

Primeiros Elogios ao *Use a Cabeça! Python*

“O *Use a Cabeça! Python* é uma ótima introdução sobre como a linguagem Python é usada no mundo real. O livro vai além da sintaxe, para ensiná-lo a criar aplicativos para telefones Android, Google App Engine e muito mais.”

— **David Griffiths, autor e instrutor da Agile**

“Onde outros livros começam com a teoria e passam para os exemplos, o *Use a Cabeça! Python* salta diretamente para o código e explica a teoria à medida em que você lê. Este é um ambiente muito mais eficaz de aprendizagem porque envolve o leitor desde o início. Foi uma alegria lê-lo. Foi divertido, sem ser petulante e informativo sem ser condescendente. A amplitude de exemplos e explicações abrange a maior parte do que você faz em seu trabalho diário. Eu recomendo este livro a qualquer um que esteja começando no Python.”

— **Jeremy Jones, coautor do *Python for Unix and Linux System Administration***

“O *Use a Cabeça Python* é um livro ótimo para começar a se aprimorar em uma linguagem que é cada vez mais relevante.”

— **Phil Hartley, Universidade de Avanço Tecnológico**

Elogios sobre outros livros da série

“O *Use a Cabeça! Java*, de Kathy e Bert transforma a página impressa na coisa mais próxima à interface gráfica do usuário que você já viu. Em tom irônico, de maneira descolada, os autores tornam o aprendizado do Java uma experiência envolvente do tipo ‘o que você fará em seguida?’”

— **Warren Keuffel**, *Software Development Magazine*

“Além do estilo envolvente que o leva de leigo para o status de guerreiro Java, o *Use a Cabeça! Java* abrange uma enorme quantidade de questões práticas que os outros livros abandonam, como o temido ‘exercício para o leitor...’. É inteligente, irônico, descolado e prático – não há muitos livros que podem fazer essa reivindicação e viver de acordo com ela, enquanto também ensinam sobre a serialização do objeto e os protocolos de inicialização da rede.”

— **Dr. Dan Russell**, Diretor de Ciências e Pesquisa da Experiência do Usuário
Centro de Pesquisa Almaden da IBM (e ensina Inteligência Artificial na
Universidade de Stanford)

“É rápido, irreverente, divertido e cativante. Tenha cuidado – você pode realmente aprender algo!”

— **Ken Arnold**, ex-engenheiro sênior da Sun Microsystems e coautor
(com James Gosling, criador do Java), *Linguagem de Programação Java*

“Eu sinto como se o peso de milhares de livros acabasse de ser retirado da minha cabeça.”

— **Ward Cunningham**, inventor do Wiki e fundador do grupo Hillside

“O tom certo para o guru entendido, casual e legal em todos nós. A referência certa para as estratégias de desenvolvimento prático – faz meu cérebro avançar sem ter que passar pela fala velha e cansada do professor.”

— **Travis Kalanick**, fundador da Scour and Red Swoosh
Membro do MIT TR100

“Há livros que você compra, livros que você guarda, livros que você mantém em sua mesa e, graças à O'Reilly e à equipe do *Use a Cabeça!*, existem livros *Use a Cabeça!*. Eles são os únicos que estão com as orelhas dobradas, deformados e são levados a todos os lugares. O *Use a Cabeça! SQL* está no topo da minha pilha. Ah, até o PDF que tenho para revisão está surrado e rasgado.”

— **Bill Sawyer**, Gerente de Currículos ATG, Oracle

“A admirável clareza, humor e doses substanciais de inteligência fazem deste o tipo de livro que ajuda até os não programadores a pensar bem sobre a resolução do problema.”

— **Cory Doctorow**, coeditor do Boing Boing autor,
Down and Out in the Magic Kingdom e
Someone Comes to Town, Someone Leaves Town

Elogios sobre outros livros da série

“Eu recebi o livro ontem e comecei a lê-lo... e não consegui parar. Isto é definitivamente *très* ‘legal’. É divertido e eles abordam muita coisa. Estão certos ao meu ver. Estou realmente impressionado.”

— **Erich Gamma, Engenheiro da IBM e coautor de *Design Patterns***

“Um dos livros mais divertidos e inteligentes sobre design de software que eu já li.”

— **Aaron LaBerge, Tecnologia VP, ESPN.com**

“O que costumava ser um processo longo de aprendizagem com tentativas e erros, agora foi reduzido a um livro envolvente.”

— **Mike Davidson, CEO, Newsvine, Inc.**

“O design elegante é o destaque de cada capítulo, cada conceito transmitido em doses iguais de pragmatismo e inteligência.”

— **Ken Goldstein, Vice-Presidente executivo da Disney Online**

“Eu ♥ o *Use a Cabeça HTML com CSS & XHTML* – ensina tudo que você precisa aprender em um formato divertido.”

— **Sally Applin, Designer IU e Artista**

“Geralmente, quando leio um livro ou um artigo sobre padrões de projetos, tenho que ficar de olho em alguma coisa só para me certificar que estava prestando atenção. Não é o caso. Por mais estranho que possa parecer, este livro torna divertido o aprendizado sobre padrões de projetos.”

“Enquanto outros livros sobre padrões de projetos estão dizendo ‘Bueller... Bueller... Bueller...’, este está sobre o carro alegórico, cantando ‘*Shake it up, baby!*’”

— **Eric Wuehler**

“Eu literalmente amo este livro. Na verdade, eu beijei este livro na frente da minha esposa.”

— **Satish Kumar**

Outros livros da série *Use a Cabeça!*

Use a Cabeça! Ajax Profissional

Use a Cabeça! Álgebra

Use a Cabeça! Análise & Projeto Orientado ao Objeto

Use a Cabeça! Análise de Dados

Use a Cabeça! C# - 2ª Edição

Use a Cabeça! Desenvolvimento de Software

Use a Cabeça! Desenvolvendo para iPhone

Use a Cabeça! Estatística

Use a Cabeça! Excel

Use a Cabeça! Física

Use a Cabeça! Geometria 2D

Use a Cabeça! Java 2ª Edição

Use a Cabeça! JavaScript

Use a Cabeça! Padrões de Projetos - 2ª Edição

Use a Cabeça! Programação

Use a Cabeça! PHP & MySQL

Use a Cabeça! PMP

Use a Cabeça! Rails - 2ª Edição

Use a Cabeça! Redes de Computadores

Use a Cabeça! Servlets & JSP - 2ª Edição

Use a Cabeça! SQL

Use a Cabeça! Web Design

Use a Cabeça! Python

Não seria um sonho se existisse um livro Python que não fizesse você querer estar em qualquer lugar, menos ficar preso na frente do seu computador escrevendo código? Eu acho que isso é apenas uma fantasia...



Paul Barry



ALTA BOOKS

E D I T O R A

Rio de Janeiro, 2012

Dedico este livro a todas aquelas pessoas generosas na comunidade Python, que ajudaram a tornar esta grande pequena linguagem a tecnologia de programação excelente que ela é.

E àquelas que tornaram a aprendizagem do Python, e suas tecnologias, complexas o suficiente para que as pessoas precisassem de um livro como este para aprendê-las.

Autor de Use a Cabeça Python

Paul ↘



Paul Barry recentemente concluiu que vem programando por quase um quarto de século, um fato que surgiu como um choque. Nesse tempo, Paul programou em diferentes linguagens de programação, viveu e trabalhou em dois países em dois continentes, casou-se, teve três filhos (bem... na verdade, sua esposa Deirdre *os teve*, mas Paul estava lá), fez bacharelado e mestrado em Informática, escreveu e coescreveu três outros livros, bem como muitos artigos técnicos para o *Linux Journal* (onde ele é editor colaborador).

Quando Paul viu pela primeira vez o *Use a Cabeça HTML com CSS & XHTML*, amou tanto que soube imediatamente que a abordagem do Use a Cabeça seria uma ótima maneira de ensinar programação. Ele ficou muito feliz então em criar o *Use a Cabeça Programação*, junto com David Griffiths, em uma tentativa de provar que o seu palpite estava correto.

Paul trabalha hoje como professor no Instituto de Tecnologia, Carlow, na Irlanda. Como parte do Departamento de Informática e Redes, Paul começa o seu dia explorando, aprendendo e ensinando tecnologias da programação, incluindo o Python, aos seus alunos.

Paul terminou recentemente seu curso de pós-graduação em Ensino e Aprendizagem, e ficou um pouco mais aliviado ao descobrir que grande parte do que ele faz está de acordo com o terceiro nível das melhores práticas atuais.

Conteúdo (Sumário)

	Introdução	xxiii
1	Conhecendo o Python: <i>Todos amam listas</i>	1
2	Compartilhando seu código: <i>Módulos de funções</i>	33
3	Arquivos e Exceções: <i>Lidando com erros</i>	73
4	Persistência: <i>Salvando dados em arquivos</i>	105
5	Compreendendo os Dados: <i>Trabalhe esses dados!</i>	139
6	Objetos de Dados Personalizados: <i>Reunindo códigos com dados</i>	173
7	Desenvolvimento Web: <i>Juntando tudo</i>	213
8	Desenvolvimento de aplicativo móvel: <i>Pequenos dispositivos</i>	255
9	Gerencie seus dados: <i>Tratamento da entrada</i>	293
10	Dimensionando seu aplicativo web: <i>Caindo na real</i>	351
11	Lidando com a complexidade: <i>Disputa de dados</i>	397
i	Sobras: <i>As dez melhores coisas (que não cobrimos)</i>	435

Conteúdo (a coisa real)

Introdução

Seu cérebro no Python. Aqui está você tentando aprender algo, enquanto seu cérebro faz o favor de se certificar que o aprendizado não se fixará. Seu cérebro está pensando: “melhor deixar espaço para coisas mais importantes, como por exemplo, quais animais selvagens evitar e se praticar snowboard nu é uma má ideia”. Então, como você enganará seu cérebro para pensar que sua vida depende de conhecer Python?

A quem se destina este livro?	xxiv
Sabemos o que você está pensando	xxv
Metacognição pensando sobre pensar	xxvii
Force seu cérebro à submissão.	xxix
Leia-me	xxx
A equipe de revisão técnica	xxxii
Agradecimentos	xxxiii

conhecendo o python

1

Todos amam listas

Você está fazendo uma pergunta: “o que torna o Python diferente?”

A resposta curta é: muitas coisas. As respostas mais longas já começam afirmando que há muitas coisas familiares também. O Python é muito parecido com qualquer outra linguagem geral de programação, com instruções, expressões, operações, funções, módulos, métodos e classes. Todas as coisas habituais, na verdade. Então, há outras coisas que o Python fornece que tornam a vida do programador – a sua vida – um pouco mais fácil. Você começará o seu tour no Python aprendendo sobre as listas. Mas, antes de chegar lá, há outra pergunta importante que precisa ser respondida...

O que há para se gostar sobre o Python?	2
Instale o Python 3	3
Use o IDLE para ajudá-lo a aprender o Python	4
Trabalhe efetivamente com o IDLE	5
Lide com dados complexos	6
Crie listas simples no Python	7
As listas são como arrays	9
Adicione mais dados à sua lista	11
Trabalhe com os dados de sua lista	15
Os loops for trabalham com listas de qualquer tamanho	16
Armazene listas dentro de listas	18
Verifique uma lista para obter outra lista	20
Dados complexos são difíceis de processar	23
Lide com os muitos níveis de listas aninhadas	24
Não repita o código; crie uma função	28
Crie uma função no Python	29
Recursão ao resgate	31
Sua caixa de ferramentas Python	32

The Holy Grail, 1975, Terry Jones & Terry Gilliam, 91 mins
Graham Chapman
Michael Palin, John Cleese, Terry Gilliam, Eric Idle & Terry Jones

compartilhando seu código

2

Módulos de funções

Código reutilizável é ótimo, mas módulo compartilhável

é melhor. Ao compartilhar seu código como um módulo Python, você abre seu código para a comunidade Python inteira... e é sempre bom compartilhar, não é? Neste capítulo, você aprenderá a criar, instalar e distribuir seus próprios módulos compartilháveis. Então, carregará seu módulo no software do site de compartilhamento do Python, para que todos possam aproveitar seu trabalho. Ao longo do caminho, você aprenderá alguns truques novos relativos às funções Python também.

É bom demais para não compartilhar	34
Transforme sua função em um módulo	35
Os módulos estão em todos os lugares	36
Comente seu código	37
Prepare sua distribuição	40
Construa sua distribuição	41
Uma rápida revisão de sua distribuição	42
Importe um módulo para usá-lo	43
Os módulos Python implementam namespaces	45
Registrando-se no site PyPI	47
Faça o upload de seu código para o PyPI	48
Bem-vindo à comunidade PyPI	49
Controle o comportamento com um argumento extra	52
Antes de escrever um código novo, pense no BIF	53
O Python faz o melhor para executar seu código	57
Rastreie seu código	58
Trabalhe no que está errado	59
Atualize o PyPI com seu novo código	60
Você mudou sua API	62
Use argumentos opcionais	63
Seu módulo suporta as duas APIs	65
Sua API ainda não está certa	66
A reputação do seu módulo é restaurada	70
Sua caixa de ferramentas Python	71



```

├── nester.py
└── setup.py

```

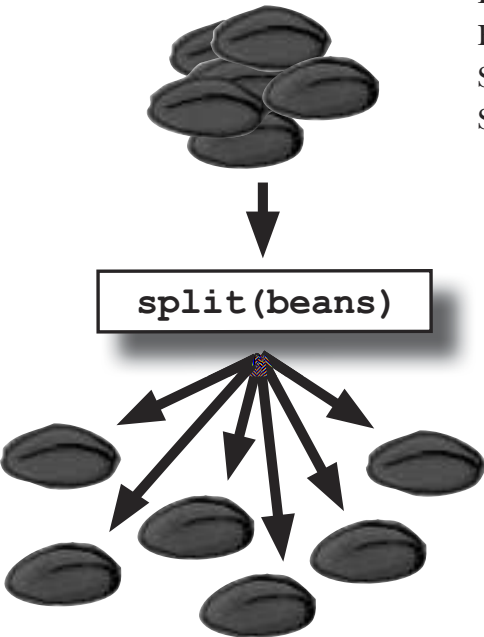
arquivos e exceções

3

Lidando com erros

Simplesmente não é suficiente processar os dados de sua lista em seu código. Você precisa fornecer os dados em seus programas com facilidade também. Não é nenhuma surpresa que o Python torne a leitura dos dados dos arquivos fácil. O que é ótimo, até você considerar o que pode dar *errado* ao interagir com dados externos aos seus programas... e há muitas coisas esperando para enganá-lo! Quando coisas ruins acontecem, você precisa de uma estratégia para solucionar o problema, e tal estratégia é lidar com quaisquer situações excepcionais usando o mecanismo de tratamento de exceção do Python mostrado neste capítulo.

Os dados são externos ao seu programa	74
São todas linhas de texto	75
Dê uma olhada nos dados	77
Conheça seus dados	79
Conheça seus métodos e peça ajuda	80
Conheça seus dados (melhor)	82
Duas abordagens muito diferentes	83
Adicione uma lógica extra	84
Lide com as exceções	88
Tente primeiro, depois recupere	89
Identifique o código a proteger	91
Continue com o erro	93
E os outros erros?	96
Adicione mais código de verificação de erros...	97
... Ou adicione outro nível de tratamento de exceção	98
Então, qual abordagem é a melhor?	99
Está pronto... exceto por um pequeno detalhe	101
Seja específico em suas exceções	102
Sua caixa de ferramentas Python	103



persistência

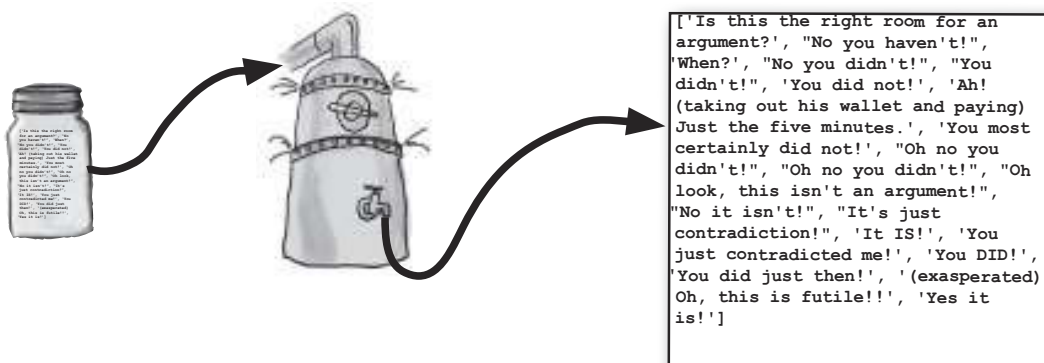
4

Salvando dados em arquivos

É muito bom ser capaz de processar seus dados

baseados em arquivos. Mas, o que acontece com seus dados quando você termina? Claro, é melhor salvá-los em um arquivo do disco, o que permite usá-los novamente mais tarde. Pegar os dados baseados em memória e armazená-los no disco é o que define a persistência. O Python suporta todas as ferramentas habituais para gravar os arquivos e também fornece algumas facilidades legais para armazenar de *forma eficiente* seus dados.

Programas produzem dados	106
Abra seu arquivo no modo de gravação	110
Os arquivos ficam abertos após uma exceção!	114
Estenda try com finally	115
Conhecer o tipo de erro não é o suficiente	117
Use with para trabalhar com arquivos	120
Os formatos padrão são inadequados para os arquivos	124
Por que não modificar print_lol()?	126
Conserve seus dados	132
Salve com dump e restaure com load	133
A E/S genérica do arquivo com pickle é o caminho a seguir	137
Sua caixa de ferramentas Python	138



compreendendo os dados

5

Trabalhe com dados!

Os dados têm todas as formas e tamanhos, formatos e codificações. Para trabalhar de forma eficaz com seus dados, muitas vezes você tem que manipulá-los e transformá-los em um formato comum para permitir processamento, classificação e armazenamento eficientes. Neste capítulo, você irá explorar os atrativos do Python que o ajudam a trabalhar com os dados sem preocupações, permitindo obter a grandeza da mudança neles.

O treinador Kelly precisa de sua ajuda	140
Escolha um dos dois caminhos	144
O problema com o tempo	148
Compreendendo as listas	155
Itere para remover as duplicatas	161
Remova as duplicatas com conjuntos	166
Sua caixa de ferramentas Python	172

Garanto que este capítulo dará a você muitos exercícios!



objetos de dados personalizados

6

Reunindo código com dados**É importante coincidir sua escolha de estrutura de dados com seus dados.**

E essa escolha pode fazer uma grande diferença na complexidade do seu código. No Python, apesar de muito úteis, as listas e os conjuntos não são as únicas opções. O **dicionário** Python permite que você organize seus dados para fazer uma pesquisa rápida *associando seus dados com nomes*, não números. E quando as estruturas de dados predefinidas do Python não servem bem, a instrução da **classe** Python permite que você defina uma própria. Este capítulo mostra como fazer.

O treinador Kelly está de volta (com um novo formato de arquivo)	174
Use um dicionário para associar dados	178
Agrupe seu código e seus dados em uma classe	189
Defina uma classe	190
Use a classe para definir classes	191
A importância de <i>self</i>	192
O primeiro argumento de cada método é <i>self</i>	193
Herde a lista predefinida do Python	204
O treinador Kelly está impressionado	211
Sua caixa de ferramentas Python	212



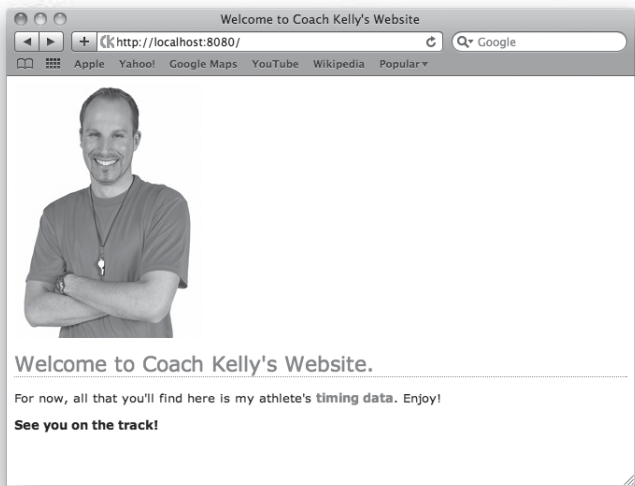
desenvolvimento web

7

Juntando tudo

Cedo ou tarde, você vai querer compartilhar seu aplicativo com muitas pessoas. Você tem muitas opções para fazer isso. Abra seu código no PyPI, envie muitos e-mails, coloque seu código em um CD ou USB, ou simplesmente instale seu aplicativo manualmente nos computadores das pessoas que precisam dele. Parece muito trabalhoso... sem mencionar a chatice. Além disso, o que acontece quando você produz uma versão melhor do seu código? O que acontece, então? Como você gerencia a atualização? Encaremos: é um problema tal que você realmente inventará desculpas criativas para não fazê-lo. Felizmente, você não precisa fazer nada disso: basta criar um aplicativo web. E como este capítulo demonstra, usar o Python para o desenvolvimento web é fácil.

É bom compartilhar	214
Você pode colocar seu programa na Web	215
O que o aplicativo web precisa fazer?	218
Planeje seu aplicativo web com MVC	221
Modele seus dados	222
Visualize sua interface	226
Controle seu código	234
O CGI permite que seu servidor web execute programas	235
Exiba a lista de atletas	236
O temido erro 404!	242
Crie outro script CGI	244
Ative o rastreamento CGI para ajudar com os erros	248
Uma pequena mudança pode fazer toda a diferença	250
Seu aplicativo web é um sucesso!	252
Sua caixa de ferramentas Python	253



8

desenvolvimento de aplicativo móvel

Pequenos dispositivos

Colocar seus dados na web abre todos os tipos de possibilidades.

Qualquer pessoa de qualquer lugar pode não só interagir com seu aplicativo web, como também está, cada vez mais, fazendo isso a partir de uma coleção de diversos dispositivos de computação: PCs, laptops, tablets, palmtops e até telefones celulares. E não são só os humanos que interagem com o seu aplicativo web com os quais você tem que se preocupar e suportar: *bots* são pequenos programas que podem automatizar as interações Web e normalmente desejam seus dados, não seu amigável HTML. Neste capítulo, você irá explorar o Python no celular do treinador Kelly para escrever um aplicativo que interaja com os dados do seu aplicativo web.

O mundo está ficando menor	256
O treinador Kelly está no Android	257
Não se preocupe com o Python 2	259
Configure seu ambiente de desenvolvimento	260
Configure o SDK e o emulador	261
Instale e configure o Android Scripting	262
Adicione o Python à sua instalação SL4A	263
Teste o Python no Android	264
Defina as exigências do seu aplicativo	266
A API do SL4A Android	274
Selecione em uma lista no Android	278
O script CGI de dados do atleta	281
Os dados parecem ter mudado de tipo	284
O JSON não pode lidar com seus tipos de dados personalizados	285
Execute o aplicativo em um telefone real	288
Configure o AndFTP	289
O treinador está feliz com seu aplicativo	290
Sua caixa de ferramentas Python	291



gerencie seus dados

9

Tratamento da entrada

A Web e o seu telefone não são apenas ótimas maneiras de exibir dados.

Também são ótimas ferramentas para aceitar a entrada de seus usuários. Naturalmente, uma vez que seu aplicativo web aceita dados, ele precisa colocá-los em algum lugar e as escolhas que você faz quando decide quais são e onde estão “esses lugares” são, muitas vezes, a diferença entre um aplicativo web fácil de aumentar e estender, e um que não é. Neste capítulo, você estenderá seu aplicativo web para aceitar os dados da web (através de um navegador ou um telefone Android), bem como verá e aperfeiçoará seus serviços de gerenciamento dos dados de *back-end*.

O aplicativo dos tempos dos atletas obteve alcance nacional	294
Use um formulário ou uma caixa de diálogo para aceitar a entrada	295
Crie um modelo de formulário HTML	296
Os dados são entregues ao seu script CGI	300
Peça a entrada em seu telefone Android	304
É hora de atualizar os dados do seu servidor	308
Evite as condições <i>race</i>	309
Você precisa de um mecanismo de armazenamento de dados melhor	310
Use um sistema de gerenciamento do banco de dados	312
Python inclui o SQLite	313
Explore a API do banco de dados do Python	314
A API do banco de dados como código Python	315
Um pequeno projeto de banco de dados vai longe	316
Defina seu esquema do banco de dados	317
Como são os dados?	318
Transfira os dados do arquivo pickle para o SQLite	321
Qual ID é atribuído a qual atleta?	322
Insira seus dados de tempo	323
Ferramentas de gerenciamento de dados SQLite	326
Integre o SQLite em seu aplicativo web existente	327
Você ainda precisa da lista dos nomes	332
Obtenha os detalhes de um atleta com base no ID	333
Você precisa corrigir seu aplicativo Android também	342
Atualize os dados do atleta com base no SQLite	348
A NUAC está na lua!	349
Sua caixa de ferramentas Python	350



10

dimensionando seu aplicativo web

Caindo na real

A web é um ótimo lugar para hospedar seu aplicativo... até que as coisas caiam na real. Cedo ou tarde, você irá tirar a

sorte grande e seu aplicativo web será um *sucesso estrondoso*. Quando isso acontecer, seu aplicativo web passará de um punhado de acessos por dia para milhares, possivelmente dezenas de milhares ou *até mais*. Você estará pronto? Seu servidor Web lidará com a **carga**? Como você saberá? Qual será o **custo**? Quem *pagará*? Seu modelo de dados pode dimensionar-se para milhões e milhões de itens de dados sem *se arrastar*? Ter um aplicativo web instalado e funcionando é fácil com o Python e agora, graças ao Google App Engine, **dimensionar** um aplicativo web Python é possível também.

Existem observadores de baleias em todos os lugares	352
O HFWWG precisa automatizar-se	353
Construa um aplicativo web com o Google App Engine	354
Baixe e instale o App Engine	355
Certifique-se que o App Engine esteja funcionando	356
O App Engine usa o padrão MVC	359
Modele seus dados com o App Engine	360
Para que serve um modelo sem visualização?	363
Use modelos no App Engine	364
A estrutura de validação de formulário do Django	368
Verifique seu formulário	369
Controlando seu aplicativo web App Engine	370
Limite a entrada fornecendo opções	376
Conheça a “tela em branco da morte”	378
Processe o POST em seu aplicativo web	379
Coloque seus dados em um armazenamento de dados	380
Não quebre o “princípio da robustez”	384
Aceite praticamente qualquer data e hora	385
Parece que você não acabou ainda	388
Às vezes, a menor mudança pode fazer toda a diferença...	389
Capture o Google ID de seu usuário também	390
Implemente seu aplicativo web na nuvem do Google	391
Seu aplicativo Web HFWWG está implementado!	394
Sua caixa de ferramentas Python	395



lidando com a complexidade

11

Disputa de dados

É ótimo quando você pode aplicar o Python a uma área de domínio específico. Seja um desenvolvimento Web, gerenciamento do banco de dados, sejam aplicativos móveis, o Python ajuda você a fazer o trabalho não atrapalhando o modo como codifica sua solução. E depois, há outros tipos de problemas: aqueles que você não pode categorizar ou anexar a um domínio. Problemas que em si são tão únicos, que você tem que vê-los de uma maneira diferente e altamente específica. Criar soluções de software sob medida para esses tipos de problemas é uma área onde o Python se destaca. Neste seu capítulo final, você estenderá suas habilidades no Python ao limite e resolverá problemas ao longo do caminho.

Qual é uma boa meta de tempo para a próxima corrida?	398
Então... qual é o problema?	400
Comece com os dados	401
Armazene cada tempo como um dicionário	407
Disseque o código de previsão	409
Obtenha a entrada a partir do seu usuário	413
Obter entrada gera um problema	414
Pesquise a correspondência mais próxima	416
O problema está no tempo	418
O módulo hora-segundos-hora	419
O problema ainda está no tempo...	422
Porta para o Android	424
Seu aplicativo Android é um monte de caixas de diálogos	425
Reúna seu aplicativo	429
Seu aplicativo está pronto!	431
Sua caixa de ferramentas Python	432

File Edit View Insert Format Form Tools Help									
Formula: V02									
	A	B	C	D	E	F	G	H	I
1	V02	84.8	82.9	81.1	79.3	77.5	75.8	74.2	72.5
2	2mi	8:00	8:10	8:21	8:33	8:44	8:56	9:08	9:20
3	5k	12:49	13:06	13:24	13:42	14:00	14:19	14:38	14:58
4	5mi	21:19	21:48	22:17	22:47	23:18	23:50	24:22	24:55
5	10k	26:54	27:30	28:08	28:45	29:24	30:04	30:45	31:26
6	15k	41:31	42:27	43:24	44:23	45:23	46:24	47:27	48:31
7	10mi	44:46	45:46	46:48	47:51	48:56	50:02	51:09	52:18
8	20k	56:29	57:45	59:03	1:00:23	1:01:45	1:03:08	1:04:33	1:06:00
9	13.1mi	59:49	1:01:09	1:02:32	1:03:56	1:05:23	1:06:51	1:08:21	1:09:53
10	25k	1:11:43	1:13:20	1:14:59	1:16:40	1:18:24	1:20:10	1:21:58	1:23:49
11	30k	1:27:10	1:19:08	1:31:08	1:33:11	1:35:17	1:37:26	1:39:37	1:41:52
12	Marathon	2:05:34	2:08:24	2:11:17	2:14:15	2:17:16	2:20:21	2:23:31	2:26:44

sobras



As dez melhores coisas (que não cobrimos)

Você percorreu um longo caminho. Mas, aprender o Python é uma atividade que nunca para. Quanto mais Python você codificar, mais precisará aprender novas maneiras de fazer certas coisas. Você precisará dominar novas ferramentas e novas técnicas também. Simplesmente não há espaço suficiente neste livro para mostrar tudo o que você possivelmente precisaria saber sobre o Python. Então, aqui vai nossa lista das dez melhores coisas que não cobrimos e que você pode querer saber mais na próxima vez.

1: Usando um IDE “profissional”	436
2: Lutando com o escopo	437
3: Testando	438
4: Os recursos avançados da linguagem	439
5: Expressões regulares	440
6: Mais sobre as estruturas Web	441
7: Mapeadores relacionais do objeto e NoSQL	442
8: Programando as GUIs	443
9: Coisas a evitar	444
10: Outros livros	445

