

1. Record Nr.	UNINA9910809979403321
Autore	Giridhar Chetan
Titolo	Learning Python design patterns : leverage the power of Python design patterns to solve real-world problems in software architecture and design // Chetan Giridhar ; [foreword by Anand B. Pillai, board member, Python Software Foundation]
Pubbl/distr/stampa	Birmingham : , : Packt Publishing, , 2016
ISBN	1-78588-737-8
Edizione	[2nd ed.]
Descrizione fisica	1 online resource (164 p.)
Collana	Community experience distilled
Soggetti	Python (Computer program language) Software patterns
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Cover ; Copyright; Credits; Foreword; About the Author; About the Reviewer; www.PacktPub.com; Table of Contents; Preface; Chapter 1: Introduction to Design Patterns ; Understanding object-oriented programming; Objects; Classes; Methods; Major aspects of object-oriented programming; Encapsulation; Polymorphism; Inheritance; Abstraction; Composition; Object-oriented design principles; The open/close principle; The inversion of control principle; The interface segregation principle; The single responsibility principle; The substitution principle; The concept of design patterns Advantages of design patterns Taxonomy of design patterns; Context - the applicability of design patterns; Patterns for dynamic languages; Classifying patterns; Creational patterns::; Structural patterns; Behavioral patterns; Summary; Chapter 2: The Singleton Design Pattern ; Understanding the Singleton design pattern; Implementing a classical Singleton in Python; Lazy instantiation in the Singleton pattern; Module-level Singletons; The Monostate Singleton pattern; Singletons and metaclasses; A real-world scenario - the Singleton pattern, part 1 A real-world scenario - the Singleton pattern, part 2 Drawbacks of the Singleton pattern; Summary; Chapter 3: Factory Pattern - Building Factories to Create Objects ; Understanding the Factory pattern; The Simple Factory pattern; The Factory Method pattern; Implementing the

Factory Method; Advantages of the Factory method pattern; The Abstract Factory pattern; Implementing the Abstract Factory pattern; The Factory Method versus Abstract Factory; Summary; Chapter 4: The Facade Pattern - Being Adaptive with Facade ; Understanding Structural design patterns  
Understanding the Facade design patternA UML class diagram; Facade; System; Client; Implementing the Facade pattern in the real world; The principle of least knowledge; Frequently asked questions; Summary; Chapter 5: The Proxy Pattern - Controlling Object Access ;  
Understanding the Proxy design pattern; A UML class diagram for the Proxy pattern; Understanding different types of Proxies; A virtual proxy; A remote proxy; A protective proxy; A smart proxy; The Proxy pattern in the real world; Advantages of the Proxy pattern; Comparing the Facade and Proxy patterns; Frequently asked questions  
SummaryChapter 6: The Observer Pattern - Keeping Objects in the Know ; Introducing Behavioral patterns; Understanding the Observer design pattern; A UML class diagram for the Observer pattern; The Observer pattern in the real world; The Observer pattern methods; The pull model; The push model; Loose coupling and the Observer pattern; The Observer pattern - advantages and disadvantages; Frequently asked questions; Summary; Chapter 7: The Command Pattern - Encapsulating Invocation ; Introducing the Command pattern; Understanding the Command design pattern  
A UML class diagram for the Command pattern

---

## Sommario/riassunto

Leverage the power of Python design patterns to solve real-world problems in software architecture and design About This Book Understand the structural, creational, and behavioral Python design patterns Get to know the context and application of design patterns to solve real-world problems in software architecture, design, and application development Get practical exposure through sample implementations in Python v3.5 for the design patterns featured Who This Book Is For This book is for Software architects and Python application developers who are passionate about software design. It will be very useful to engineers with beginner level proficiency in Python and who love to work with Python 3.5 What You Will Learn Enhance your skills to create better software architecture Understand proven solutions to commonly occurring design issues Explore the design principles that form the basis of software design, such as loose coupling, the Hollywood principle and the Open Close principle among others Delve into the object-oriented programming concepts and find out how they are used in software applications Develop an understanding of Creational Design Patterns and the different object creation methods that help you solve issues in software development Use Structural Design Patterns and find out how objects and classes interact to build larger applications Focus on the interaction between objects with the command and observer patterns Improve the productivity and code base of your application using Python design patterns In Detail With the increasing focus on optimized software architecture and design it is important that software architects think about optimizations in object creation, code structure, and interaction between objects at the architecture or design level. This makes sure that the cost of software maintenance is low and code can be easily reused or is adaptable to change. The key to this is reusability and low maintenance in design patterns. Building on the success of the previous edition, Learning Python Design Patterns, Second Edition will help you implement real-world scenarios with Python's latest release, Python v3.5. We start by introducing design patterns from the Python perspective. As you progress through the book, you will learn about Singleton

patterns, Factory patterns, and Façade patterns in detail. After this, we'll look at how to control object access with proxy patterns. It also covers observer patterns, command patterns, a...

---