermain game sampai saya mencapai dua puluh jam. Pada saat itu, saya dengan nyaman memecahkan masalah 20 kyu, peningkatan peringkat tujuh belas dari peringkat pemula 35 kyu.

Ke Mana Saya Pergi Dari Sini

Saya memiliki perasaan campur aduk tentang Go. Saya telah belajar cukup banyak tentang game untuk menghargai seberapa dalam dan seberapa jauh saya dari menguasainya. Saya telah berkembang pesat sejak saya mulai, tetapi untuk benar-benar bagus dalam permainan, jalan saya masih sangat panjang. Di sisi lain, waktu luang saya terbatas, dan Go tampaknya membutuhkan konsentrasi intens dan terfokus yang sama seperti yang dituntut oleh penulisan dan pemrograman. Meskipun Go bisa menyenangkan, pada akhirnya, Go terasa terlalu seperti bekerja.

Selain itu, permainan penuh sembilan belas kali sembilan belas dapat memakan waktu lebih dari satu jam untuk diselesaikan, dan tidak terlalu sopan bagi pemain manusia untuk harus keluar dari permainan jika tugas keluarga memanggil. Sejauh ini, sebagian besar game saya telah dimainkan melawan komputer, karena selalu tersedia dan tidak peduli jika saya harus melakukan sesuatu yang lain untuk sementara waktu. Saya telah tumbuh untuk menghargai kualitas asah otak dari masalah Go, jadi saya mungkin akan terus melakukannya ketika saya ingin memecahkan teka-teki. Saya menikmati latihan Go jauh lebih dari teka-teki silang, sudoku, dan jenis permainan serupa lainnya, jadi program Go saya akan tetap ada di ponsel saya, siap digunakan saat saya memiliki beberapa waktu luang. Saya berencana untuk mengajarkan Lela cara bermain ketika dia sedikit lebih tua. Aturannya cukup sederhana untuk dipahami oleh anak berusia empat tahun, dan saya memiliki kenangan indah bermain catur dengan ayah saya ketika saya masih muda. Go adalah cara yang baik untuk mengajarkan keterampilan penting seperti analisis, strategi, dan perdagangan, dan saya pikir kami akan bersenang-senang bermain di tahun-tahun mendatang. Selain itu, saya tidak memiliki keinginan yang membara untuk menghabiskan lebih banyak waktu untuk menguasai Go. Saya telah belajar cukup banyak tentang permainan untuk memuaskan rasa ingin tahu saya, dan terus maju ke penguasaan bukanlah prioritas utama. Itu tidak apa-apa. Tidak ada hukum universal yang mengatakan Anda harus menguasai semua yang pernah Anda pelajari. Hidup membutuhkan perdagangan, dan tidak ada salahnya mengeksplorasi sesuatu yang baru, belajar banyak, dan kemudian memutuskan untuk mengeksplorasi sesuatu yang lain. Anda tidak harus menjadi sabuk hitam dalam segala hal untuk menjalani kehidupan yang memuaskan.



8Ukulele

e

Pelajaran: Mengisolasi, Berlatih, Ulangi

Hari ini, seperti setiap hari, kita bangun dengan kosong dan ketakutan. Jangan membuka pintu ke belajar dan mulai membaca. Turunkan alat musik. Biarkan keindahan yang kita cintai menjadi apa yang kita lakukan. Ada ratusan cara untuk berlutut dan mencium tanah.
—RUMI, PENYAIR PERSIA ABAD KETIGA BELAS

• • •

Untuk gambar tambahan, video, dan komentar tentang bab ini,
visithttp://first20hours.com/ukulele.

Beberapa tahun yang lalu, saya membaca serangkaian buku yang menggugah pikiran oleh Tom Hodgkinson berjudul *How to Be Idle* (2005) dan *The Freedom Manifesto* (2007). Tesis Hodgkinson, singkatnya, adalah bahwa kita bekerja terlalu banyak, dan semua berdiri untuk mendapatkan manfaat dengan mengendur. Alih-alih mendorong diri kita sendiri hingga batasnya, kita harus belajar untuk rileks, bersantai, dan tidak menganggap semuanya terlalu serius. Buku-buku tersebut terutama merupakan seruan untuk mengabaikan budaya konsumen dan karierisme. Alih-alih mengejar kelimpahan materi yang berlebihan, Hodgkinson merekomendasikan untuk puas dengan cara sederhana dan kembali ke pengejaran masa-masa yang lebih sederhana: berkebun, memasak, dan bermain musik. Dalam *The Freedom Manifesto*, Hodgkinson merekomendasikan untuk belajar bermain ukulele:

Keajaiban empat senar ini sangat murah, sangat portabel, dan sangat mudah dimainkan. Oleh karena itu, bahkan lebih punk daripada gitar. Berikut adalah tiga akord yang Anda butuhkan untuk memainkan sebagian besar lagu: C F G

Dapatkan ukelele, dan Anda tidak akan pernah bosan lagi.

Nasihat Hodgkinson beresonansi dengan saya. Pada saat itu, Kelsey dan saya tinggal di apartemen studio seluas 340 kaki persegi di Upper East Side New York City, dan kami berdua banyak bekerja, bergegas untuk maju dalam karir kami dan membayar sewa Manhattan yang gila. Adiversion akan disambut baik, dan belajar bermain ukulele terdengar sangat menyenangkan. Di sekolah menengah, saya terlibat dalam setiap grup musik sekolah saya. Saya sudah menyebutkan bahwa saya belajar bermain terompet, tetapi saya juga bernyanyi di paduan suara, dan sangat menikmatinya. Namun, begitu saya pergi ke perguruan tinggi, saya berhenti melakukan sesuatu yang musikal, dan buku Hodgkinson mengingatkan saya betapa saya merindukannya. Bagian dari masalahnya adalah bahwa terompet keras, dan tidak terlalu menyenangkan untuk dimainkan sendiri. Saya cukup sibuk dengan hal-hal lain dan tidak tertarik untuk mencari band untuk bergabung. Selain itu, Anda tidak dapat menyanyi dan memainkan terompet secara bersamaan, seperti yang Anda bisa dengan gitar, piano, atau ukulele. Setelah sedikit penelitian, saya akhirnya membeli Washburn Oscar Schmidt OU5 Concert Ukulele. Itu terbuat dari kayu Koa Hawaii, terdengar bagus, dan harganya terjangkau: antara ukulele dan casing, saya menghabiskan sekitar \$ 175. Ketika ukulele tiba, saya memetik senar sebentar, tetapi saya belum pernah memainkan alat musik gesek sebelumnya, selain dari eksperimen yang sangat singkat dengan gitar murah di perguruan tinggi. Tanpa gagasan bagus tentang cara memulai, hal-hal lain menjadi prioritas, dan ukulele menyesatkan lemari di samping gitar, mengumpulkan debu.

Pahlawan Ukulele

Ketertarikan saya untuk bermain ukulele dihidupkan kembali ketika Kelsey dan saya menyambut Lela ke dunia. Sungguh menakjubkan bagaimana memiliki anak mengubah prioritas Anda dalam semalam. Tiba-tiba, saya bekerja lebih sedikit

mendukung menghabiskan waktu di rumah, dan saya mulai berpikir tentang musik lagi. Salah satu sahabat saya, Nate Siebert, memainkan gitarnya untuk putranya, Jackson dan Finley, setiap malam sebelum mereka tidur. Mereka menyukainya, dan selama bertahun-tahun itu berkembang menjadi ritual yang konsisten dan menenangkan. Sebagai manfaat tambahan, musik sangat mungkin baik untuk mereka: paparan awal musik telah dikaitkan selama bertahun-tahun dengan berbagai ukuran yang terkait dengan perkembangan kognitif. Tidak jelas persis bagaimana musik membantu otak kecil berkembang, tetapi memang demikian.¹ Saya ingin Lela tumbuh di rumah di mana bermain musik adalah acara umum, dan saya semua untuk melakukan apa yang saya bisa untuk membantunya berkembang menjadi gadis kecil yang cerdas dan bahagia. Belajar bermain ukulele tiba-tiba tampak seperti penggunaan waktu yang sangat baik.

Temui Kapak Baru Saya

Saya sudah memiliki ukulele, tetapi instrumen lain telah menarik perhatian saya. Sepupu Kelsey, Erik Smith, kebetulan adalah seorang ahli kayu otodidak yang membuat instrumen. Perusahaan Erik, CrowHill Guitars,² mengkhususkan diri dalam membuat gitar akustik dan elektrik khusus buatan tangan, dan Erik memamerkannya di festival musik di seluruh Amerika Serikat. Erik memutuskan untuk mencoba membuat ukulele tenor pertamanya. Temui Grizzly:



Ukulele ini adalah salah satu dari jenisnya: tubuhnya dibuat dari mahoni Afrika dan wenge, dengan tatahan cedar Spanyol, Indianrosewood, kayu merah keriting, dan maple keriting. Cakar beruang mutiara putih tertanam di fretboard, dan cakar beruang abalon menghiasi lubang suara di wajah. Saya menginginkan Grizzly segera setelah saya melihatnya. Namun, ada tangkapan: instrumen Erik berharga ribuan dolar. Mereka bernilai setiap sen dalam hal kualitas, tetapi apakah saya cukup serius untuk belajar bermain untuk membenarkan biayanya? Saya memutuskan untuk membeli Grizzly karena dua alasan. Pertama, memainkan instrumen berkualitas tinggi, bahkan sebagai pemula, memberi Anda dorongan psikologis yang penting: Anda langsung terdengar lebih baik. Dibandingkan dengan Grizzly, Oscar Schmidt lama saya terdengar benar-benar membosankan. Akibatnya, bermain di Grizzly jauh lebih menyenangkan. Kedua, akan sangat memalukan jika memiliki instrumen sebagus ini di rumah saya dan tidak bisa memainkannya. Selain itu, jika saya tidak belajar bermain, saya akan membuang-buang banyak uang. Saya sekarang berinvestasi secara substansial dalam hasilnya, jadi kemungkinan besar saya akan meluangkan waktu untuk berlatih, menjadikan pembelian ini sebagai contoh yang bagus untuk menggunakan prakomitmen untuk mengubah perilaku.

Perlengkapan Pendukung

Bersama dengan Grizzly, saya mengambil dua peralatan penting lainnya:

Snark SN-6 Ukulele Tuner—tuner elektronik murah ini terjepit ke kepala ukulele, membuatnya mudah untuk menjaga setiap senar tetap disetel dengan nada yang tepat. Senar Ukulele D'Addario J71 Pro-Arte, Tenor—senar gitar klasik nilon tegangan ekstra keras yang memberikan suara yang lebih penuh pada ukulele dan membuatnya lebih mudah untuk memetik dengan cepat. 3. Jika senar putus, Anda harus menggantinya sebelum melanjutkan, jadi ada baiknya untuk memiliki beberapa set tambahan, untuk berjaga-jaga.

Bersama dengan peralatan, saya membeli tiga buku ukulele:

Absolute Beginners Ukulele, Omnibus Edition oleh Steven Sproat (2010), tutorial pemula
Peta Jalan Fretboard—Ukulele oleh Jim Belo dan Fred Sokolow (2006), yang mencakup teknik lanjutan
The Daily Ukulele oleh Jim Belo dan Liz Belo (2012), sebuah buku referensi yang berisi berbagai macam lagu mulai dari folk hingga rock klasik

Tantangan yang Menarik

Tak lama setelah saya membeli Grizzly, saya mendapat email dari teman saya Chris Guillebeau, penulis *The \$100 Startup* dan *The Art of Non-Conformity*. Setiap tahun, Chris menyelenggarakan konferensi yang sangat menyenangkan yang disebut KTT Dominasi Dunia. Bayangkan menyatukan lebih dari seribu orang yang masing-masing melakukan hal-hal gila/aneh/menarik di area umum yang sama: itu adalah WDS. Chris awalnya mengundang saya untuk berbicara tentang *The Personal MBA*, tetapi sepuluh hari sebelum acara, pembicara lain membatalkan sesi mereka pada menit terakhir. Chris tahu saya sedang mengerjakan buku ini, jadi dia mengiriminya catatan: apakah saya bersedia memimpin sesi kedua tentang akuisisi keterampilan cepat? Yakin... Mengapa tidak? KTT Dominasi Dunia akan menjadi pertama kalinya saya mempresentasikan ide-ide ini di depan umum.

Saya sedang menyusun presentasi baru, yang berfokus pada ide-ide yang kami diskusikan di bab 1-3, ketika saya menerima email lain dari Chris, yang dia kirimkan ke setiap pembicara sebelum acara. Antara lain, catatan Chris berisi nasihat ini: "Jika Anda dapat melakukan sesuatu yang istimewa, lakukanlah. ... Apa pun yang dapat Anda lakukan untuk mengejutkan [para peserta] akan menonjol dan melangkah jauh." Itu membuat saya berpikir ... Bermain ukulele akan menjadi cara yang bagus untuk menunjukkan cara kerja ide-ide ini. Bisakah saya belajar cara bermain inten days? Ini akan menjadi peregrangan, tetapi saya pikir saya bisa melakukannya. Saya membalas Chris dan menerima undangan itu. Saya sekarang memiliki sepuluh hari untuk belajar cara bermain ukulele.

Gila untukmu

Dengan hanya sepuluh hari sebelum penampilan publik pertama saya, saya harus membuat rencana, dan cepat. Ini bukan komitmen yang dapat saya mundur dengan mudah, dan ini akan menjadi pertama kalinya dalam dua belas tahun saya melakukan sesuatu yang bermusik di depan umum. Setelah saya berkomitmen, bagian otak saya yang ingin saya terlihat bagus di depan orang lain secara alami muncul. Apa yang saya pikirkan? Apakah inuts? Apa yang membuat ide gila ini semakin menegangkan adalah bahwa saya akan melakukannya sendiri: hanya saya dan Grizzly. Saya tidak akan memiliki band untuk mendukung saya, dan jika saya membuat kesalahan, semua orang akan tahu. Jika saya mengerikan, semua orang akan melihat dan mendengar betapa buruknya saya. Hanya memikirkan kemungkinan itu membuatku gugup. Saya sedang mempertimbangkan untuk berbicara, memainkan alat musik yang belum pernah saya mainkan sebelumnya, dan mungkin bernyanyi di depan orang-orang yang tidak saya kenal. Yikes! Semakin banyak alasan untuk mengetahui apa yang saya lakukan sebelum saya naik ke panggung. Saya memiliki banyak pekerjaan yang harus dilakukan, dan waktu terus berdetak.

Kesuksesan Denning

Bagian pertama dari tingkat kinerja target saya telah ditetapkan: Saya akan memainkan ukulele di depan umum dalam sepuluh hari. Itu awal yang baik, tetapi bagaimana saya bisa membuatnya lebih spesifik? Saya mengeluarkan buku catatan dan membuat sketsa seperti apa sesi yang saya inginkan. Saya punya satu jam untuk berbicara. Jika saya menjelaskan ide-ide utama dalam tiga puluh menit atau lebih, itu menyisakan dua puluh menit untuk bermain dan sepuluh menit untuk pertanyaan. Saya akan berencana bermain selama sekitar dua puluh menit. Apa yang ingin saya fokuskan? Pembicaraan itu bukan konser. Ini terutama sesi pengajaran, jadi saya ingin fokus menggunakan apa pun yang saya mainkan untuk mengilustrasikan beberapa ide menarik dan penting tentang perolehan keterampilan yang cepat. Alangkah baiknya juga untuk menyoroti beberapa hal menarik tentang musik. Saat saya membuat sketsa, saya mulai mendapatkan ide ... Ini akan menyenangkan.

"Lagu Empat Akor"

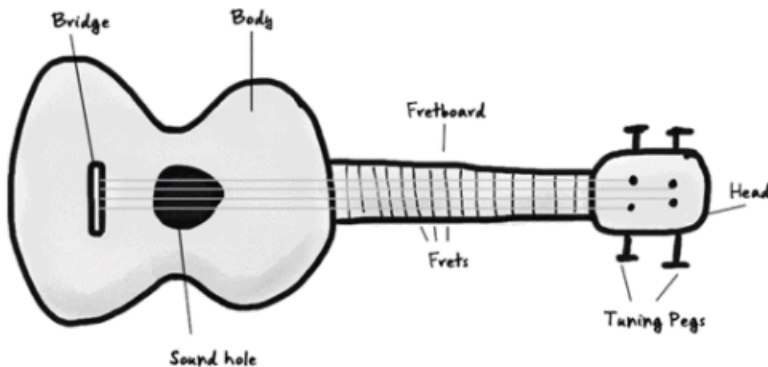
Saat saya memikirkan apa yang mungkin ingin saya mainkan, saya teringat video lucu dari sebuah band yang saya lihat beberapa bulan sebelumnya. Axis of Awesome⁴ adalah band rock komedi yang berbasis di Sydney, Australia. Grup ini paling dikenal luas karena "Four Chord Song," medley hits pop masa lalu dan sekarang, yang telah ditonton lebih dari 24 juta kali di YouTube. Inilah leluconnya: seperti yang ditunjukkan band, hampir setiap lagu populer dalam beberapa dekade terakhir terdiri dari empat akord yang sama. Mainkan akord berulang kali, overlay beberapa lirik acak, dan Anda memiliki hit pop yang pasti. Tidak ada artis yang kebal: Elton John, The Beatles, John Denver, U2, The Red Hot Chili Peppers, Bob Marley, Beyonce, Lady Gaga. Bahkan lagu rakyat Australia yang terhormat "Waltzing Matilda" muncul. Lagu ini berlangsung selama tiga menit, memperkenalkan lagu baru

tekan setiap ve hingga sepuluh detik, disertai dengan tawa dan tepuk tangan dari penonton. "Four Chord Song" secara bersamaan histeris dan menakjubkan. Dengan menunjukkan struktur yang mendasari lagu-lagu terkenal, ini menggambarkan bahwa musik populer memang memiliki formula. Ada banyak kemungkinan variasi pada tema, tetapi pikiran manusia tampaknya tertarik pada metode khusus untuk membangun lagu yang menarik. Berikut adalah empat akord dalam "Four Chord Song":

L / D / DI / C

Saya akan belajar cara memainkan "Lagu Empat Akor." Jika lelucon itu benar, saya juga akan secara bersamaan belajar cara memainkan setiap lagu pop yang pernah ditulis. Kedengarannya seperti tempat kuno untuk memulai! Ternyata, keempat akord ini sangat mudah dimainkan setelah Anda mengetahui cara kerja instrumen. Mari kita periksa ukulele untuk melihat apa yang kita hadapi.

Anatomi Ukulele



Ukulele memiliki empat senar, yang memanjang sepanjang badan instrumen. Potongan panjang yang menonjol dari badan instrumen disebut fretboard.

Tepat di atas fretboard adalah sepotong kayu yang disebut head orheadstock. Tertanam di kepala adalah pasak penyetelan, yang menahan ujung atas empat senar instrumen dengan aman. Senar ini memanjang dari kepala, di atas fretboard, melewati lubang suara terbuka, ke jembatan, yang berada di dekat bagian bawah tubuh ukulele. Jembatan menjaga senar tetap di tempatnya, digantung erat di atas lubang suara dan fretboard. Ukulele secara tradisional dimainkan dengan membungkus tangan kiri di sekeliling fretboard, dan menggunakan lengan kanan untuk memeluk instrumen dekat dengan dada. Pemain memetik senar dengan nger tangan kanan mereka (biasanya lingger atau ibu jari kanan), sementara tangan kiri mereka menekan fretboard untuk mengubah nada senar.

Tuning

Setiap senar memiliki penyetelan: nada yang dibuat jika Anda memetik senar tanpa menekan di mana pun di fretboard. Nada setiap senar disesuaikan dengan memutar pasak penyetelan. Jika pasak penyetelan diputar untuk menambah ketegangan pada senar, nada bergerak lebih tinggi. Jika pasak penyetelan diputar untuk mengurangi ketegangan senar, nada bergerak lebih lambat. Inilah penyetelan ukulele tenor default yang paling populer:5

Senar ke-4 (bawah) = G (196 Hz)senar ke-3 = C (261,6 Hz)senar ke-2 = E (329,6 Hz)senar ke-1 (atas) = A (440 Hz)

Menyetel ukulele sangat mudah jika Anda memiliki perangkat penyetelan yang bagus. Di situlah Snark SN-9 masuk. Saat Anda menjepitnya ke headstock, menyalakannya, dan memetik senar, tuner mengukur nada senar, menerjemahkan frekuensi menjadi nada, dan menunjukkan kepada Anda apakah nada berada pada (sedikit di bawah) atau tajam (sedikit di atas) yang tepat

Frekuensi. Ketika Anda memiliki nada yang benar, lampu sinyal muncul, dan Anda beralih ke senar berikutnya. Ketika keempat senar diselaraskan, Anda siap untuk bermain. Penyetelan yang tepat sangat penting: jika senar tidak disetel dengan benar, ukulele terdengar o dengan cara yang bahkan dapat diperhatikan oleh orang yang mengaku tuli nada. Semakin akurat penyetelannya, semakin baik suara instrumennya.

Catatan dan Akord

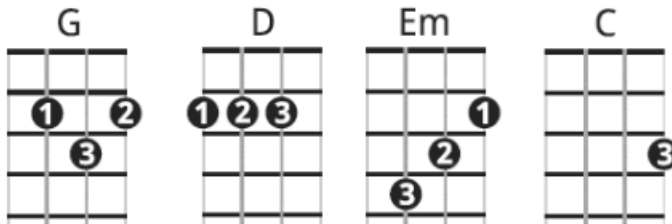
Jika Anda ingin memainkan satu nada pada ukulele, yang perlu Anda lakukan hanyalah memetik senar, yang menyebabkannya bergetar di atas lubang suara di tubuh, menciptakan suara. Jika Anda menekan fretboard saat Anda memetik senar, senar menjadi lebih pendek untuk sementara, dan sebagai hasilnya, bergetar lebih cepat saat Anda memetikinya. Semakin tinggi frekuensigetaran, semakin tinggi nada suara. Apa yang membuat ukulele, dan instrumen gesek lainnya, seperti gitarpaspas, dan piano, menarik adalah bahwa mereka mampu memainkan beberapa nada pada saat yang bersamaan. Jika seorang pemain ukulele memetik setiap senar dengan tangan kanannya, mereka dapat memainkan empat nada secara bersamaan. Seorang pemain yang terampil dapat menekan senar di fretboard dengan cara yang membuat nada yang dihasilkan setiap senar saling melengkapi, menciptakan achord. Akord adalah apa yang membuat alat musik gesek terdengar begitu kaya dan bersemangat. Dengan memainkan kombinasi nada alih-alih nada tunggal, pemain dapat menciptakan melodi dan harmoni secara bersamaan. Dalam arti, akord memungkinkan pemain untuk menjadi pengiring mereka sendiri. Lebih baik lagi, tidak seperti terompet, klarinet, ute, atau saksofon, alat musik gesek hanya mengharuskan pemain menggunakan tangan mereka. Dengan latihan yang cukup, dimungkinkan untuk memainkan akord yang kompleks dan bernyanyi pada saat yang sama, itulah sebabnya instrumen seperti gitar dan piano sangat populer di kalangan vokalis.

Memainkan Akord

Karena saya memulai dari awal, saya memiliki pilihan untuk dibuat: Saya dapat fokus mempelajari cara memainkan nada individu dengan membaca lembaran musik, atau berlatih memainkan akord. Karena akord baru bagi saya, dan saya ingin bisa bernyanyi dan bermain pada saat yang sama, saya condong ke arah itu. Diagram akor adalah representasi visual dari senar mana yang akan ditekan pada fretboard dengan tangan kiri Anda. Jika Anda menekan senar dalam kombinasi yang ditunjukkan pada diagram, Anda akan memainkannya akord. Berikut adalah empat akord dalam "Four Chord Song":

L / D / DI / C

Inilah tampilan akord ini pada ukulele:



Jika Anda memegang ukulele tegak, itu terlihat seperti diagram. Bilah tebal di bagian atas diagram mewakili bagian atas fretboard ukulele, yang disebut mur. Setiap baris di bawahnya adalah afret, yang sesuai dengan potongan logam yang terangkat yang mencuat keluar dari fretboard. Frets membuatnya lebih mudah untuk menghasilkan nada yang terdengar bersih saat Anda menekan fretboard. Saat Anda bermain, Anda tidak menekan fret itu sendiri: Anda menekan ruang di antara mereka. Saat Anda menekan, fret di bawah nger Anda juga menekan senar, memperpendeknya menjadi panjang yang tepat, dan mengubah nada yang sesuai.

Diagram akor menunjukkan senar mana yang perlu Anda tekan, dan di mana Anda perlu menekan, untuk memainkan akord tertentu. Senar diwakili oleh empat garis vertikal: senar di sebelah kiri adalah senar ke-4, yang merupakan senar di bagian atas jika Anda menahan kembali ukulele dalam posisi bermain. Senar di sebelah kanan adalah senar ke-1, yang akan berada di bagian bawah. Lingkaran hitam menunjukkan di mana harus menekan, dan angka-angka adalah singkatan dari nger mana yang harus Anda gunakan:

1 = indeks
2 = tengah
3 = cincin
4 = kecil
("kelingking") nger

Menurut diagram, untuk memainkan akord G, saya perlu menekan tiga senar terbawah dengan indeks, tengah, dan dering saya. Indeks saya harus menekan senar ketiga di atas fret kedua; Inggris-Inggris-Ku harus berada di posisi yang sama pada senar Therst, dan Ring Nger Ku harus menekan senar kedua di atas fret ketiga. Itu banyak yang bisa dikatakan, tetapi dalam praktiknya, itu mudah dilakukan. Setelah Anda memanggil nger mana yang pergi ke mana, memainkan akord menjadi keterampilan motorik standar: mempelajari bentuk apa yang perlu diambil oleh nger Anda, dan di mana Anda perlu mendorong, untuk membuat akord tertentu. Untuk berlatih akord, ada baiknya memiliki satu set diagram referensi yang menunjukkan cara memainkan akord umum. Pencarian singkat di Internet membawa saya ke "Kiwi Ukulele Indispensable Chord Chart,"⁶ panduan yang dapat diunduh untuk memainkan akord pada ukulele. Panduan ini dibuat oleh Mike Dickison, penulis Kiwi Ukulele: The New Zealand Ukulele Companion (2008). Ada banyak tempat untuk melihat bagan akor: cukup banyak buku everyukulele berisi satu set diagram akor. Apa yang membuat Mike's chart unik adalah diatur berdasarkan frekuensi penggunaan. Common chord, seperti akord yang ditampilkan dalam "Four Chord Song", terdaftar

rst. Akord langka, seperti G#KMK4, tercantum di bagian bawah, karena Anda jarang melihatnya digunakan. Jenis analisis frekuensi ini sangat, sangat berguna: hukum kekuasaan juga berlaku untuk musik. Untuk bisa memainkan lagu, kamu tidak perlu mengetahui ratusan akord. Sepuluh hingga dua belas akord mencakup sebagian besar lagu.

Latihan Akord

Saya memilih diagram akord dalam "Four Chord Song," dan meluangkan waktu untuk mempelajari siapa yang pergi ke mana. Ketika saya membuat bentuk yang sesuai dengan ngers saya di fretboard, saya memetik semua keempat senar. Ini berhasil: Saya dapat menghasilkan akord yang terdengar bagus! Tantangan terbesar sejak awal adalah menekan setiap senar ke bawah yang memastikan fret melakukan tugasnya. Jika Anda tidak menekan cukup keras, senar tidak akan menghasilkan nada yang jernih. Sebagai gantinya, Anda akan mendapatkan suara "berdengung" yang sangat tidak menarik bersama dengan nada yang Anda inginkan, yang mengganggu. Setelah saya mempelajari empat akord, saya menghabiskan satu jam untuk berlatih memainkan masing-masing, mengulangi urutan berulang kali:

G ... D... Di... C ... G
... D... Di... C ... G ...
D... Di... C ...

Ini adalah tahap pertama dari pemerolehan keterampilan motorik: membantu otak saya mengaitkan pikiran tertentu, seperti nama akord, dengan pola gerakan otot di tangan kiri saya. Nantinya, saya akan dapat memainkan akord Em kapan pun saya mau, tetapi untuk sampai ke sana, saya perlu berlatih membuat bentuk yang tepat dengan nger saya, menekan dengan cukup keras pada fretboard, dan beralih di antara akord secara berurutan. Saya juga berlatih sebelum tidur, menggunakan strategi yang sama yang saya gunakan untuk belajar kembali untuk mengetik sentuhan. Dengan berlatih sebelum tidur, saya dapat memastikan bahwa otak saya dapat mengkonsolidasikan gerakan-gerakan ini sebaik mungkin.

Pola Memetik

Keesokan harinya, saya dapat memainkan keempat akord dengan baik, dan beralih di antara mereka sesuka hati. Tidak buruk untuk satu jam latihan. Sekarang setelah saya memiliki gagasan tentang apa yang dilakukan tangan kiri saya, saya perlu fokus pada tangan kanan saya. Sementara tangan kiri sibuk memainkan akord, tangan kanan bertanggung jawab untuk memetik senar untuk menghasilkan suara. Sampai saat ini, saya telah memetik menggunakan pola yang sangat mendasar: menghitung tofour dan memetik dengan setiap angka:

1 2 3 4 1

2 3 4 1 2

3 4

Sebagian besar lagu, ketika dimasukkan dalam notasi musik di atas kertas, ditulis dalam apa yang disebut waktu 4/4. Tanpa membahas terlalu jauh ke detailnya, lagu-lagu dalam waktu 4/4 diatur di sekitar empat ketukan kuat. Lain kali Anda mendengar sebuah lagu, cobalah menghitung dari satu hingga empat di kepala Anda sesuai dengan irama. Lebih sering daripada tidak, Anda akan melihat pola ts. Tentu saja, jika setiap lagu terdiri dari memainkan nada atau akord dalam pola yang sederhana dan dapat diprediksi berulang kali, musik akan menjadi membosankan dengan sangat cepat. Untuk menambah minat dan variasi pada musik, musisi sengaja mematahkan pola penghitungan sederhana ini dengan menggunakan teknik yang disebut sinkop. Sinkop berarti bermain dengan cara yang menyimpang dari perkembangan ketukan yang teratur. Begini kedengarannya: dengan salah satu tangan Anda, ketuk ketukan sederhana pada permukaan di depan Anda—meja, meja, atau konter sangat cocok. Jaga ketukan yang kuat dan teratur. Saat Anda mengetuk, hitung dari satu hingga empat berulang kali:

1 2 3 4 1

2 3 4 1 2

3 4

Cukup sederhana, bukan? Sekarang, saat Anda mengetuk ketukan, tambahkan sentuhan. Gunakan tangan Anda yang lain untuk mengetuk di antara ketukan, seperti ini:

1 dan 2 dan 3 dan 4 dan 1
dan 2 dan 3 dan 4 dan 1
dan 2 dan 3 dan 4 dan

Itu sinkop yang sangat mendasar. Ketukan pada "dan" adalah ketukan standar, yang menambahkan sedikit variasi yang membuat ritme lebih menarik. Musisi yang terampil menghabiskan banyak waktu untuk bereksperimen dengan sinkop. Setiap ketukan dapat dibagi menjadi banyak bagian. Sangat umum untuk ketukan dibagi menjadi empat bagian, yang dihitung seperti ini:

1e&a 2e&a 3e&a 4e&a

Itu adalah "satu-e-dan-a, dua-e-dan-a, tiga-e-dan-a, empat-e-dan-a." "e" panjang, seperti dalam "elang", dan "a" pendek, seperti dalam "ah." Dengan menekankan atau memberi aksentuasi pada ketukan tertentu, dan dengan meninggalkan beberapa ketukan sepenuhnya, musisi dapat menciptakan segala macam ritme menarik yang masuk ke dalam lagu. Ini penting untuk diketahui karena tangan kanan pada ukulele menjaga ritme. Jika saya ingin dapat melakukan apa pun selain memetik pada irama, yang membosankan, saya perlu meluangkan waktu untuk memetik pola memetik yang menarik. Setelah beberapa eksperimen, saya memutuskan pola:

1 &a E&A 3 &a E&A

Butuh beberapa saat untuk melatih tangan kanan saya untuk memetik ritme, tetapi saya akhirnya mengembalikannya. Pertama, saya memainkan akord C sederhana saat Istrum. Setelah saya merasa nyaman, saya mulai bergerak melalui urutan empat akord, mengubah akor pada ketukan 1 dan 3:

GD1 &a e&a 3 &a
e&aEmC1 &a e&a
3 &a e&a e&a

Pada titik ini, saya memainkan lagu yang sebenarnya: Saya memukul akord dalam urutan yang benar dengan tangan kiri saya, dan menjaga pola memetik dengan tangan kanan saya. Untuk pertama kalinya, saya memainkan lagu di ukulele. Sederhana, tapi berhasil! Saya terus mengulangi urutan itu berulang kali. Setelah saya merasa nyaman, saya menutup mata, dan mencoba bermain tanpa melihat fretboard atau senar. Saya membuat lebih banyak kesalahan, tetapi saya dapat menjaga lagu tetap bergerak, dan kembali ke jalurnya ketika saya membuat kesalahan. Tidak buruk setelah hanya dua jam latihan.

Bermain dan Bernyanyi pada Waktu yang Sama

Sekarang saya dapat memainkan akord dan mempertahankan pola pemetik, saya memiliki satu layer lagi untuk ditambahkan. Bisakah saya mengerti cara bernyanyi dan bermain pada saat yang bersamaan? "Four Chord Song" lucu karena mudah untuk mengenali lagu-lagunya. Tanpa liriknya, kedengarannya seperti saya memainkan hal yang sama berulang kali. Itulah intinya, tentu saja, tetapi tanpa kata-katanya, lagu itu tidak masuk akal. Selama bertahun-tahun, saya telah bertemu banyak orang yang sangat keren melalui bisnis saya. Di antara yang paling keren adalah Derek Sivers, pendiri ofCDbaby.com. Derek adalah penyanyi dan gitaris ulung, jadi saya meminta saran kepadanya tentang bagaimana melanjutkan. Inilah yang dia rekomendasikan:

Hafalkan lagu hanya dengan menyanyikan rst! Penting untuk memisahkan pengetahuan instrumen dari pengetahuan lagu. Anda harus bisa menyanyikan semuanya, tanpa instrumen di tangan. (Kualitas suara tidak masalah: bernyanyi, bersenandung, bersiul, apa saja.)Setelah Anda menghafal lagu, nyanyikan nama-nada, bukan liriknya. Hafalkan lagunya seperti ini, mata tertutup. Terakhir, tambahkan instrumen, nyanyikan nama-nada saat Anda memainkannya pada senar.

Ini adalah contoh dekonstruksi yang brilian: memecah proses yang kompleks menjadi bagian-bagian yang lebih sederhana. Untuk bernyanyi dan bermain pada saat yang sama, Anda harus mengetahui kata-kata dan melodi lagu dengan cukup baik untuk mengingatnya saat Anda memainkan akord dan menjaga pola memetik tetap berjalan. Dengan memisahkan pembelajaran lagu dari instrumentat, Anda memastikan bahwa Anda mengetahui lagu dengan cukup baik sehingga tidak perlu terlalu memikirkannya saat menambahkan instrumen. Setelah Anda memiliki kata-kata, mengganti lirik ke nama akord membantu Anda merasakan kapan harus mengubah akor saat Anda bermain. Karena liriknya memiliki ritme tersendiri, Anda dapat menggunakannya untuk mengingat kapan harus beralih: satu hal yang lebih sedikit untuk dilacak! Akhirnya, dengan menyanyikan nama-nama akord sesuai irama dengan perubahan akord dan pola memetik, Anda membantu otak Anda menyatukan potongan-potongannya. Jika Anda dapat menyatukan semuanya, beralih dari nama akord ke lirik itu sederhana, karena Anda sudah menghafalnya. Saya mengambil nasihat Derek ke dalam hati, dan mulai menghafal lirik "Four Chord Song" dengan mendengarkannya berulang kali untuk mengambil lagunya. Saya kemudian mempraktikkan kata-kata dengan menuliskannya di buku catatan, lalu mengulangnya berulang kali dari ingatan. Kemudian, saya memainkan lagu di ukulele sambil bersiul, menjaga nama-nama akord di kepala saya. Sangat mudah untuk memperhatikan kapan harus mengubah akord, dan tiba-tiba, saya mulai bermain dan bersiul pada saat yang bersamaan. Menambahkan lirik datang lebih mudah dari yang saya harapkan. Sese kali, saya lupa kata-kata atau mengacaukan akord, jadi saya berhenti, mundur, dan mencoba lagi. Setelah berjam-jam latihan, saya dapat memainkan dan menyanyikan lagu penuh pertama saya. Kasar di beberapa tempat, tetapi semua bagian ada di sana. Lima hari sampai waktu pertunjukan. Saya mampu tampil cukup baik sendiri, tetapi bagaimana saya akan bertahan ketika orang-orang menonton saya?

Membuatnya Otomatis

Untuk memastikan saya bisa bermain ketika itu penting, sisa latihan saya pada lagu itu adalah pengulangan murni, selalu tepat sebelum tidur. Tujuannya sederhana: Saya ingin membuat mengubah akord, mempertahankan pola memetik, dan menyanyikan kata-kata membutuhkan sedikit semangat mental dan perhatian. Bagian dari latihan ini tidak glamor, tetapi penting. Segala sesuatu hingga saat ini telah menjadi bagian dari fase kognitif perolehan keterampilan motorik: menganalisis, mendekonstruksi, dan bereksperimen. Sekarang, saya perlu mendorong ke fase asosiatif, dan mulai membiarkan otot-otot saya mengambil alih tanpa berpikir sadar. Setiap hari, saya memainkan dan menyanyikan lagu itu berulang kali: Saya tidak menghitung berapa kali. Transisi akord saya menjadi lebih halus, pola memetik semakin konsisten, dan saya mengingat sebagian besar kata. Beberapa transisi antar lagu itu rumit, jadi saya menghabiskan waktu ekstra untuk mempraktikkannya. Sebelum saya menyadarinya, saatnya untuk bepergian ke Portland. Saya tidak akan bepergian sendirian: kali ini, Grizzly datang bersama saya.

Waktu pertunjukan

Penampilan saya adalah pada hari terakhir konferensi. Kabar baiknya adalah jadwal memberi saya setidaknya dua jam lagi untuk berlatih. Kabar buruknya adalah itu memberi saya dua hari untuk merasa cemas tentang bagaimana itu akan berjalan. Pada hari pembicaraan, saya menyetel Grizzly dan berlatih satu jam terakhir sebelum sesi saya. Kemudian, ukulele di tangan, saya berjalan ke tempat itu. Saya tidak yakin berapa banyak orang yang diharapkan, jadi saya lega karena itu adalah ruangan yang relatif kecil. Saya tidak yakin bagaimana saya akan menangani auditorium sekarang. Tempatnya bertambah, dan kehadirannya luar biasa. Ruangan ini menampung empat puluh, tetapi lebih banyak orang datang, jadi beberapa orang berdiri di samping. Si

energi di dalam ruangan tinggi. Setelah menjelaskan penelitian dan metodenya, saya mengeluarkan ukulele, dan semua orang tampak bersemangat. Saya mencoba yang terbaik untuk tidak terlihat teruk. Di sini tidak ada apa-apa. Saya mulai bermain, lalu bernyanyi. Itu berjalan dengan sangat baik: jauh lebih baik dari yang saya harapkan. Itu bukan penampilan yang layak untuk Carnegie Hall, tentu saja, tetapi saya memainkan seluruh lagu tanpa mengacaukan akord, kehilangan ketukan, atau melupakan kata-katanya. Penonton tersenyum, mengetuk kaki mereka, menertawakan liriknya, dan bertepuk tangan di akhir. Pelatihan saya berhasil: alih-alih melihat fretboard atau senar sepanjang waktu, saya dapat melihat ke kerumunan dan menikmati momen itu.

Memilih Jari

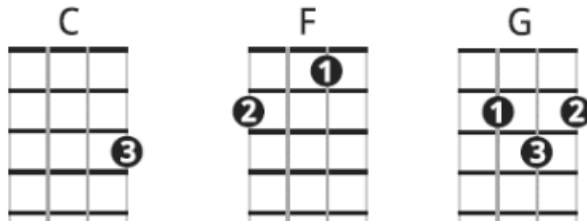
Setelah lagu selesai, saya menunjukkan kepada penonton trik lain yang Igured keluar saat saya berlatih: Anda dapat memainkan lagu-lagu yang terdengar rumit jika Anda memetik senar individu dalam urutan alih-alih memetik semuanya pada saat yang sama. Akornya persis sama. Teknik ini disebut finger picking, dan ada banyak cara Anda bisa melakukannya. Pola yang saya tunjukkan sangat sederhana: memetik senar ke-1, senar ke-4, senar ke-2, dan senar ke-3 pada ketukan, dalam urutan itu. Dengan mengulangi pola ini saat tangan kiri memainkan akord, setiap nada secara alami melengkapi nada berikutnya. Hasilnya terdengar bagus dan kompleks, meskipun tidak lebih rumit daripada memetik. Pembicaraan itu sukses, dan menyenangkan di samping itu. Saya senang saya mengambil risiko, mendorong diri saya sendiri, dan berlatih dengan baik. Beralih dari nol ke pertunjukan publik pertama dengan total sepuluh jam latihan sama sekali tidak buruk.

Satu-Empat-Lima

Secara kebetulan, saya kebetulan bertemu dengan sesama peserta WDS, Melissa Dinwiddle, di luar auditorium utama. Melissa adalah seorang seniman, dan dia kebetulan membawa ukulele-nya, jadi kami mengeluarkan instrumen dan macet sebentar. Salah satu hal yang Melissa ajarkan kepada saya hari itu adalah struktur lagu yang disebut "12 bar blues," yang didasarkan pada progresi akord yang disebut sebagai satu-empat-ve. Tanpa terlalu banyak membahas teori musik, setiap nada memiliki akord terkait, serta "keluarga" akord yang melengkapinya. Jika Anda dapat memainkan akor akar, serta akord dalam keluarga terkait, Anda tiba-tiba dapat memainkan ribuan lagu umum. Ambil, misalnya, "Twinkle, Twinkle, Little Star." Seluruh lagu dapat dimainkan menggunakan tiga akord, yang mengikuti perkembangan ini:

C / F / G

Berikut adalah diagram akor:



Dalam hal ini, C adalah akar, atau akord I. F dan G adalah dua akord lainnya dalam keluarga ini. F adalah akord IV, dan G adalah akord V. Hanya dengan menggunakan tiga akord ini, Anda dapat memainkan segala macam lagu menyenangkan, seperti:

"Lagu ABC"
"Frosty the Snowman"
"Lagu Elmo"

Itu berarti saya bisa mengungkapkan cara memainkan lagu-lagu anak-anak untuk Lelaketika saya sampai di rumah, serta beberapa lagu blues yang menyenangkan. Sepuluh jam latihan saya berikutnya dikhususkan untuk menjelajahi semua lagu baru yang dapat saya mainkan. Antara mengerjakan *The Daily Ukulele* (Belo, 2010), mengambil pola memetik baru dari *AbsoluteBeginners Ukulele* (2009), dan mempelajari teori musik yang lebih dasar dari *Fretboard Roadmaps—Ukulele* (2006), jam y.Tom Hodgkinson benar: jika Anda mengambil ukulele, Anda tidak akan pernah bosan lagi.

Meninjau Metode

Mari kita tinjau inti dari metode yang saya gunakan untuk mempelajari cara memainkanukulele:

Saya memperoleh ukulele, senar, tuner, dan peralatan lain yang diperlukan. Saya memutuskan untuk memulai dengan fokus mempelajari progresi akord yang paling umum dan pola memetik, yang memungkinkan saya memainkan ribuan lagu populer. Setelah saya bisa memainkan akord dan memetik tanpa melihat fretboard atau string, saya berlatih sampai saya dapat mempertahankan pola dan mengubah akord tanpa memikirkannya. Ketika saya merasa nyaman dengan akord dasar dan pola memetik, saya mulai belajar lagu dengan bersenandung bersama saat saya bermain, belajar di mana kata-kata dan perubahan akord tumpang tindih, lalu menambahkan lirik sampai saya bisa bernyanyi dan bermain pada saat yang bersamaan. Saya bereksperimen dengan nger memilih progresi akord yang sama alih-alih memetik, yang memungkinkan saya memainkan lagu yang lebih kompleks. Saya mempelajari progresi akord I, IV, V, yang membantu saya mempelajari cara memainkan lebih banyak lagu.

Ke Mana Saya Pergi Dari Sini

Saya suka bermain ukulele. Antara *The Daily Ukulele Songbook* dan mencari lagu online, saya dapat menemukan cara memainkan hampir semua lagu yang saya inginkan. Beberapa akord lebih sulit daripada yang lain, dan ada ribuan pola memetik dan memetik nger untuk dipelajari, yang akan membuat saya sibuk untuk waktu yang lama.

Salah satu bagian terbaik tentang bermain ukulele adalah bertekanan rendah. Selain eksperimen awal saya, saya tidak bermain untuk mempersiapkan pertunjukan, atau karena saya berharap menjadi musisi profesional di masa depan. Ketika saya mengambil ukulele, saya bisa bersantai dan berkeliling sebentar, mempelajari lagu baru, teknik memetik nger, atau pola memetik. Lela sudah cukup tua untuk mengingat lagu-lagu sekarang, dan Oscar Schmidt saya yang berdebu tua sangat berguna: sekarang ukulele Lela, dan itu berfungsi sebagai umpan saat saya berlatih, atau sebagai instrumen utama kami ketika dia ingin duduk di pangkuan saya dan memetik senar saat saya mengganti akord. Sekarang setelah saya bisa memainkan lagu-lagu yang dia kenali, dia meminta untuk memainkan "ukuyaya" sebelum kami tidur. Sangat menyenangkan memiliki musik dalam hidup kita.



9Windsurfing

g

Pelajaran: Lingkungan Penting

Kita bergulat bukan dengan esh dan darah, tetapi dengan epoksi, karbon ber, monolm, dan ego, mimpi, dan kehendak kita sendiri.—CHRIS ZEITVOGEL, PESELANCAR ANGIN

...

Untuk gambar, video, dan komentar tambahan tentang bab ini,
visithttp://first20hours.com/windsurfing.

Saya berada di tengah danau, berusaha mati-matian untuk mengangkat layar saya. Ini dikultik: otot-otot di lengan, kaki, dan punggung saya berteriak. Angin barat laut yang kencang menendang ombak jambul kecil, memantulkan papan layar saya bolak-balik di bawah saya. Aku membiarkan layar jatuh kembali ke dalam air dan berhenti, mengambil waktu sejenak untuk beristirahat dan menyesuaikan diri dengan goyangan papan di bawah kakiku. Saya melihat ke barat: badai bertiup, mendorong awan gelap di atas pegunungan. Saatnya untuk kembali masuk. Saya meraih ke bawah, mengambil tali uphaul, lalu menarik, mengangkat layar saya keluar dari air setidaknya untuk ketiga puluh kalinya dalam tiga puluh menit terakhir. Saya harap saya bisa kembali ke darat sebelum badai melanda. Ini akan dekat ... Saya tidak pandai mengarahkan alat aneh ini, tetapi saya harus belajar. Mendayung kembali ke pantai menjadi tua dengan cepat.

Kehidupan di atas air

Saya memiliki kenangan yang sangat indah tentang musim panas yang dihabiskan di atas air. Tumbuh dewasa, saya sangat terlibat dengan Boy Scouts of America, dan menghabiskan beberapa musim panas melayani sebagai anggota sta di dua kamp di Ohio Utara: Camp Firelands dan Camp Avery Hand. Selama musim panas terakhir saya di perkemahan, saya mendapat kehormatan untuk melayani sebagai asisten direktur akuatik di bawah "Aqua" legendaris Bob Sliney. Bob secara tradisional memimpin program kolam renang: berenang, menyelamatkan nyawa, dan berenang mil. Saya bertanggung jawab atas semua jenis kegiatan perahu: kano, mendayung, berlayar, dan berperahu motor. Saya menyukai setiap menit perkemahan. Setiap hari, saya harus menghabiskan sebagian besar waktu saya di atas air, mengajari Pramuka yang lebih muda cara menikmati menggunakan perahu dengan aman. Itu adalah pekerjaan yang menantang dan melelahkan dengan semua cara terbaik. Ketika saya kuliah, saya meninggalkan air. Pekerjaan perusahaan pertama saya di Cincinnati memakan waktu, dan Sungai Ohio, perairan besar terdekat, tidak ideal untuk berperahu rekreasi. Kemudian, ketika Kelsey dan saya pindah ke tengah New York City, yang paling dekat dengan berperahu adalah naik gondola di Central Park. Ketika kami menetap di Colorado, saya merasa yakin hari-hari berperahu saya telah selesai: bagaimanapun, bagian dunia ini tidak benar-benar dikenal karena kekayaan airnya. Namun, baru-baru ini, saya dapat mengatur akses ke danau pribadi kecil di utara kota, beberapa mil dari rumah. Saya pusing: ini adalah waktu ketiga dalam bertahun-tahun saya akan bisa menghabiskan waktu di atas air. Apa yang harus saya lakukan rst?

Menangkap Serangga Windsurng

Ide pertama saya adalah untuk mendayung: danau ini cukup besar untuk mendayung. Sayangnya, perahu dayung standar besar dan besar, dan bahkan sculls terkecil sangat panjang dan relatif rapuh,

yang menyakitkan, karena saya tidak memiliki kemampuan untuk menyimpan kapal besar atau panjang dengan aman di danau. Masalah yang sama berlaku untuk perahu layar, dengan kelemahan tambahan dari biaya: bahkan peralatan dasar pun harganya cukup mahal. Selain komplikasi tambatan, saya lebih suka tidak menjatuhkan sepuluh grand ongear. Berkano lebih mudah dan lebih murah, tetapi tidak terlalu menyenangkan untuk dilakukan sendiri: panjang sebagian besar kano membuatnya lebih baik untuk mendayung dua orang. Perahu motor dan Jet Ski tidak mungkin, karena pemilik danau membatasi motor hingga 9 tenaga kuda atau kurang, dan saya lebih suka kerajinan bertenaga manusia pada umumnya. Saat saya meneliti pilihan saya, saya kebetulan menemukan video yang direkam oleh James Douglass, seorang profesor biologi kelautan di Florida Gulf Coast University. Selain keahlian biologinya, Jim adalah peselancar angin ulung yang menerbitkan informasi yang sangat baik tentang olahraga ini di blog pribadinya.¹ Dalam video tersebut, Jim memasang papan layar kelas Formula dengan kamera video HD tahan air, yang terpasang pada ujung boom layar, menangkap semua aksi sekaligus.² Dia kemudian berjalan di atas air, melompati ombak kecil dan mengeksekusi dengan cepat, belokan yang mengesankan. Saya ketagihan, dan saya menonton video itu beberapa kali berturut-turut. Saya belum pernah naik layar sebelumnya, tetapi tiba-tiba saya ingin mencoba. Windsurfg memenuhi semua kriteria saya: Saya bisa melakukannya sendiri, dan perlengkapan yang diperlukan tidak besar atau berat. Ini juga, berdasarkan penelitian misterius, tidak terlalu mahal: Saya perlu membeli perlengkapan, tetapi pengaturan dasar tidak akan menguras kantong. Windsurfg terlihat menyenangkan, tetapi sebelum saya terjun, saya ingin memastikan itu aman. Saya pada dasarnya bukan pecandu adrenalin atau pencari sensasi. Jika windsurfg ternyata berbahaya, saya akan sama bahagiannya melakukan sesuatu yang lain.

Jika saya akan berlayar, saya ingin tahu risikonya sebelum saya berkomitmen. Apa yang bisa salah?

Apakah selancar angin berbahaya?

Windsurfg sering diberi label sebagai olahraga "ekstrem", tetapi tingkat ekstremitasnya tergantung pada jenis sailboarding yang ingin Anda lakukan. Danau-danau kecil yang tertutup dari jenis yang akan saya pelajari, itu lebih intens daripada mengemudikan perahu layar kecil, tetapi itu tidak gila. Windsurfg di lautan, di sisi lain, membawa risiko yang jauh lebih signifikan. Ombak yang menghantam pantai membuat peluncuran lebih sulit, dan kehilangan rig Anda jauh dari pantai adalah masalah yang sangat besar. Lautan menambahkan elemen kesenangan, seperti melompat ombak, tetapi juga menambah risiko. Dibandingkan dengan waterboarding laut, windsurfg di danau pedalaman sangat aman. Apa pun jenis sailboarding yang Anda lakukan, bagaimanapun, ada risiko yang signifikan, terutama, tenggelam dan hipotermia. Tidak ada yang bisa menghindarinya: ketika Anda berdiri terbuka di atas oatingpiece Styrofoam di tengah perairan besar, memegang layar besar tegak dengan tangan pada hari yang berangin, hal-hal buruk dapat terjadi jika Anda tidak siap. Risiko tenggelam memiliki dua faktor utama: cedera dan kelelahan. Jika Anda jatuh dengan cara yang mengakibatkan cedera besar anggota tubuh atau gegar otak, Anda berisiko kehilangan kesadaran di dalam air. Jika Anda lelah, Anda tidak bisa kembali ke papan atau kembali ke pantai, tenggelam mungkin terjadi. Cara terbaik untuk mencegah tenggelam adalah dengan (1) memakai perangkat personalotasi (sering disebut PFD atau jaket pelampung); dan (2) selalu pergi keluar dengan seseorang di dekatnya, baik di air atau di pantai. Jika Anda mendapat masalah, Anda menginginkan seseorang di dekatnya yang dapat membantu.³Hipotermia bisa sama mematikannya. Dalam kondisi dingin, basah, dan berangin, tubuh Anda kehilangan panas dengan sangat cepat. Setelah suhu tubuh inti Anda turun di bawah sembilan puluh derajat Fahrenheit (tiga puluh

derajat Celcius), jantung, paru-paru, dan sistem saraf Anda mulai mati, menyebabkan kematian kecuali suhu tubuh inti dikembalikan ke tingkat operasi normal, yaitu antara sembilan puluh delapan dan seratus derajat Fahrenheit. Apa yang membuat hipotermia sangat berbahaya adalah bahwa itu terjadi secara bertahap. Saat suhu tubuh inti turun, gejala seperti menggigil, kebingungan, kehilangan koordinasi, dan kelelahan muncul, mengganggu ketangkasan fisik dan penilaian. Jika Anda mengandalkan penilaian yang baik dan keterampilan fisik untuk membawa Anda pulang, timbulnya hipotermia adalah ancaman besar yang terlalu mudah untuk diabaikan. Pada suhu air di atas enam puluh dua derajat Fahrenheit, hipotermia bukanlah faktor risiko yang besar. Di bawah ambang batas itu, ada baiknya memastikan Anda memiliki insulasi sebelum Anda mengambil risiko memasuki air. Di situlah pakaian basah masuk. Pakaian basah terbuat dari bahan tipis dan mudah dipisahkan yang mengisolasi di dalam air, seperti neoprene. Pakaian selam modern dinilai dalam hal ketebalan milimeter, dan biasanya dirancang agar lebih tebal di batang tubuh daripada anggota badan. Desain ini melayani dua tujuan: lebih banyak insulasi di sekitar batang tubuh menjaga panas tubuh dengan lebih efektif, sementara lebih sedikit bahan di sekitar tungkai mempertahankan jangkauan dan kemudahan gerakan. Untuk windsurfing dalam kondisi dingin, Anda membutuhkan insulasi dan eksibilitas. Kombinasi dua pakaian basah memberikan jangkauan perlindungan terluas dalam kondisi cuaca umum. Setelan basah "shorty" lengan pendek, yang menjaga lengan dan kaki tetap terbuka, paling baik untuk suhu yang lebih hangat: 3 milimeter di batang tubuh dan 2 milimeter di anggota badan (3/2) adalah sufficient. Pada suhu air di bawah enam puluh derajat, setelan basah 5/4 seluruh tubuh dengan sepatu bot, sarung tangan, dan neoprene hood atau topi adalah yang terbaik. 4 Daerah pesisir California adalah tujuan angin yang populer, jadi departemen berperahu dan saluran air negara bagian menyusun daftar tips keselamatan yang praktis: 5

1. Pertimbangkan prakiraan cuaca dan pasang surut setempat.
2. Selalu beri tahu seseorang tentang di mana Anda berencana untuk berlayar dan kapan Anda berharap untuk kembali.
3. Kenakan pakaian yang sesuai dengan kondisi.
4. Kenakan jaket pelampung yang disetujui Penjaga Pantai AS dengan peluit terpasang.
5. Dalam kondisi panas, cerah, lembab, minum banyak air.
6. Periksa peralatan Anda apakah ada tanda-tanda kerusakan atau kelelahan.
7. Berlayar dengan teman.
8. Ketika angin bertiup ke darat, jangan berlayar lagi.
9. Dingin bisa membunuh. Pertama kali Anda menggigil, kembali ke pantai dan melakukan pemanasan.
10. Selalu tetap dengan papan Anda—jangan pernah mencoba berenang ke pantai.

Negara bagian juga menyediakan daftar periksa pra-peluncuran sederhana: Sebelum Meluncurkan ...

1. Periksa kembali tali pengaman Anda.
2. Waspadalah terhadap awan gelap di cakrawala—badai melanda dengan cepat.
3. Jika ragu, jangan keluar.
4. Seorang pelaut yang cerdas akan selalu mencoba mengambil tindakan yang paling aman sebelum penyelamatan adalah satu-satunya jalan keluar.

Itu akal sehat, tetapi itu penting. Windsurging itu menyenangkan. Mati tidak. Namun, dengan sedikit persiapan dan perencanaan, risiko utama windsurging dapat diminimalkan.

Di Mana Saya Memulai?

Seperti yang Anda duga, Colorado utara bukanlah titik panas angin utama. Ada angin yang cukup untuk membuatnya bermanfaat jika Anda memiliki tempat untuk berlatih, tetapi berbeda dengan tujuan utama seperti Columbia River Gorge Washington, angin lokal bervariasi dan tidak dapat diprediksi, tidak kencang dan berkelanjutan. Selain itu, badan besar air yang dapat diakses tidak terlalu umum di Colorado utara. Karena sebagian besar Front Range adalah dataran tinggi, stepa semi-kering, badan air besar seperti danau tidak terbentuk secara alami. Sebagian besar "danau" di daerah tersebut adalah waduk buatan yang dirancang untuk menampung air bagi petani lokal, dialiri oleh sungai seperti alam liar

dan Cache La Poudre yang indah, yang membawa salju runo turun dari pegunungan ke dataran di bawahnya. Akibatnya, geografi lokal mendukung pemandangan kayak air putih yang sangat berkembang, tetapi tidak banyak angin. Saya kebetulan tidak mengenal siapa pun yang berselancar angin secara teratur, dan sepertinya tidak ada pengecer lokal yang menjual papan atau peralatan. Demikian juga, setelah mencari instruktur di daerah tersebut, sepertinya saya sendiri. Tempat terdekat yang bisa saya temui yang memberikan pelajaran adalah di Denver, satu jam berkendara. Itu membuat penjadwalan rumit: beberapa hari berangin, dan beberapa hari tidak, jadi menjadwalkan pelajaran akan membawa risiko pembatalan yang signifikan. Meskipun akan ideal untuk memulai dengan beberapa instruksi pribadi, pekerjaan dan tanggung jawab keluarga membuat saya dekat dengan rumah untuk saat ini. Saya yakin bahwa jika saya bisa mendapatkan peralatan yang tepat, saya dapat mengambil teknik ini begitu saya berada di atas air. Itu besar jika: bagaimana cara membeli perlengkapan jika tidak ada toko di dekatnya? Dari apa yang saya pahami, papan dan layar windsurung lebih besar daripada paket yang biasanya dikirim oleh perusahaan seperti FedEx dan UPS. Jika saya tidak bisa mendapatkan peralatan yang sesuai, proyek ini akan selesai sebelum dimulai.

Bersiap

Sumber informasi pertama saya adalah situs web Jim, yang memiliki banyak posting cara untuk pemula. Salah satu esai pertama yang saya baca adalah postingan berjudul "18 Pertanyaan Windsurung Teratas Dijawab,"6 yang mencakup tutorial tentang peralatan yang diperlukan. Dari Tanya Jawab, sepertinya saya akan membutuhkan, setidaknya, papan, layar, PFD, dan pakaian selam. Saya membaca setiap posting di situs Jim, membuat daftar berbagai perlengkapan yang terlihat berguna, mencoba mengeditnya menjadi apa yang saya yakini sebagai dua opsi terbaik, lalu menulis email kepada Jim untuk mengucapkan terima kasih, serta meminta sarannya.

Yang paling penting adalah jenis papan dan layar yang harus saya dapatkan. Video Jim tentang berkeliling di papan kelas Formula sangat mengesankan, dan saya ingin sampai ke titik itu, tetapi saya tidak yakin apakah memulai dengan papan Formula adalah ide yang bagus. Pilihan lain, dari penelitian saya, adalah papan berpengalaman luas yang disebut Rio, yang diproduksi oleh perusahaan bernama Starboard. Rio mengembangkan reputasi yang sangat ramah pemula, serta papan serba guna yang baik untuk sebagian besar kondisi: tidak akan secepat formula, tetapi akan bekerja dalam kondisi yang lebih luas. Saya tidak tahu apa yang harus dilakukan, jadi saya meminta nasihatnya kepada Jim:

Saya benar-benar baru mengenal windsurfg—belum menginjak papan. Saya berharap Anda dapat memberi saya beberapa saran untuk memulai. Saya ingin mengambil satu set perlengkapan lengkap, dan saya mencoba memutuskan antara Starboard Rio Medium dan Starboard Formula 167. Rio terdengar seperti papan pemula yang bagus, tetapi saya ingin belajar [hydroplane] secepat mungkin. Saya hanya akan berangin di air, dan saya ingin bisa berlayar dengan angin yang lebih sedikit (4-5 knot). Papan formula, dari apa yang saya baca sejauh ini, [hydroplane] lebih cepat dan bekerja lebih baik dalam angin yang lebih sedikit. Inilah pertanyaan saya: apakah gila belajar di papan Formula? Saya memiliki toleransi yang sangat tinggi untuk frustrasi awal, dan saya lebih suka membeli satu papan daripada beberapa papan. Jika saya belajar dengan Formula, apakah saya berisiko merusak peralatan atau terluka? Terima kasih atas saran apa pun yang dapat Anda sampaikan—saya menghargai bantuan Anda!

Satu jam atau lebih kemudian, Jim menjawab:

1. Hal tentang papan formula yang lebih baik dalam angin ringan menyedatkan. Mereka memang memiliki potensi untuk terbang dalam angin yang lebih ringan daripada papan lain (7-8knots), tetapi Anda harus menjadi pelaut yang sangat baik menggunakan layar besar 11-12 meter persegi untuk memanfaatkan potensi itu. Dan papan formula ketika TIDAK merencanakan adalah anjing karena sangat pendek dan lebar dan tidak memiliki papan belati. Untuk kondisi non-planing (secara realistis, apa pun di bawah 10 knot) papan yang lebih panjang dan lebih sempit dengan papan belati (Seperti Rio M) jauh lebih cepat dan lebih mudah dinavigasi daripada papan formula.2. Anda tidak mungkin melukai diri sendiri saat mencoba belajar di papan formula, kecuali jika itu melukai punggung Anda saat mencoba mengangkat layar yang terlalu besar. Alasan Anda tidak mungkin melukai diri sendiri adalah karena Anda mungkin tidak akan bisa membuat papan pergi sama sekali! Namun, Anda bisa melukai papan dengan sangat baik, karena papan formulamemiliki kulit tipis yang akan mudah berwarna saat Anda menjatuhkan rig selama jatuh.3. Rio M mungkin adalah papan terbaik yang pernah ada untuk belajar cepat dan merasa nyaman merencanakan dan menggunakan tali kaki, meskipun GO 171 juga akan

jadilah baik.⁴ Strategi saya adalah memiliki beberapa layar berukuran berbeda sehingga Anda dapat mendorong batas berapa banyak kekuatan yang dapat Anda tangani terlepas dari kekuatan angin. Untuk anak tangga pertama dan untuk angin kencang nanti, Anda akan menginginkan asail yang lebih kecil dari 6 meter persegi ... Saya memiliki kalkulator windsurging yang akan memberi Anda gambaran tentang ukuran layar yang Anda perlukan untuk memiliki kesempatan merencanakan kekuatan angin tertentu.⁷⁵ Peluang terbaik Anda untuk merencanakan adalah dengan layar yang lebih kecil saat angin kencang, karena membutuhkan lebih sedikit teknik untuk terbang dengan layar kecil dalam angin kencang daripada dengan layar besar dalam angin ringan.

Jenis informasi ini adalah emas mutlak. Jim menjernihkan beberapa kesalahpahaman yang saya miliki tentang jenis papan yang harus saya mulai, serta jenis layar yang harus saya mulai. Kebajikan Jim dalam membagikan nasihatnya menghemat saya beberapa ribu dolar dan banyak frustrasi. Jim juga memperkenalkan saya pada Isthmus Sailboards,⁸ sebuah toko di Madison, Wisconsin, yang menjual perlengkapan angin secara online. Saya menelepon Isthmus, dan Gary Stone, salah satu pemiliknya, membantu saya menyusun daftar perlengkapan yang saya perlukan untuk memulai. Inilah yang akhirnya saya pesan:

Papan kanan RioPapan Chinook Powerglide 4,7 meter persegi
berlayarChinook Sport AL boomChinook 400cm tiangChinook US
tiang ekstensiChinook US 1-baut tiang baseChinook bungee
uphaulMystic Crossre 5/4 full wet suit + neoprene boots, sarung
tangan, dan topiMystic Crossre 3/2 shorty smoking + neoprene
Vibram Five FingersDakine Surface perangkat otasi pribadi

Total investasi: sekitar tiga ribu dolar, termasuk pengiriman. Windsurging tidak murah, tetapi jika saya merawat perlengkapan dengan baik, itu akan bertahan lama. Pada saat pertama, saya condong untuk membeli layar yang lebih besar, tetapi Gary telah mengajari orang-orang cara selancar angin selama lebih dari dua dekade, dan

dengan cepat membujukku keluar dari itu. Layar besar, katanya, bisa bagus untuk kondisi angin yang lebih rendah, tetapi hanya jika Anda tahu cara menggunakannya. Kelemahan dari layar besar adalah mereka lebih berat dan lebih sulit untuk diangkat keluar dari air. Jika saya mulai dengan layar besar saat saya masih terbiasa menyeimbangkan dan berputar, saya akan, dalam kata-kata Gary, "membenci hidup saya." Saya lebih suka tidak membenci hidup saya, jadi saya mengikuti saran Gary dan memilih layar yang lebih kecil. Sekali lagi, ada baiknya untuk mengikuti saran dari mentor yang lebih berpengalaman sebelum membuat keputusan. Setelah setengah jam di telepon dengan Gary, saya memesan Isthmus.As ternyata, perusahaan pengiriman barang komersial mampu dan bersedia mengirimkan benda-benda seukuran papan selancar di mana pun Anda suka, jadi mendapatkan peralatan lebih mudah dari yang saya harapkan. Gary memperkirakan bahwa akan memakan waktu sekitar satu minggu untuk menerima perlengkapan, jadi sementara itu, saya mulai mendidik diri sendiri tentang cara menggunakannya.

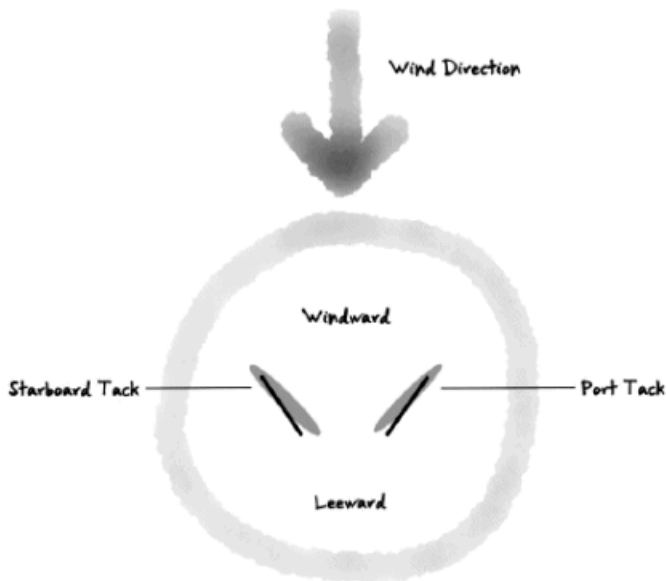
Avast, Kamu Lubbers!

Sedikit waktu menjelajahi Internet membawa saya ke beberapa sumber daya pengantar:

Panduan Pemula untuk Zen dan Seni Selancar Angin oleh Frank Fox (1988)Selancar Angin oleh Peter Hart (2005)Belajar Selancar Angin di Akhir Pekan oleh Phil Jones (1992)Pemula hingga Pemenang (DVD) oleh Jem Hall (2006)

Semua sumber daya ini melakukan pekerjaan yang baik dalam menjelaskan istilah, teori, dan teknik dasar. DVD instruksional Jem Hall sangat bagus dalam menjelaskan cara menangani papan di dalam air: lebih mudah untuk menjelaskan gerakan kompleks dengan mendemonstrasikannya. Buku berada dalam posisi yang kurang menguntungkan di sana, karena bahkan ilustrasi terbaik pun bisa membingungkan untuk diuraikan dibandingkan dengan video panduan yang terperinci. Satu hal yang langsung mengejutkan saya adalah seberapa banyak terminologi pelayaran tradisional yang digunakan dalam windsurfg. Masuk akal, karena

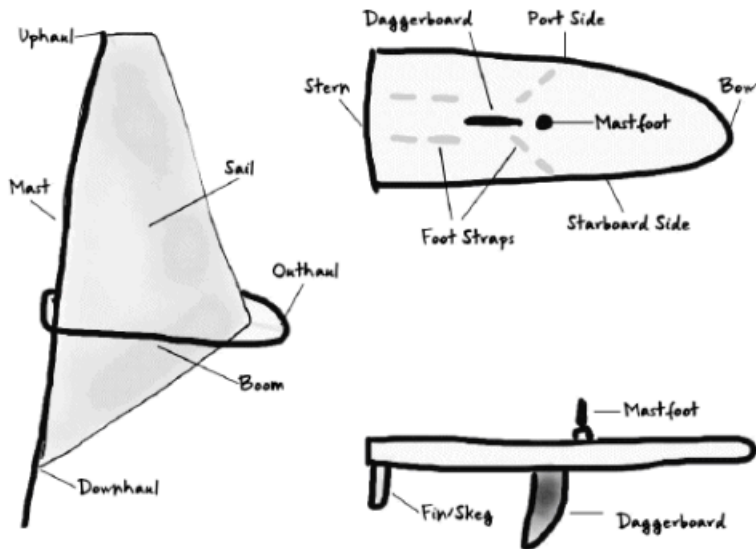
Papan layar pada dasarnya adalah perahu layar yang menggunakan papan selancar alih-alih ahull, tetapi itu masih mengejutkan saya. Peselancar angin disebut sebagai "pelaut", dan istilah yang digunakan dalam instruksi berasal langsung dari berlayar. Sudah beberapa tahun sejak saya berlayar, jadi saya perlu meninjau. Jika Anda tidak tahu istilahnya, membaca buku instruksional bisa membuat frustrasi, jadi penting untuk mengetahui ide-ide kuncinya. Berikut adalah beberapa:



- Arah angin—arah seperti utara, selatan, timur, dan barat tidak terlalu berguna untuk berlayar, karena angin berubah. Arah diberikan relatif terhadap arah angin: angin berarti bergerak ke arah angin yang berlaku, dan angin berarti bepergian dengan angin. Arah kerajinan—demikian juga, "kiri" dan "kanan" relatif terhadap orientasi pesawat saat ini. Port berarti sisi kiri kapal, sedangkan kanan berarti kanan. Depan dan belakang—bagian depan kapal disebut haluan, dan bagian belakang disebut buritan. Ke arah haluan disebut ke depan, dan ke arah buritan disebut buritan. Berputar—istilah untuk memutar papan adalah menempel dan jibbing. Perbedaan di antara mereka adalah ujung papan mana yang kebetulan melewati angin selama belokan. Jika Anda bergerak ke arah angin (ke arah angin) dan Anda berbalik,

Anda menempel. Jika Anda menjauh dari angin (ke bawah angin) dan Anda berbalik, Anda sedang jibbing.

Menggabungkan istilah-istilah ini adalah di mana mulai menjadi rumit. Jika Anda mengarah ke arah angin dan Anda memutar haluan kapal ke kanan (ke kanan), Anda berada di "paku pelabuhan", karena angin datang di sisi kiri (pelabuhan) kapal. Ini membingungkan, jadi saya membutuhkan beberapa jam untuk memecahkan kode instruksi dalam buku dan membayangkannya di atas air. Proses simulasi mental ini akan sangat membantu ketika saya berada di atas air: dengan membayangkan konsep yang diterapkan pada kerajinan sungguhan, saya membuatnya lebih mudah untuk mengingatnya ketika itu penting.



Bagian-bagian papan layar juga memiliki nama yang khas:

Papan—kami sudah menutupi haluan, buritan, pelabuhan, dan kanan, tetapi bukan itu saja yang perlu kami ingat. Papan angin biasanya memiliki potongan tambahan, seperti *n* besar di tengah (papan belati) dan *n* yang lebih kecil di buritan (sirip atau skeg), yang keduanya memanjang ke dalam air. Di bagian atas papan, ada tempat untuk memasang layar (mastfoot), yang diamankan oleh alas mekanis (mast base).

Biasanya juga ada tali kaki di bagian atas papan, yang membantu pelaut tetap seimbang dan berada di posisi yang benar saat papan benar-benar bergerak. Layar berlayar kira-kira segitiga, dan melekat pada papan melalui tiang vertikal besar yang disebut tiang. Tiang dimasukkan ke dalam saku di layar (selongsong tiang) yang memanjang dari kaki tiang ke atas (kepala). Setelah tiang terpasang, pegangan horizontal (boom) ditempatkan kira-kira tegak lurus dengan tiang yang diamankan ke tiang dengan penjepit. Tiga sisi layar diikat ke boom mastand dengan tali nilon (garis), yang dikencangkan dengan membungkusnya di sekitar ujung boom, kemudian memasukkannya melalui perangkat penahan tali khusus yang disebut clew. Setelah layar dipasang sepenuhnya, tiang dipasang ke tiang, membuat papan siap untuk berlayar.

Mencurangi papan dengan benar membutuhkan penilaian dan pengalaman. Ada beberapa cara untuk memasang dasar tiang ke papan, dan setiap variasi mengubah pusat gravitasi papan. Tergantung pada papan, mungkin juga ada cara berbeda untuk memasang tali kaki buritan, yang semuanya sesuai dengan kinerja papan di atas air. Demikian juga, cara Anda memasang layar berdampak besar pada cara papan menangani. Dua variabel terbesar adalah downhaul dan outhaul, yang mengacu pada seberapa kencang Anda membuat tali yang menahan layar ke bagian bawah tiang dan sisi jauh boom. Lessdownhaul dan outhaul menciptakan "kantong" yang lebih besar di layar, yang berarti layar mampu menangkap lebih banyak angin, menghasilkan lebih banyak tenaga. Menambahkan lebih banyak downhaul dan outhaul mengurangi ukuran saku, mengurangi potensi daya, yang membuat layar lebih mudah dikelola dalam kondisi yang sangat berangin.

Mundur, saya akan melakukan fisika!

Layar adalah perangkat yang menarik. Bertentangan dengan kepercayaan populer, papan layar (dan perahu layar pada umumnya) tidak bergerak karena angin mendorong layar. Cerita lengkapnya sedikit lebih rumit dari itu. Layar bekerja dengan menciptakan perbedaan tekanan udara di bagian depan dan belakang layar. Dalam sebagian besar kondisi, saat udara mengalir di sekitar layar, theow menciptakan area tekanan rendah di depan (ke arah haluan),

dan area tekanan tinggi di belakang (ke arah buritan). Gabungan dari dua zona tekanan yang berbeda ini menciptakan gayayang menggerakkan pesawat ke arah area tekanan rendah. Akibatnya, angin menarik layar sebanyak mendorongnya. Airplanewings bekerja dengan cara yang hampir sama. Itu penting untuk diketahui saat Anda berada di atas air, mencoba bergerak. Jika model mental Anda tentang cara kerja layar adalah "maksimalkan jumlah layar yang dapat ditiup angin", Anda akan kesulitan menambahkan kekuatan, serta berlayar ke arah mana pun selain di mana angin bertiup. Semua informasi ini sulit untuk dilacak, tetapi saya senang saya memiliki sedikit waktu untuk mempelajari teori dasar sebelum saya mencoba menyerang. Windsurng pada dasarnya adalah keterampilan motorik, tetapi jika saya tidak mengerti cara kerja kerajinan, saya berisiko menghabiskan banyak waktu untuk berlatih gerakan yang salah.

Mengamati Angin

Selain membaca buku, saya juga mulai memperhatikan cuaca. Sebelum tertarik pada angin, saya tidak pernah memperhatikan angin kecuali bertiup sangat keras. Sekarang, saya terus menontonnya terus-menerus, meneliti puncak pohon untuk memperkirakan kecepatan angin. Teknologi membantu di sini: situs web seperti Weather Underground,⁹Windnder,¹⁰ dan iWindsurf¹¹ mengumpulkan informasi angin dari lokasi di seluruh dunia, membuatnya lebih mudah untuk memeriksa kecepatan angin di berbagai tempat. Tentu saja, satu-satunya angin yang benar-benar penting adalah angin yang mengalir di atas air tempat Anda ingin berselancar angin. Untuk memeriksa kecepatan dan arah angin di danau, saya mengambil Kestral 3000 Pocket WindMeter.¹² Perangkat kecil yang praktis ini memungkinkan saya memeriksa kecepatan angin saat ini di danau, sehingga lebih akurat daripada informasi online.

Unit ini juga mampu mengukur suhu udara dan air, jadi saya dapat memastikan saya tidak akan keluar jika risiko hipotermia tinggi. Saat saya memperhatikan angin selama beberapa hari, saya mulai memperhatikan pola. Di danau, angin ringan cenderung datang dari timur di pagi hari, kemudian meningkat dan bergeser untuk bertiup dari utara-barat laut pada sore hari, biasanya sekitar pukul empat sore. Karena saya akan meluncur dari sisi timur danau, yang terbaik adalah keluar di sore hari. Empat juga merupakan waktu yang tepat untuk jadwal saya secara umum: jika saya merencanakan hari kerja saya dengan baik, saya dapat keluar selama satu jam atau lebih tanpa gangguan besar.

Anda Tidak Bisa Berselancar Angin Tanpa Angin

Saya juga memperhatikan bahwa angin tidak terlalu dapat diprediksi di sini: Ada banyak variabilitas. Beberapa hari berangin, dan beberapa hari tidak. Berdasarkan apa yang saya baca, jika angin kurang dari enam mil per jam, itu tidak layak untuk dipasang. Itu berarti saya akan membutuhkan rencana cadangan. Untungnya, saya punya ide: Saya telah melihat orang-orang di danau dengan papan dayung berdiri, yang juga terlihat seperti aktivitas menyenangkan yang juga memenuhi kriteria berperahu saya. Jika saya menghasilkan kekuatan saya sendiri dengan dayung, saya tidak membutuhkan angin, jadi paddleboarding akan menjadi alternatif yang baik pada hari-hari ketika saya tidak bisa berselancar angin. Saya sudah memiliki PFD dan dua pakaian selam, jadi yang saya butuhkan hanyalah papan dayung berdiri yang layak dan dayung panjang. Setelah sedikit penelitian, saya memilih peralatan berikut:

Ocean Kayak Nalu 11 'paddleboardQuick
Blade Kahana Elite 80 " dayung

Paddleboard dikirimkan dengan truk barang yang sama dengan mysailboard. Di antara papan layar dan papan dayung saya, saya dapat menghabiskan waktu di atas air terlepas dari kondisi angin.

Menyatukan Potongan-potongan

Sekarang setelah saya memiliki semua perlengkapan saya, saya perlu mencari cara menyatukan papan dan berlayar, sebuah proses yang disebut tali. Untungnya, Isthmus telah membantu saya. Rigging adalah area masalah umum bagi pemilik papan layar baru, karena ini adalah proses rumit yang melibatkan beberapa bagian yang berbeda. Saat Anda mencoba mencari tali mana yang menuju ke mana, mudah untuk bingung. Isthmus memecahkan masalah ini dengan menyediakan video rigging-rigging-video terperinci,¹³ sehingga Anda dapat menonton saat seorang profesional memasang papan secara real time. Video instruksional sangat ideal untuk jenis pembelajaran ini: Saya menonton instruksi beberapa kali dan membuat catatan sebelum mencobanya dengan perlengkapan saya sendiri. Berbekal catatan saya, saya dapat memasang papan saya dalam waktu sekitar tiga puluh menit: tidak buruk untuk upaya pertama saya. Saya mengenakan pakaian basah saya, papan saya sudah siap, dan angin bertiup. Saatnya meluncurkan.

Pelayaran Perdana

Saya membawa papan saya dan berlayar ke tepi air, menempelkannya ke alasnya, lalu mengarahkan rig ke danau sampai airnya setinggi pinggang: cukup dalam untuk menurunkan papan belati. Angin datang dari barat laut dengan kecepatan sekitar dua belas mil per jam, menciptakan sedikit potongan yang bagus di permukaan danau. Tidak masalah: Saya punya ini. Saya mendorong diri saya ke papan, berdiri, lalu membungkuk untuk meraih uphaul, tali bungee dikepang yang menempel pada tiang. Dengan menarik uphaul, saya dapat menarik layar keluar dari air sampai lurus, kira-kira tegak lurus dengan papan. Saat layar meninggalkan air, outhaul secara alami bergerak ke arah tolee, menjauh dari angin. Itu memungkinkan saya untuk mengangkat layar tanpa menambah tenaga, jadi saya tetap kira-kira di tempat saya berada, memegang tiang dengan kedua tangan. Posisi "netral" ini adalah titik awal dari angin, jadi saya merasa cukup baik tentang diri saya sendiri. Sejauh ini, saya baik-baik saja.

Urutan bisnis saya berikutnya adalah membungkus layar: gerakkan salah satu tangan saya ke tiang, pegang boom, dan tarik. Dengan melakukan ini, saya akan menambahkan daya, dan mulai bergerak. Di sini tidak ada apa-apa ...

Di bawah belas kasihan Ibu Pertiwi

Saya tidak siap untuk apa yang terjadi selanjutnya. Saat saya menarik ledakan ke arah saya, beberapa hal terjadi sekaligus. Papan mulai bergerak lebih cepat dari yang saya harapkan, dan dikombinasikan dengan goyangan papan di bawah kaki saya, saya panik dan kehilangan keseimbangan. Pusat gravitasi saya bergeser ke belakang, dan sebelum saya bisa memproses apa yang sedang terjadi, saya berada di dalam air. Tangan saya masih memegang boom, jadi hal terakhir yang saya lihat sebelum tenggelam adalah tiang yang bergerak sangat cepat ke arah kepala saya. Apakah saya menyebutkan bahwa saya tidak memakai helm? Saya tidak yakin seperti apa yang saya harapkan jatuh dari papan layar, tetapi tentu saja tidak sekeras ini. Untungnya, tiang itu meleset dari kepala saya sejauh satu kaki, tetapi jatuhnya membuat angin keluar dari saya, dan saya menelan beberapa suap air danau yang jahat. Di bawah permukaan, saya tidak tahu ke arah mana yang naik. Untungnya, saya tidak berada di bawah air lama: PFD saya membawa saya ke permukaan, batuk dan tersendat-sendat. Saat itulah saya menyadari masalah saya berikutnya: Saya tidak bisa melihat apa-apa. Saya mengandalkan kacamata untuk memperbaiki penglihatan saya. Lensa kontak tidak bekerja dengan baik untuk saya, jadi saya menyerah bertahun-tahun yang lalu. Sebelum pergi ke air, saya tidak memikirkan kacamata hitam resep saya, dan tentu saja, saya tidak memakai tali apa pun untuk menempelkannya di kepala saya. Akibatnya, kejatuhan pertama saya merobek kacamata saya dari wajah saya. Mereka saat ini menghiasi dasar danau, tidak pernah terlihat lagi. Saya sudah beberapa menit dalam perjalanan angin pertama saya, dan saya sudah basah kuyup, buta, dan terguncang setelah nyaris menghindari

gegar otak. Ini tidak berjalan dengan baik. Saya mengertakkan gigi dan berenang menuju papan. Pada saat saya dapat mengarahkan kembali diri, angin telah meniupnya sepuluh hingga tujuh belas kaki dari saya. Saya meraih papan, mengangkat diri, dan mengangkat sekali lagi.

Penyelamatan Diri

Dalam empat puluh menit saya berada di atas air hari itu, saya jatuh dengan segala cara yang mungkin: mundur, maju, menyamping. Saya menelan cukup banyak air untuk menyebabkan mual, dan kaki, lengan, dan punggung saya sakit karena mengangkat layar berulang kali. Akhirnya, saya memutuskan sudah cukup. Saya kebingungan, sakit, dan kelelahan. Saatnya pulang. Sayangnya, saya tidak tahu bagaimana mengarahkan papan sialan ini, dan setiap kali saya mencoba mengangkat layar, saya akhirnya menghirup lebih banyak air. Angin meniup saya ke selatan, menjauh dari tempat peluncuran saya. Saya tidak bisa bermanuver kembali ke sana jika hidup saya bergantung padanya. Pada saat itu, saya memutuskan sudah waktunya untuk mempraktikkan teknik penyelamatan diri yang saya pelajari dari salah satu buku yang saya ambil. Saya berbaring di papan dan mengatur layar sedemikian rupa sehingga diletakkan di atas saya, dengan bagian atas layar mengarah ke belakang ke buritan. Kemudian, saya mulai mendayung dengan tangan saya, perlahan-lahan menarik diri saya ke pantai. Itu berjalan lambat. Saya tidak terlalu jauh, tetapi saya butuh sepuluh menit mendayung keras untuk kembali. Senjata tidak terlalu kuno bahkan dalam keadaan terbaik, dan ini bukan keadaan terbaik. Ketika saya mendarat, cobaan saya belum berakhir. Karena saya cukup jauh dari posisi peluncuran saya, itu berarti tamasya pertama saya berakhir dengan "jalan malu" yang tepat. Saya memaksa tubuh saya yang kelelahan untuk membawa perlengkapan saya kembali ke mobil, pulang, dan pingsan. Bukan awal yang menguntungkan.

Analisis Peristiwa Pasca-Trauma

Setelah istirahat, saya meninjau tamasya. Apa yang terjadi di luar sana? Apa yang salah? Pertama, kondisi angin terlalu kencang untuk pemula mutlak. Untuk seseorang dengan pengalaman apa pun, mereka mungkin, tetapi untuk pertama kali saya di papan, itu terlalu banyak, terlalu cepat. Pelajaran yang dipetik. Kedua, saya tidak memiliki pengalaman menyeimbangkan di papan apa pun. Saya tidak pernah berselancar, tidak pernah bermain skateboard, atau melakukan hal lain yang membutuhkan penyeimbangan di permukaan yang bergerak. Ombak yang diciptakan oleh angin mengguncang papan, dan itu membuat saya ketakutan. Ketiga, menambahkan daya ke layar menggeser pusat gravitasi di papan. Jika saya tidak menggeser tubuh saya dengan cara yang benar untuk mengimbangnya, kemungkinan besar saya akan jatuh. Jika saya tidak melepaskan layar dengan cukup cepat jika saya merasa saya di luar kendali, saya mungkin akan menyelam. Saya harus merasa nyaman berpindah cukup ke segala arah untuk tetap berada di papan, terutama selama hembusan angin. Keempat, saya tidak memiliki semua perlengkapan keselamatan yang tepat. Saya jelas membutuhkan helm untuk melindungi kepala saya dari tiang jika saya jatuh. Saya pernah melakukan panggilan dekat, dan saya beruntung. Saya tidak akan membuat kesalahan yang sama lagi: Saya segera memesan helm.

Belajar dari Masa Lalu

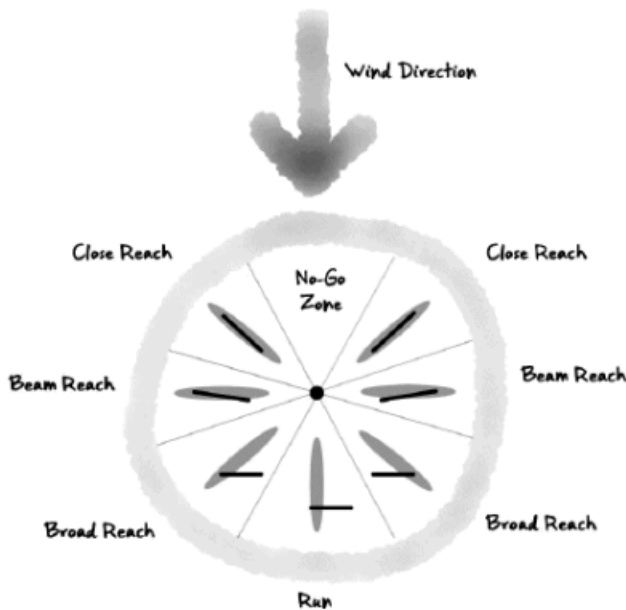
Ada beberapa hal yang dapat saya lakukan untuk membantu mencegah sesi latihan saya berikutnya menyerupai rst saya. Pertama, saya dapat memastikan sesi saya berikutnya berlangsung pada hari dengan angin yang lebih sedikit. Ini mungkin tidak begitu mengasyikkan, tetapi itu akan membuat saya berlatih tanpa kewalahan. Kedua, saya bisa merasakan keseimbangan di papan dengan mengeluarkannya tanpa layar terpasang, menggunakan dayung delapan puluh inci saya untuk bermanuver. Dengan melepas variabel layar, saya bisa merasakan bagaimana rasanya menyeimbangkan di papan, dan belajar seberapa jauh saya dapat bersandar di mana pun

arah tanpa terbalik. Ini tidak akan menjadi ujian yang sempurna, karena layar akan mengubah pusat gravitasi papan setelah terpasang, tetapi lebih baik daripada terus jatuh. Ketiga, saya dapat memfokuskan latihan saya berikutnya dengan layar untuk belajar merasakan keseimbangan layar. Jika saya memperhatikan, saya akan dapat merasakan ke arah mana layar bergerak, dan bagaimana hal itu mengarah ke papan. Saya dapat belajar kapan harus menggeser berat badan saya ke belakang untuk mengimbangi kekuatan layar, kapan harus melepaskan untuk menghindari jatuh ke belakang, dan kapan harus melepaskan jika hembusan angin tiba-tiba membuat layar terlalu berat untuk ditangani. Beberapa hari ke depan tenang, yang memberi saya kesempatan untuk mengisolasi keseimbangan di papan. Tes itu berhasil: pada akhir sesi latihan saya, saya tidak lagi panik, dan kemudian, hari berangin kencang memberi saya kesempatan untuk berlatih menyeimbangkan di papan saat air berombak. Saya masih harus mengatasi keanehan melihat ke bawah dan melihat air bergerak di bawah kaki saya, tetapi hanya mengalami sensasi untuk sementara waktu sangat menenangkan saraf saya. Beberapa hari kemudian, cerah dan hangat, dengan angin sepoi-sepoi yang kencang tetapi tidak terlalu gila. Saya memasang, meluncurkan, dan mengombak menjadi netral. Apakah saya akan mengulangi hari pertama? Tidak sama sekali: Saya hanya jatuh dua kali hari itu. Mengisolasi keseimbangan di papan sangat membantu, dan saya bisa menghindari jatuh ketika saya merasakan papan menjadi lepas kendali. Saya banyak melakukan perombakan hari itu, tetapi tidak apa-apa: itu lebih baik daripada minum air danau. Saya berlatih menambahkan daya ke layar, dan tiba-tiba saya bergerak.

Bagaimana Anda mengubah benda ini?

Membuat papan bergerak tanpa jatuh adalah kemenangan, tetapi itu memperkenalkan masalah baru yang mendesak: Bagaimana Anda berputar? Saya bereksperimen dengan hal-hal yang saya pelajari dari buku dan DVD: memutar papan di bawah kaki saya sambil memegang layar dalam keadaan netral

posisi sangat membantu. Ini juga membantu untuk mengingat bahwa papan layar (dan perahu layar pada umumnya) tidak dapat bergerak langsung ke arah angin. Ada zona "mati" atau "dilarang" yang memanjang empat puluh derajat ke kedua sisi angin: cobalah untuk berlayar di mana saja di dalam zona itu, dan Anda akan menemukan diri Anda "dalam besi", dan Anda akan berhenti, atau mulai bergerak mundur. Untuk bergerak ke arah angin, Anda harus mengarahkan papan setidaknya empat puluh derajat ke kedua sisi angin, lembaran di layar, lalu menempel untuk bergerak ke arah lain setelah Anda bergerak sebentar. Dengan bolak-balik, Anda dapat zig-zag menuju tujuan, meskipun itu di atas angin.



Penting juga untuk dapat mengubah papan di bawah daya. Saat Anda bergerak, menggeser layar ke kiri atau kanan mengubah pusat eort, titik fokus dari semua gaya yang dihasilkan angin pada layar, relatif terhadap pusat resistansi lateral papan, titik fokus resistansi antara papan dan

air. Dengan menggeser layar ke depan, belakang, pelabuhan, atau kanan, hubungan antara titik fokus ini berubah, dan papan berputar untuk mengimbangnya. Saat Anda kembali ke papan setelah jatuh, biasanya menempatkan diri Anda dalam posisi yang aneh. Idealnya, Anda ingin papan tegak lurus dengan angin, dengan layar mengarah ke arah angin. Dalam posisi ini, Anda dapat mengangkat layar tanpa terlalu banyak kesulitan. Namun, jika layar mengarah ke arah angin, Anda berada dalam bahaya boom memukul wajah Anda saat Anda mengangkat saat menangkap angin. Demikian juga, Anda dapat mengarah ke angin, langsung ke zona mati. Untuk mengimbangnya, yang terbaik adalah menggunakan kaki Anda untuk memutar layar sehingga tegak lurus dengan papan, terlepas dari arah angin. Setelah layar dan papan tegak lurus satu sama lain, Anda dapat naik perlahan, memungkinkan angin memutar papan ke posisi yang benar. Setelah layar berada di sisi angin, Anda dapat menghambak sepenuhnya, lalu menempel ke arah mana pun yang Anda inginkan.¹⁴ Setelah beberapa jam berlatih, saya menguasainya. Saya tidak terlalu gesit atau cepat, dan giliran saya tidak cantik, tetapi saya dapat meluncurkan papan, berlayar ke tengah danau, dan kembali ke tempat saya memulai. Dibandingkan dengan pengalaman windsurfing pertama saya, ini adalah peningkatan yang sangat besar.

Untuk segalanya, ada musim

Musim gugur melanda dengan cepat dan keras di Front Range of Colorado, biasanya pada akhir September. Suatu hari tujuh puluh derajat dan cerah, berikutnya, tiga puluh derajat dan mengancam salju. Saya telah berada di air sebanyak mungkin setiap sore, mencoba memaksimalkan waktu latihan saya sebelum cuaca berubah. Suhu air stabil pada enam puluh delapan derajat: agak dingin, tetapi tidak di zona bahaya hipotermia. Dengan pakaian basah, awal

Kejutan memasuki air adalah bagian terburuk: setelah itu, jatuh bukanlah masalah besar. Pada hari-hari ketika angin lebih dari enam mil per jam, saya memasang papan layar. Pada hari-hari ketika angin tenang, saya malah mendayung papan dayung. Saya telah melakukan lebih banyak paddleboarding daripada windsurfing: sangat sulit untuk berlayar saat tidak ada angin. Ini membuat frustrasi, karena saya ingin mendapatkan latihan angin sebanyak yang saya bisa, tetapi saya tidak bisa bernegosiasi dengan Ibu Pertiwi. Angin bertiup cukup kencang ketika saya bisa keluar di air, atau tidak. Suatu pagi, suhu anjlok menjadi empat puluh derajat. Selama tiga hari, suhu air danau turun ke derajat dingin. Dengan pakaian basah 5/4 saya, saya bisa bertahan hidup untuk sementara waktu, tetapi berisiko hipotermia atau tenggelam tidak termasuk dalam daftar prioritas saya. Bagi saya, musim angin angin telah berakhir. Saya menghitung total jam latihan saya, dan tidak mencapai tujuan saya: sembilan jam latihan secara total, jauh lebih sedikit dari dua puluh yang ingin saya habiskan pada saat ini. Saya menghabiskan lebih banyak waktu dari itu di waterpaddleboarding. Ada pelajaran penting di sini: lingkungan itu penting. Sangat mudah untuk kecewa karena saya tidak mencapai target latihan saya, tetapi keinginan saya untuk belajar tidak bisa membuat angin bertiup. Meski begitu, tidak bijaksana untuk menghabiskan terlalu banyak waktu di atas air, karena kelelahan adalah faktor risiko utama. Akibatnya, setiap hari saya dapat berlatih sama sekali dengan total tiga puluh hingga empat puluh menit di atas air. Meskipun saya kurang dari dua puluh jam, saya belajar banyak hal. Saya memiliki rig yang solid, belajar cara merakitnya dengan benar, mengungkapkan cara meluncur dalam berbagai kondisi angin, dan berlatih kembali ke papan dari air lebih banyak daripada yang dapat saya hitung. Saya dapat mengangkat layar, mengarahkan diri ke arah umum yang benar, dan mengendalikan papan di bawah kekuatan tanpa terbalik. Saya dapat memutar papan saat saya membutuhkannya, dan kembali ke pantai dengan selamat, di area pantai yang kira-kira sama dengan tempat saya meluncur. Itu banyak kemajuan dalam waktu singkat.

Sebagai bonus, saya mengambil keterampilan mini: paddleboarding. Pengaturan dan teknik tidak serumit atau menuntut, tetapi sangat menyenangkan. Bahkan ketika angin tidak bertiup, saya sangat menikmati meluncur di atas air, mengejar bebek dari satu ujung danau ke ujung lainnya. Sangat menyenangkan untuk mendayung saat matahari terbenam, dan antara latihan dan pemandangan, ini adalah cara yang bagus untuk bersantai.

Meninjau Metode

Mari kita tinjau inti dari metode yang saya gunakan untuk mempelajari cara selancar angin:

Saya memperoleh peralatan yang diperlukan: papan, layar, pakaian basah, perangkat otosi pribadi, helm, dan perlengkapan keselamatan penting lainnya. Saya belajar cara memasang papan dan layar, serta membongkar, memelihara, mengangkut, dan menyimpan rig dengan aman. Saya belajar bagaimana melindungi diri dari risiko besar, termasuk tenggelam, gegar otak, dan hipotermia. Saya belajar cara meluncurkan papan dari pantai, menaikkan layar ke posisi netral, dan menambahkan tenaga ke layar untuk mulai bergerak. Saya belajar cara memutar papan (tack/jibe), dan cara memosisikan layar untuk bergerak dalam berbagai kondisi angin. Saya belajar bagaimana menghindari jatuh dari papan, dan bagaimana kembali ke papan dan mengangkat layar jika saya jatuh.

Ke Mana Saya Pergi Dari Sini

Pada saat Anda membaca ini, itu akan sekali lagi menjadi musim angin. Saya akan keluar segera setelah cuaca dan kondisi air keluar dari zona bahaya hipotermia, memperoleh kembali dasar-dasar tali, keseimbangan, dan belok. Saya akan memperoleh kembali keterampilan dasar ini dengan cepat, jadi saya mempersiapkan tantangan berikutnya: membuat papan ke hydroplane, yang meningkatkan kecepatan secara dramatis. "Planing" hanya dimungkinkan dalam kondisi angin kencang, dan peningkatan kecepatan meningkatkan risiko menabrak dan cedera, jadi saya harus benar-benar nyaman dengan dasar-dasarnya sebelum mencobanya.

Saya juga akan menggunakan layar yang lebih besar. Kondisi angin yang umumnya rendah (dan bervariasi) di danau membuat menggunakan layar yang lebih besar merupakan ide yang bagus, jadi setelah saya merasa nyaman menggunakan layar seluas 4,7 meter persegi, saya akan mulai berlatih dengan 7,5. Di antara dua layar, saya harus menutupi: Saya akan menggunakan layar besar untuk hari-hari dengan angin lebih rendah, dan layar yang lebih kecil untuk hari-hari berangin kencang. Secara keseluruhan, windsurung sangat menyenangkan. Saya menikmati waktu saya di atas air, terutama sekarang karena saya tidak terlalu jatuh. Saya menantikan musim berikutnya, dan memanfaatkan hari-hari berangin kapan pun mereka muncul. Saya tidak dapat mengendalikan Ibu Pertiwi, tetapi saya dapat mengontrol bagaimana saya berlatih ketika kondisinya menguntungkan. Itu sudah cukup.

Penutup

Pencapaian tampaknya terkait dengan tindakan. Pria dan wanita sukses terus bergerak. Mereka membuat kesalahan, tetapi mereka tidak berhenti.—CONRAD HILTON, PENDIRI JARINGAN HOTEL HILTON

• • •

Dalam waktu kurang dari setahun, saya mempelajari enam keterampilan kompleks. Saya bukan jenius atau orang aneh di alam. Saya tidak berbakat secara alami. Saya tidak berhenti dari pekerjaan harian saya. Saya tidak meninggalkan segalanya dan pindah ke belahan dunia lain. Saya tidak mengabaikan keluarga saya. Saya hanya menyisihkan satu jam atau lebih setiap hari untuk berlatih, dan saya berlatih dengan cara yang cerdas. Keterampilan yang dimulai sebagai misteri lengkap menjadi dapat dipahami dalam hitungan hari, seringkali berjam-jam. Yang dibutuhkan hanyalah sedikit penelitian dan sekitar dua puluh jam latihan yang konsisten, fokus, dan disengaja. Lebih baik lagi, latihan saya menjadi rutinitas: keterampilan ini sekarang terpisah dari kehidupan sehari-hari saya. Mempelajari subskill terpenting rst membuatnya sangat mudah untuk terus maju. Pada saat Anda membaca ini, saya akan menjadi lebih baik di masing-masing bidang ini. Seberapa baik akan tergantung pada seberapa banyak saya berlatih. Jika Anda ingin memperoleh keterampilan baru, Anda harus berlatih. Tidak ada cara lain. Anda bisa mempersiapkan. Anda dapat meneliti. Anda dapat menghilangkan gangguan dan mengubah lingkungan Anda agar lebih mudah berlatih. Anda dapat menemukan cara cerdas untuk membuat praktik Anda lebih efektif. Tapi, pada akhirnya, Anda harus berlatih.

Apa yang terasa seperti jalan panjang adalah jalan terpendek. Pintasan tanpa latihan tidak ada. Tidak ada latihan, tidak ada perolehan keterampilan. Sesederhana itu. Mengapa kita tidak berlatih? Sederhana: kami sibuk dan kami takut. Shakespeare mengatakannya dengan baik sejak lama, dalam sebuah drama berjudul Measure for Measure: "Keraguan kita adalah pengkhianat, dan membuat kita kehilangan kebaikan yang sering kita menangkan, dengan takut untuk mencoba." Penghalang utama untuk perolehan keterampilan yang cepat bukanlah fisik atau intelektual: itu emosional. Melakukan sesuatu yang baru selalu tidak nyaman, dan mudah membuang banyak waktu dan energi untuk berpikir tentang berlatih alih-alih berlatih. Untungnya, penghalang frustrasi sangat mudah untuk ditembus: perolehan keterampilan selalu terasa lebih besar dari yang sebenarnya. Dengan menciptakan waktu untuk latihan, melakukan sedikit penelitian awal, dan bersandar pada ketidaknyamanan awal, Anda akan selalu melihat kemajuan besar dalam sepuluh hingga dua puluh jam latihan. Yang diperlukan untuk menuai hasilnya adalah ledakan kecil keberanian, ketekunan, dan sedikit ketabahan. Anda tidak perlu memilih banyak keterampilan untuk diperoleh: pilih saja satu. Ambil keterampilan di daftar "ingin dilakukan" Anda dan berkomitmen untuk mencobanya. Pelajari bahasa itu, mainkan instrumen itu, jelajahi permainan itu, kerjakan proyek itu, masak hidangan itu, ciptakan seni itu. Ini lebih mudah daripada yang dirasakan. Berkomitmen untuk melatih keterampilan itu selama satu jam atau lebih sehari untuk bulan depan. Setelah Anda benar-benar mulai berlatih, Anda akan selalu mengambilnya lebih cepat dari yang Anda harapkan. Pecahkan, luangkan waktu, coba hal-hal baru, dan otak Anda akan mulai mengambil teknik secara otomatis: itulah yang dilakukan otak. Ketika Anda terjebak atau bingung, uji pendekatan baru. Ingat: begitu Anda memulai, Anda tidak dapat berhenti sampai Anda mencapai tingkat kinerja target atau tanda dua puluh jam. Berjuang jika Anda harus, tetapi jangan berhenti. Tunjukkan ketabahan Anda, dan terus maju. Anda akan sampai di sana: yang diperlukan hanyalah latihan. Satu pemikiran: satu-satunya waktu yang dapat Anda pilih untuk berlatih adalah hari ini.

Tidak besok. Tidak minggu depan. Tidak bulan depan atau tahun depan. Hari ini. Ketika Anda bangun di pagi hari, Anda memiliki pilihan. Anda dapat memilih untuk menginvestasikan waktu Anda untuk memperoleh keterampilan yang akan membuat hidup Anda lebih sukses, menyenangkan, dan bermanfaat ... Atau Anda dapat membuang-buang waktu Anda untuk melakukan hal lain. Apa yang akan Anda lakukan hari ini?

Catatan

• • •

BAB 1

1. Ericsson, K. Anders, Neil Charness, Paul J. Feltovich, dan Robert R. Homan, eds. Buku Pegangan Keahlian dan Kinerja Ahli Cambridge (Cambridge: Cambridge University Press, 2006).
2. <http://www.chirunning.com/>.
3. Program perguruan tinggi sarjana biasanya memakan waktu empat tahun karena konvensi dan kepentingan pribadi: perguruan tinggi menghasilkan lebih banyak pendapatan selamanya tahun siswa terdaftar. Itu tidak berarti tidak mungkin untuk menyelesaikan program yang paling menuntut bahkan dalam waktu yang lebih singkat: ScottH. Young menyelesaikan kurikulum ilmu komputer sarjana MIT dalam waktu kurang dari satu tahun. See <http://www.scotthyoung.com/blog/mit-challenge/>.
4. VanLehn, Kurt. "Akuisisi Keterampilan Kognitif." Tinjauan Tahunan Psikologi 47, no. 1 (1996): 513–539.

BAB 2

1. Jika Anda dapat berhubungan dengan frustrasi saya, saya sarankan membaca *The Renaissance Soul: Life Design for People with Too Many Passions to Pick Just One* oleh Margaret Lobenstine (New York: Harmony, 2006). Membaca buku ini membantu saya menyadari bahwa keragaman minat saya adalah

kekuatan, dan bahwa adalah mungkin untuk menyusun eorts saya untuk memanfaatkan kecenderungan alami saya untuk mempelajari banyak hal sekaligus.

2. http://www.newyorker.com/reporting/2011/10/03/111003fa_fact_gawande.

3. Saya menggunakan timer interval yang dapat diprogram Enso Pearl: <http://www.salubrion.com/products/ensopearl/>. Anda dapat menggunakan pengatur waktu apa pun yang Anda suka, termasuk pengatur waktu dapur dasar, tetapi Enso memiliki interval bawaan dan lonceng yang terdengar sangat bagus, yang lebih penting daripada yang terlihat. Saat Anda banyak berlatih, bunyi bip elektronik cepat menjadi tua.

4. Snoddy, George S. "Pembelajaran dan Stabilitas: Analisis Psikofisiologis APdari Kasus Pembelajaran Motorik dengan Aplikasi Klinis." *Jurnal Psikologi Terapan* 10, no. 1 (1926):1.

5. Newell, Allen, dan Paul S. Rosenbloom. "Mekanisme Akuisisi Keterampilan dan Hukum Praktek." *Keterampilan Kognitif dan Akuisisi Mereka* (1981): 1–55.

6. Logan, Gordon D. "Menuju Teori Contoh Otomatisasi." *Tinjauan Psikologis* 95, no. 4 (1988): 492.

BAB 3

1. <http://ankisrs.net/>.

2. <http://www.supermemo.com/>.

3. <http://smartr.be/>.

4. Ini bukan buku tentang teknik belajar akademis, tetapi karena Anda rajin membaca catatan akhir, inilah metode belajar sederhana yang menipu yang bekerja dengan sangat baik: pilih ide, keluarkan yang kosong

selembar kertas, lalu cobalah untuk menjelaskan ide itu sepenuhnya menggunakan bahasa yang sangat sederhana, seolah-olah Anda sedang mengajar seorang pemula. Kesenjangan dalam pengetahuan Anda akan menjadi jelas dengan sangat cepat, yang membuatnya mudah untuk kembali ke materi sumber untuk mempelajari apa yang hilang. Scott H. Young, seorang peneliti keterampilan belajar, menyebut ini "Teknik Feynman" (untuk menghormati fisikawan terkenal Richard Feynman), dan itu cukup efektif. Untuk lebih jelasnya, see <http://www.scotthyoung.com/learnonsteroids/grab/TranscriptFeynman.pdf>

BAB 4

1. Ada sesuatu yang sangat, sangat menyenangkan tentang berlari secepat mungkin menuju penghalang setinggi pinggang dan melompatnya dengan langkah yang meluncur. Yang paling dekat dengan perasaan seperti Superman adalah saat kaki saya terhubung langsung dengan pusat rintangan selama balapan. Batang kayu rintangan itu hancur, dan aku berlayar tanpa kehilangan kecepatan. Itu luar biasa.
 2. http://www.youtube.com/watch?v=IMC1_RH_b3k.
 3. <http://www.nytimes.com/2009/07/26/magazine/26FOB-consumed-t.html>.
 4. <http://www.gilhedley.com/>.
5. Anda dapat menonton "pidato fuzz" Gil yang terkenal di sini: <http://www.youtube.com/watch?v=FtSP-tkSug>. Perhatikan bahwa video berisi rekaman tubuh manusia yang mati, jadi Anda mungkin ingin melewatkannya jika Anda mual.
6. Sarno, John E. Menyembuhkan Sakit Punggung: Koneksi Pikiran-Tubuh (New York: Grand Central Life & Style, 2010).
7. Saya akan mencoba menjelaskan ribuan tahun sejarah yang sangat kompleks dalam beberapa paragraf. Buku-buku tentang topik ini dapat

semua perpustakaan, jadi mohon maaf singkat saya. Jika Anda tertarik dengan ikhtisar yang lebih rinci, saya sarankan membaca *The Great Transformation: The Beginning of Our Religious Traditions* oleh Karen Armstrong (New York: Anchor, 2007).

8. Armstrong, Karen. *Transformasi Besar: Awal dari Tradisi Keagamaan Kita* (New York: Anchor, 2007).

9. Desikachar, T. V. K., dan R. H. Cravens. *Heath, Healing, and Beyond: Yoga and the Living Tradition of Krishnamacharya* (New York: North Point Press, 1998). Biografi singkat juga tersedia at <http://www.yogajournal.com/wisdom/465>.

10. Seberapa banyak Krishnamacharya secara sadar mengadopsi dari sumber-sumber non-hatha seperti senam Inggris adalah masalah perdebatan. Minimal, banyak pose dan gerakan Krishnamacharya termasuk dalam urutannya memiliki kemiripan yang sangat kuat dengan senam dan latihan militer pada saat itu.

11. http://www.nytimes.com/2012/01/08/magazine/how-yoga-can-wreck-your-body.html?_r=3&pagewanted=all.

12. <http://www.manduka.com/us/shop/categories/products/gear/manduka-pro-black-sage/>.

13. <http://orthoinfo.aaos.org/topic.cfm?topic=A00063>.

BAB 5

1. <http://personalmba.com/best-business-books/>.

2. <http://wordpress.org>.

3. Bagi yang penasaran: tumpukan WordPress standar saya terdiri dari PHP5 dengan PHP-FastCGI, NGINX, APC, MSMTMP, dan WP-Supercache

pada Slicehost.com VPS yang menjalankan Ubuntu 8.04 LTS, semuanya dengan kustomconfiguration les.

4. <http://jekyllrb.com>.

5. <http://github.com>.

6. Untuk beberapa alasan, hampir setiap tutorial pemrograman dimulai dengan menunjukkan kepada Anda cara menampilkan atau mencetak "Halo, Dunia!"

7. <http://stackoverflow.com>.

8. <http://news.ycombinator.com>.

9. <http://rubyonrails.org/>.

10. <http://www.sinatrarb.com>.

11. <http://37signals.com/>.

12. <http://rubysource.com/rails-or-sinatra-the-best-of-both-worlds/>.

13. <http://paulstamatiou.com/how-to-wordpress-to-jekyll>.

14. <https://github.com/sstephenson/rbenv>.

15. Dalam praktiknya, "peretasan" tidak seperti bagaimana itu digambarkan di film, yang sangat mengecewakan.

16. <https://toolbelt.heroku.com>.

17. <http://www.heroku.com/>.

18. <http://git-scm.com/>.

19. Versi Ruby sebelum 1.9.3-p125 memerlukan program yang disebut GCC untuk menyelesaikan instalasi. GCC tersedia at <https://github.com/kennethreitz/osx-gcc-installer>.

20. Pemrogram menciptakan akronim "RTFM", yang merupakan singkatan dari "baca manual (aneh)," sebagai tanggapan standar atas pertanyaan tentang masalah yang tercakup dalam dokumentasi resmi program.

21. <http://www.ruby-lang.org/en/documentation/>.

22. <http://0xfe.muthanna.com/rubyrefresher/>.

23. <https://code.google.com/p/ruby-security/wiki/Guide>.

24. Istilah untuk modifikasi lanjutan objek, kelas, dan metode inti Ruby disebut metaprogramming. Saya mengambil sebuah buku berjudul Metaprogramming Ruby: Program Like the Ruby Pros oleh Paolo Perrotta (Raleigh, NC: Pragmatic Bookshelf, 2010), dan itu jauh di kepala saya saat ini. Hal pertama rst.

25. <http://www.ruby-doc.org/core-1.9.3/index.html>.

26. <http://ruby.learncodethehardway.org/>.

27. <https://devcenter.heroku.com/articles/keys>.

28. <https://devcenter.heroku.com/articles/ruby>.

29. <https://devcenter.heroku.com/articles/rack>.

30. <http://macromates.com/>.

31. <https://devcenter.heroku.com/articles/bundler>.

32. <http://www.sinatrarb.com/intro>.

33. <http://backpackit.com>.

34. <http://tom.preston-werner.com/2010/08/23/readme-driven-development.html>.

35. <http://www.postgresql.org/>.

36. <http://datamapper.org/>.

37. <http://stackoverflow.com/questions/1152299/what-is-an-object-relational-mapping-framework>.
38. <http://en.wikipedia.org/wiki/SQL>.
39. <http://www.sqlite.org/>.
40. <http://pow.cx/>.
41. <https://github.com/rodreegez/powder>.
42. <http://twitter.github.com/bootstrap/>.
43. Saya tidak tahu mengapa disebut siput, dan saya setuju itu aneh.
44. Lihat <http://www.regular-expressions.info/> untuk contoh ekspresi reguler umum.
45. <http://daringreball.net/projects/markdown/>.
46. <http://www.httpwatch.com/httpgallery/authentication/>.
47. Menggunakan SSL pada domain khusus lebih rumit: Anda harus melalui proses panjang untuk memverifikasi identitas Anda dan mendapatkan "sertifikat" yang mengamankan setiap sesi pengguna.
48. <https://github.com/SFEley/sinatra-ash>.

BAB 6

1. Inilah fakta menyenangkan: Tiger Woods adalah satu-satunya pegolf profesional dalam sejarah yang memenangkan US Masters dengan tiga ayunan golf yang berbeda. Tiger berlatih ulang dengan Butch Harmon setelah kemenangan Masters-nya pada tahun 1997, menang lagi pada tahun 2001 dan 2002, berlatih ulang pada tahun 2002 dengan Hank Haney, kemudian menang pada tahun 2005. Baru-baru ini, Tiger memulai swingretraining ketiganya dengan Sean Foley pada tahun 2011, menjadikan ayunan keempat Tiger ini sejak menjadi pegolf profesional.

2. Gambar keyboard vektor via <http://wowvectors.com/object/mac-keyboard-vector/>. Didistribusikan di bawah lisensi Creative Commons Attribution 3.0 Unported.
3. Untuk sejarah yang sangat menarik tentang pengembangan mesin tik QWERTY, lihat "The Fable of the Keys" oleh S. J. Liebowitz dan Stephen E. Margolis, tersedia at <http://www.utdallas.edu/~liebowit/keys1.html>.
4. <http://www.google.com/patents?id=qSVdAAAAEBAJ>.
5. <http://mkweb.bcgsc.ca/carpalx/>.
6. <http://mkweb.bcgsc.ca/carpalx/?colemak>.
7. <http://colemak.com>.
8. Sistem operasi lain mungkin perlu menginstal paket perangkat lunak kecil yang memungkinkan tata letak. Paket untuk sistem paling populer tersedia di <http://colemak.com>.
9. <http://www.typematrix.com/2030/features.php>.
10. Pada keyboard TypeMatrix, Anda dapat mengaktifkan mode perangkat keras Colemak dengan menekan Fn+F5.
11. Ini tidak akan berfungsi di semua keyboard: banyak model memiliki tombol berukuran berbeda di baris atas, tengah, dan bawah. Jika itu masalahnya, mungkin lebih mudah untuk mendapatkan keyboard lain atau memesan stiker yang dapat ditempelkan di setiap tombol.
12. <http://www.typeonline.co.uk/typingspeed.php>.
13. Materi korpus uji berasal dari buku-buku yang berada dalam domain publik, yang didistribusikan secara gratis secara online melalui Project Gutenberg. Lihat <http://www.gutenberg.org/>.
14. <http://www.mavisbeacon.com/>.

15. <http://typingtrainer.sourceforge.net/>.
16. <http://github.com/wwwtyro/keyzen>.
17. Luft, Andreas R., dan Manuel M. Buitrago. "Tahapan Pembelajaran MotorSkill." *Neurobiologi Molekuler* 32, no. 3 (2005): 205–216.
18. Walker, Matthew P., dan Robert Stickgold. "Ini adalah latihan, withSleep, yang membuat sempurna: implikasi pembelajaran yang bergantung pada tidur dan plastisitas untuk kinerja keterampilan." *Klinik dalam Kedokteran Olahraga* 24, no. 2 (2005): 301–317.
19. <http://www.daskeyboard.com/model-s-ultimate-silent/>.
20. <http://type-fu.com>.
21. Milton, Yakobus. *Mengukur Akuisisi Kosakata Bahasa Kedua* (Bristol, Inggris: Multilingual Matters, 2009).
22. <http://code.google.com/p/amphetype/>.
23. <http://norvig.com/ngrams/>.

BAB 7

1. Menurut pendapat saya, aspek yang paling mengecewakan dari kenyataan adalah kurangnya sihir: mengingat seberapa banyak yang saya baca, saya setidaknya akan menjadi penyihir level 80 sekarang. Sayangnya, menggunakan pikiran saya untuk memanipulasi struktur alam semesta masih di luar kemampuan saya.
2. Inilah fakta menyenangkan lainnya: jika Anda mewarnai setumpuk kartu secara menyeluruh, urutan kartu yang dihasilkan belum pernah terlihat sebelumnya dalam sejarah alam semesta. "52 faktorial" adalah angka yang sangat besar: 8,065 kali 1067, atau lebih dari 80 unvigintillion, kemungkinan kombinasi.
3. Game terkenal lainnya termasuk Game Muntah Darah, di mana salah satu pemain master meninggal setelah pertandingan, dan

Permainan Bom Atom, di mana pertandingan terganggu olehledakan bom atom di atas Hiroshima. Setelah bom berbunyi, para pemain beristirahat untuk makan siang, mengganti batu di papan, dan melanjutkan permainan. Lihat <http://senseis.xmp.net/?FamousGoGames> untuk game yang lebih terkenal.

4. <http://www.ymimports.com>.

5. <http://senseis.xmp.net/>.

6. Ya, dari situlah nama konsol video game Atari berasal.

7. <http://senseis.xmp.net/?TheTenGoldenRulesList>.

8. http://diiq.org/ve_stone_questions.html.

BAB 8

1. <http://cdp.sagepub.com/content/14/6/317.short>.

2. <http://www.crowhillguitars.com>.

3. <http://www.daddario.com/DADProductDetail.Page?ActiveID=3769&productid=264>.

4. <http://www.axisofawesome.net>.

5. <http://www.ukuleles.com/Technology/strings.html>.

6. <http://www.kiwiukulele.co.uz/Kiwi-Ukulele-Chord-Chart.pdf>.

BAB 9

1. <http://jimbodouglass.blogspot.com/>.

2. Jika Anda ingin melihat seperti apa rasanya melaju sangat cepat di papan Formula, lihat video boom-cam Jim:

<http://jimbodouglass.blogspot.com/2010/01/formula-windsurng-boom-mount-video.html>.

3. Pramuka menyebutnya sistem teman, dan itu adalah aturan utama keamanan air.

4. Pakaian basah menjebak lapisan air yang dekat dengan tubuh, yang kemudian dihangatkan oleh panas tubuh, bertindak sebagai lapisan isolasi. Di bawah suhu air fty derajat, yang terbaik adalah menggunakan pakaian kering, yang mencegah air menyentuh kulit.

5. <http://www.dbw.ca.gov/Pubs/Windsurf/index.htm>.

6. <http://jimbodouglass.blogspot.com/2008/02/top-16-pertanyaan-terjawab.html>.

7. <http://jimbodouglass.blogspot.com/2010/11/updated-windsurf-calculator-online.html>.

8. <http://www.isthmussailboards.com/>.

9. <http://www.wunderground.com/>.

10. <http://www.windnder.com/>.

11. <http://www.iwindsurf.com/>.

12. <http://www.kestrelmeters.com/products/kestrel-3000-wind-meter>.

13. http://www.isthmussailboards.com/info_technical_help.asp.

14. Untuk panduan terperinci tentang cara melakukan ini, lengkap dengandiagram, see<http://jimbodouglass.blogspot.com/2012/10/beginner-windsurng-how-to-reorient.html>.

Ucapan terima kasih

• • •

Kepada Kelsey dan Lela: Aku mencintaimu. Terima kasih untuk semuanya. Kepada Dave, Sheri, dan Zulema: terima kasih atas pertengkaran Lela yang memungkinkan buku ini. Untuk Leslie Kamino, Derek Sivers, Jim Douglass, dan Gary Stone: terima kasih atas kemurahan hati dan bantuan Anda. Saya telah belajar banyak dari Anda. Kepada Lauren Baker: terima kasih telah membuat ilustrasi dalam cetakan buku ini layak. Kepada Lisa DiMona: Saya sangat beruntung mendapatkan bantuan dan dukungan Anda. Kepada Adrian Zackheim dan Joel Rickett: viva Portofolio! Untuk Emily Angell: Anda memegang pena merah dengan nesse. Terima kasih atas wawasan dan ketekunan Anda. Bagi Will Weisser, Margot Stamas, Richard Lennon, dan AllisonMcLean: buku tanpa pembaca tidak terlalu menyenangkan. Terima kasih atas semua yang Anda lakukan untuk menyebarkan berita. Kepada Joe Perez dan Dan Donohue: ketika orang menilai buku dari sampul mereka, Anda selalu membuat penulis Anda terlihat hebat. Saya menghargai kerja keras Anda untuk mewujudkan ide ini. Kepada Bria Sandford, Samantha LaBue, Sarah Katie Coe, ThomasDussel, dan seluruh tim produksi dan distribusi di Penguin: terima kasih atas semua yang telah Anda lakukan, dan terus lakukan, untuk membawa buku ini kepada pembaca di seluruh dunia.

Kepada Anda, Pembaca yang budiman: Anda adalah alasan saya melakukan apa yang saya lakukan. Saya harap Anda menikmati buku ini, dan informasi yang berguna di tahun-tahun mendatang. Semoga sukses!



AWAL

Biarkan percakapan dimulai...

Ikuti Penguin Twitter.com@penguinukbooksIkuti semua cerita kami YouTube.com/penguinbooks

Sematkan 'Penguin Books' ke Pinterest AndaLike

'Penguin Books' di Facebook.com/penguinbooks

Cari tahu lebih lanjut tentang penulis dan temukan lebih banyak cerita seperti ini di Penguin.co.uk

PENGUIN PORTOFOLIO

Diterbitkan oleh Penguin Group Penguin Books Ltd, 80 Strand, London WC2R 0RL, Inggris Penguin Group (USA) Inc., 375 Hudson Street, New York, New York 10014, USA Penguin Group (Kanada), 90 Eglinton Avenue East, Suite 700, Toronto, Ontario, Canada M4P 2Y3 (divisi dari Pearson Penguin Canada Inc.) Penguin Ireland, 25 St Stephen's Green, Dublin 2, Irlandia (divisi dari Penguin Books Ltd) Penguin Group (Australia), 707 Collins Street, Melbourne, Victoria 3008, Australia (divisi dari Pearson Australia Group Pty Ltd) Penguin Books India Pvt Ltd, 11 Community Centre, Panchsheel Park, New Delhi – 110017, India Penguin Group (NZ), 67 Apollo Drive, Rosedale, Auckland 0632, Selandia Baru (divisi dari Pearson New Zealand Ltd) Penguin Books (Afrika Selatan) (Pty) Ltd, Blok D, Rosebank Océ Park, 181 Jan Smuts Avenue, Parktown North, Gauteng 2193, Afrika Selatan Penguin Books Ltd, Terdaftar Oces: 80 Strand, London WC2R 0RL, Inggris www.penguin.com Pertama kali diterbitkan 2013 Hak Cipta © Worldly Wisdom Ventures LLC, 2013 Hak moral pemegang hak cipta telah ditegaskan Desain dan ilustrasi sampul: Dan Donohue Semua hak dilindungi undang-undang Foto oleh penulis ISBN: 978-0-670-92193-5