

1. Record Nr.	UNINA9910437582003321
Autore	Hunt John
Titolo	Scala design patterns : patterns for practical reuse and design // John Hunt
Pubbl/distr/stampa	Cham [Switzerland] : , : Springer, , 2013
ISBN	3-319-02192-3
Edizione	[1st ed. 2013.]
Descrizione fisica	1 online resource (xxii, 327 pages) : illustrations (some color)
Collana	Gale eBooks
Disciplina	004 005.1 005.11 005.133
Soggetti	Scala (Computer program language) Multiparadigm programming (Computer science)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references.
Nota di contenuto	Author's Note -- Part I -- Introduction -- Introduction -- Design Patterns -- UML and Scala -- Part II -- Fundamental Patterns -- Immutability -- Singleton -- Marker Trait -- Delegation -- Part III -- Code Reuse Patterns -- Lazy Parameters -- Partially Applied Functions -- Trait Based Template Operation -- Stackable Traits -- Currying and Code Reuse -- Cake Pattern -- Structural Injection -- Implicit Injection Pattern -- Part IV -- Gang of Four Patterns -- Gang of Four Design Patterns -- Gang of Four Patterns Catalogs -- Factory Operation -- Abstract Factory Pattern -- Builder -- Adapter Pattern -- Decorator -- Facade -- Flyweight -- Proxy -- Filter -- Bridge -- Chain of Responsibility -- Command -- Strategy -- Mediator -- Observer -- State -- Visitor -- Memento -- Part V -- Functional Design Patterns -- Functor -- Applicative Functor -- Monoid Pattern -- Monad Pattern -- Foldable -- Zipper -- Lens Pattern -- View Pattern -- Arrow Pattern.
Sommario/riassunto	Scala is a new and exciting programming language that is a hybrid between object oriented languages such as Java and functional languages such as Haskell. As such it has its own programming idioms and development styles. Scala Design Patterns looks at how code reuse can be successfully achieved in Scala. A major aspect of this is the

reinterpretation of the original Gang of Four design patterns in terms of Scala and its language structures (that is the use of Traits, Classes, Objects and Functions). It includes an exploration of functional design patterns and considers how these can be interpreted in Scala's uniquely hybrid style. A key aspect of the book is the many code examples that accompany each design pattern, allowing the reader to understand not just the design pattern but also to explore powerful and flexible Scala language features. Including numerous source code examples, this book will be of value to professionals and practitioners working in the field of software engineering.
