

1. Record Nr.	UNINA9910958884503321
Autore	Gifford Matt (Web application developer)
Titolo	Object-oriented programming in ColdFusion : break free from procedural programming and learn how to optimize your applications and enhance your skills using objects and design patterns / / Matt Gifford
Pubbl/distr/stampa	Birmingham [U.K.], : Packt Pub., 2010
ISBN	9786612882166 9781282882164 1282882163 9781847196330 1847196330
Edizione	[1st ed.]
Descrizione fisica	1 online resource (192 p.)
Collana	Community experience distilled
Disciplina	005.117
Soggetti	Object-oriented programming (Computer science)
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; Acknowledgement; About the Reviewer; Table of Contents; Preface; Chapter 1: Introducing ColdFusion Components; Why use CFCs?; Grouping your functions; Organizing your components; The ColdFusion component tags; Our first component; Defining a method; Returning the data; ColdFusion 9 scripted components; Creating your object; Restricting your functions to scopes; Using arguments within your methods; The Arguments scope; Redefine the function parameters; Combining your methods; Protecting your local variables; Using the Var scope Placing your Var scoped variables Naming your Var scoped variables; Accessing your CFC; Instantiating the object; Using the createObject function; Using the cfoject tag; Using the NEW operator; Using cfinvoke; The cfinvoke tag; Using cfinvokeargument; Using attributes as arguments; Using an argument collection; Passing arguments into an instance method call; As a list; As named values; As an argumentCollection; Arguments in action; Merging your functions into

one; Using cfargument to combine your methods; Creating an object constructor; Creating an init() function; The Variables scope; Calling your init() function; The This scope; Summary; Chapter 2: Effective Component Development; Pseudo-constructors; Using the pseudo method; Suppressing whitespace; Output attribute; Pseudo-constructor or init() method; Returning values and variables; returnType; Access properties; Private; Package; Public; Remote; Getting information about your CFC; Introspection; CFC Explorer; Component Doc; Document your code; Benefits of documentation; Displayname attribute; Hint attribute; Description attribute; User-defined metadata; Obtaining CFC metadata; getMetaData; GetComponentMetaData; Returning metadata; Detailed introspection; Summary; Chapter 3: Building Your First Bean; What is a Bean?; But what is it really?; The benefit of using beans; Creating our first Bean; An introduction to UML; Person object in UML; What makes a Bean a Bean; A default/no-argument constructor; Easily accessible for introspection; Completing our Projects Bean; Calling our project Bean; Populating the Bean; Read/Write Bean; Read-only Bean; Helpful objects; Implicit accessors; Summary; Chapter 4: Inheritance and Object-Oriented Concepts; What is Inheritance?; Avoiding code duplication; Inheriting our products; The Super keyword; Overriding methods; Instantiating our products; The inheritance hierarchy; Specialization; The "IS A" relationship; Polymorphism; Composition; The "HAS A" Relationship; Implied ownership; Aggregation; Summary; Chapter 5: Data Access Objects; What is a Data Access Object?; Creating a Data Access Object; The create method; Storing a new user; The read method; Handling select results; The update method; The delete method; An alternative save method; The save method; The exists method; Caching the Data Access Objects; Dependency Injection; Summary; Chapter 6: Gateways

Sommario/riassunto

Break free from procedural programming and learn how to optimize your applications and enhance your skills using objects and design patterns
