

Aprendendo Padrões de Projeto em Python

Tire proveito da eficácia dos padrões de projeto (design patterns) em Python para resolver problemas do mundo real em arquitetura e design de software

Chetan Giridhar

Novatec

Copyright © Packt Publishing 2015. First published in the English language under the title “Learning Python Design Patterns – Second Edition” (9781785888038)

Copyright © Packt Publishing 2015. Publicação original em inglês intitulada “Learning Python Design Patterns – Second Edition” (9781785888038). Esta tradução é publicada e vendida com a permissão da Packt Publishing.

© Novatec Editora Ltda. [2016].

Todos os direitos reservados e protegidos pela Lei 9.610 de 19/02/1998. É proibida a reprodução desta obra, mesmo parcial, por qualquer processo, sem prévia autorização, por escrito, do autor e da Editora.

Editor: Rubens Prates

Tradução: Lúcia A. Kinoshita

Revisão gramatical: Marta Almeida de Sá

Editoração eletrônica: Carolina Kuwabata

ISBN: 978-85-7522-523-3

Histórico de impressões:

Setembro/2016 Primeira edição

Novatec Editora Ltda.

Rua Luís Antônio dos Santos 110

02460-000 – São Paulo, SP – Brasil

Tel.: +55 11 2959-6529

E-mail: novatec@novatec.com.br

Site: www.novatec.com.br

Twitter: twitter.com/novateceditora

Facebook: facebook.com/novatec

LinkedIn: linkedin.com/in/novatec

Sumário

Apresentação	11
Sobre o autor	14
Sobre o revisor	15
Prefácio	16
Capítulo 1 ■ Introdução aos padrões de projeto	20
Compreendendo a programação orientada a objetos.....	20
Objetos.....	21
Classes	21
Métodos	22
Principais aspectos da programação orientada a objetos	22
Encapsulamento	22
Polimorfismo	23
Herança.....	24
Abstração	24
Composição.....	25
Princípios do design orientado a objetos	26
Princípio do aberto/fechado	26
Princípio da inversão de controle	26
Princípio da segregação de interfaces	27
Princípio da responsabilidade única.....	27
Princípio da substituição	28
Conceito de padrões de projeto.....	28
Vantagens dos padrões de projeto	30
Taxonomia dos padrões de projeto	30
Contexto – aplicabilidade dos padrões de projeto.....	31

Padrões para linguagens dinâmicas.....	31
Classificando os padrões	32
Padrões de criação	32
Padrões estruturais	32
Padrões comportamentais	33
Resumo	33
Capítulo 2 ■ Padrão de projeto Singleton	34
Compreendendo o padrão de projeto Singleton	34
Implementando um Singleton clássico em Python	35
Instanciação preguiçosa no padrão Singleton	36
Singletons no nível de módulo	37
Padrão Singleton Monostate	38
Singletons e metaclasses	39
Um cenário do mundo real – o padrão Singleton, parte 1	41
Um cenário do mundo real – o padrão Singleton, parte 2	43
Desvantagens do padrão Singleton.....	45
Resumo	46
Capítulo 3 ■ Padrão Factory – construindo fábricas para criar objetos	47
Compreendendo o padrão Factory.....	48
Padrão Simple Factory.....	49
Padrão Factory Method.....	51
Implementando o Factory Method.....	52
Vantagens do padrão Factory Method.....	54
Padrão Abstract Factory	55
Implementando o padrão Abstract Factory	56
Comparação entre Factory Method e Abstract Factory	59
Resumo	60
Capítulo 4 ■ Padrão Façade – sendo adaptável com o Façade	61
Compreendendo os padrões de projeto estruturais	62
Compreendendo o padrão de projeto Façade.....	63
Um diagrama de classes UML	64
Façade	64
Sistema.....	65
Cliente.....	65

Implementando o padrão Façade no mundo real	65
Princípio do conhecimento mínimo	70
Perguntas frequentes	70
Resumo	71
Capítulo 5 ■ Padrão Proxy – controlando o acesso a objetos.....	73
Compreendendo o padrão de projeto Proxy	74
Um diagrama de classes UML para o padrão Proxy.....	76
Compreendendo os diferentes tipos de proxies.....	78
Proxy virtual.....	78
Proxy remoto	78
Proxy de proteção	79
Proxy inteligente	79
Padrão Proxy no mundo real	79
Vantagens do padrão Proxy	83
Comparação entre os padrões Façade e Proxy	84
Perguntas frequentes	84
Resumo	85
Capítulo 6 ■ Padrão Observer – de olho nos objetos.....	86
Introdução aos padrões comportamentais.....	87
Compreendendo o padrão de projeto Observer	87
Um diagrama de classes UML para o padrão Observer.....	90
Padrão Observer no mundo real	91
Modelos do padrão Observer.....	96
Modelo pull	96
Modelo push.....	96
Baixo acoplamento e o padrão Observer	97
Padrão Observer – vantagens e desvantagens	98
Perguntas frequentes	98
Resumo	99
Capítulo 7 ■ Padrão Command – encapsulando chamadas.....	100
Introdução ao padrão Command.....	101
Compreendendo o padrão de projeto Command	102
Um diagrama de classes UML para o padrão Command	103
Implementando o padrão Command no mundo real	106
Considerações de design	106

Vantagens e desvantagens dos padrões Command.....	110
Perguntas frequentes	111
Resumo	111
Capítulo 8 ■ Padrão Template Method – encapsulando algoritmos	112
Definindo o padrão Template Method	113
Compreendendo o padrão de projeto Template Method	115
Um diagrama de classes UML para o padrão Template Method.....	117
Padrão Template Method no mundo real	119
Padrão Template Method – hooks.....	124
Princípio de Hollywood e o Template Method.....	124
As vantagens e as desvantagens do padrão Template Method.....	125
Perguntas frequentes	125
Resumo	126
Capítulo 9 ■ Modelo-Visão-Controlador – padrões compostos.....	127
Uma introdução aos padrões compostos	128
Padrão Modelo-Visão-Controlador	128
Modelo – conhecimento da aplicação	131
Visão – a aparência	131
Controlador – a cola.....	132
Um diagrama de classes UML para o padrão de projeto MVC	134
Padrão MVC no mundo real	136
Módulos	137
Vantagens do padrão MVC.....	144
Perguntas frequentes	144
Resumo	145
Capítulo 10 ■ Padrão de projeto State	146
Definindo o padrão de projeto State.....	146
Compreendendo o padrão de projeto State.....	147
Compreendendo o padrão de projeto State com um diagrama UML ...	149
Um exemplo simples do padrão de projeto State.....	150
O padrão de projeto State com implementação em Python v3.5	152
Vantagens e desvantagens do padrão State	155
Resumo	156

Capítulo 11 ■ Antipadrões.....	157
Uma introdução aos antipadrões	158
Antipadrões no desenvolvimento de software.....	159
Código espaguete.....	160
Martelo de Ouro.....	160
Fluxo de Lava	161
Programação do tipo copiar e colar ou cortar e colar	162
Antipadrões na arquitetura de software	163
Reinventando a roda	163
Vendor lock-in	164
Design by Committee	165
Resumo	166