

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Beginning Solid Principles And Design Patterns For Aspnet Developers

Harness the power of Apex design patterns to build robust and scalable code architectures on the Force.com platform About This Book Apply Creational, Structural and behavioural patterns in Apex to fix governor limit issues. Have a grasp of the anti patterns to be taken care in Apex which could have adverse effect on the application. The authors, Jitendra Zaa is a salesforce MVP and Anshul Verma has 12+ years of experience in the area of application development. Who This Book Is For If you are a competent developer with working knowledge of Apex, and now want to deep dive into the world of Apex design patterns to optimize the application performance, then this book is for you. Prior knowledge of Salesforce and Force.com platform is recommended. What You Will Learn Apply OOPs principal in Apex to design a robust and efficient solution to address various facets to a business problem Get to grips with the benefits and applicability of using different design patterns in Apex Solve problems while instantiating, structuring and giving dynamic behavior to Apex classes Understand the implementation of creational, structural, behavioral, concurrency and anti-patterns in your application Follow the Apex best practices to resolve

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

governor limit issues Get clued up about the Inheritance, abstract classes, polymorphism in Apex to deal with the object mechanism Master various design patterns and determine the best out of them Explore the anti patterns that could not be applied to Apex and their appropriate solutions In Detail Apex is an on-demand programming language providing a complete set of features for building business applications – including data models and objects to manage data. Apex being a proprietor programming language from Salesforce to be worked with multi tenant environment is a lot different than traditional OOPs languages like Java and C#. It acts as a workflow engine for managing collaboration of the data between users, a user interface model to handle forms and other interactions, and a SOAP API for programmatic access and integration. Apex Design Patterns gives you an insight to several problematic situations that can arise while developing on Force.com platform and the usage of Design patterns to solve them. Packed with real life examples, it gives you a walkthrough from learning design patterns that Apex can offer us, to implementing the appropriate ones in your own application. Furthermore, we learn about the creational patterns that deal with object creation mechanism and structural patterns that helps to identify the relationship between entities. Also, the behavioural and concurrency patterns are put forward explaining the communication

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

between objects and multi-threaded programming paradigm respectively. We later on, deal with the issues regarding structuring of classes, instantiating or how to give a dynamic behaviour at a runtime, with the help of anti-patterns. We learn the basic OOPs principal in polymorphic and modular way to enhance its capability. Also, best practices of writing Apex code are explained to differentiate between the implementation of appropriate patterns. This book will also explain some unique patterns that could be applied to get around governor limits. By the end of this book, you will be a maestro in developing your applications on Force.com for Salesforce Style and approach This book is a step-by-step guide, complete with well-tested programs and real world situations to solve your common occurring problems in Apex design by using the anti-patterns. It gets crackling from exploring every appropriate solution to comparing the best one as per OOps principal.

Praise for Design Patterns in Ruby " Design Patterns in Ruby documents smart ways to resolve many problems that Ruby developers commonly encounter. Russ Olsen has done a great job of selecting classic patterns and augmenting these with newer patterns that have special relevance for Ruby. He clearly explains each idea, making a wealth of experience available to Ruby developers for their own daily work." —Steve Metsker, Managing

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Consultant with Dominion Digital, Inc. "This book provides a great demonstration of the key 'Gang of Four' design patterns without resorting to overly technical explanations. Written in a precise, yet almost informal style, this book covers enough ground that even those without prior exposure to design patterns will soon feel confident applying them using Ruby. Olsen has done a great job to make a book about a classically 'dry' subject into such an engaging and even occasionally humorous read." —Peter Cooper "This book renewed my interest in understanding patterns after a decade of good intentions. Russ picked the most useful patterns for Ruby and introduced them in a straightforward and logical manner, going beyond the GoF's patterns. This book has improved my use of Ruby, and encouraged me to blow off the dust covering the GoF book." —Mike Stok "Design Patterns in Ruby is a great way for programmers from statically typed objectoriented languages to learn how design patterns appear in a more dynamic, flexible language like Ruby." —Rob Sanheim, Ruby Ninja, Relevance Most design pattern books are based on C++ and Java. But Ruby is different—and the language's unique qualities make design patterns easier to implement and use. In this book, Russ Olsen demonstrates how to combine Ruby's power and elegance with patterns, and write more sophisticated, effective software with far fewer lines of code. After reviewing the history, concepts, and goals

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

of design patterns, Olsen offers a quick tour of the Ruby language—enough to allow any experienced software developer to immediately utilize patterns with Ruby. The book especially calls attention to Ruby features that simplify the use of patterns, including dynamic typing, code closures, and "mixins" for easier code reuse. Fourteen of the classic "Gang of Four" patterns are considered from the Ruby point of view, explaining what problems each pattern solves, discussing whether traditional implementations make sense in the Ruby environment, and introducing Ruby-specific improvements. You'll discover opportunities to implement patterns in just one or two lines of code, instead of the endlessly repeated boilerplate that conventional languages often require. Design Patterns in Ruby also identifies innovative new patterns that have emerged from the Ruby community. These include ways to create custom objects with metaprogramming, as well as the ambitious Rails-based "Convention Over Configuration" pattern, designed to help integrate entire applications and frameworks. Engaging, practical, and accessible, Design Patterns in Ruby will help you build better software while making your Ruby programming experience more rewarding.

There is always room for improving one's coding ability, and SOLID design principles offer one way to see marked improvements in final output. With SOLID Principles Succinctly, author Gaurav Kumar Arora will instruct you in

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

how to use SOLID principles to take your programming skills to the next level.

Practical Software Architecture Solutions from the Legendary Robert C. Martin ("Uncle Bob") By applying universal rules of software architecture, you can dramatically improve developer productivity throughout the life of any software system. Now, building upon the success of his best-selling books Clean Code and The Clean Coder, legendary software craftsman Robert C. Martin ("Uncle Bob") reveals those rules and helps you apply them. Martin's Clean Architecture doesn't merely present options. Drawing on over a half-century of experience in software environments of every imaginable type, Martin tells you what choices to make and why they are critical to your success. As you've come to expect from Uncle Bob, this book is packed with direct, no-nonsense solutions for the real challenges you'll face—the ones that will make or break your projects. Learn what software architects need to achieve—and core disciplines and practices for achieving it Master essential software design principles for addressing function, component separation, and data management See how programming paradigms impose discipline by restricting what developers can do Understand what's critically important and what's merely a "detail" Implement optimal, high-level structures for web, database, thick-client,

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

console, and embedded applications Define appropriate boundaries and layers, and organize components and services See why designs and architectures go wrong, and how to prevent (or fix) these failures Clean Architecture is essential reading for every current or aspiring software architect, systems analyst, system designer, and software manager—and for every programmer who must execute someone else's designs. Register your product for convenient access to downloads, updates, and/or corrections as they become available.

Reusable Approaches in C# and F# for Object-Oriented Software Design

Apex Design Patterns

Clean Code in Python

Principles, Patterns, and Practices

Agile Principles, Patterns, and Practices in C#

Hands-On Dependency Injection in Go

The Melaleuca Story

Agile coding with design patterns and SOLID principles As every developer knows, requirements are subject to change. But when you build adaptability into your code, you can respond to change more easily and avoid disruptive rework. Focusing on Agile programming, this book describes the best practices, principles, and patterns that

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

enable you to create flexible, adaptive code--and deliver better business value. Expert guidance to bridge the gap between theory and practice Get grounded in Scrum: artifacts, roles, metrics, phases Organize and manage architectural dependencies Review best practices for patterns and anti-patterns Master SOLID principles: single-responsibility, open/closed, Liskov substitution Manage the versatility of interfaces for adaptive code Perform unit testing and refactoring in tandem See how delegation and abstraction impact code adaptability Learn best ways to implement dependency interjection Apply what you learn to a pragmatic, agile coding project Get code samples at: <http://github.com/garymclean/AdaptiveCode> Explore various dependency injection methods in Go such as monkey patching, constructor injection, and method injection Key Features Learn to evaluate Code UX and make it better Explore SOLID principles and understand how they relate to dependency injection Use Google's wire framework to simplify dependence management Book Description Hands-On Dependency Injection in Go takes you on a journey, teaching you about refactoring existing code to adopt dependency injection (DI) using various methods available in Go. Of the six methods introduced

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

in this book, some are conventional, such as constructor or method injection, and some unconventional, such as just-in-time or config injection. Each method is explained in detail, focusing on their strengths and weaknesses, and is followed with a step-by-step example of how to apply it. With plenty of examples, you will learn how to leverage DI to transform code into something simple and flexible. You will also discover how to generate and leverage the dependency graph to spot and eliminate issues. Throughout the book, you will learn to leverage DI in combination with test stubs and mocks to test otherwise tricky or impossible scenarios. Hands-On Dependency Injection in Go takes a pragmatic approach and focuses heavily on the code, user experience, and how to achieve long-term benefits through incremental changes. By the end of this book, you will have produced clean code that's easy to test. What you will learn Understand the benefits of DI Explore SOLID design principles and how they relate to Go Analyze various dependency injection patterns available in Go Leverage DI to produce high-quality, loosely coupled Go code Refactor existing Go code to adopt DI Discover tools to improve your code's testability and test coverage Generate and interpret Go dependency

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

graphs Who this book is for Hands-On Dependency Injection in Go is for programmers with a few year s experience in any language and a basic understanding of Go. If you wish to produce clean, loosely coupled code that is inherently easier to test, this book is for you.

When you use Hibernate in your projects, you quickly recognize that you need to do more than just add @Entity annotations to your domain model classes. Real-world applications often require advanced mappings, complex queries, custom data types and caching. Hibernate can do all of that. You just have to know which annotations and APIs you need to use. Hibernate Tips - More than 70 solutions to common Hibernate problems shows you how to efficiently implement your persistence layer with Hibernate's basic and advanced features. Each Hibernate Tip consists of one or more code samples and an easy to follow step-by-step explanation. You can also download an example project with executable test cases for each Hibernate Tip. Throughout this book, you will get more than 70 ready-to-use solutions that show you how to:

- Define standard mappings for basic attributes and entity associations.*
- Implement your own attribute mappings and support custom data types.*
- Use Hibernate's Java 8 support and other*

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

proprietary features. - Read data from the database with JPQL, Criteria API, and native SQL queries. - Call stored procedures and database functions. This book is for developers who are already working with Hibernate and who are looking for solutions for their current development tasks. It's not a book for beginners who are looking for extensive descriptions of Hibernate's general concepts. The tips are designed as self-contained recipes which provide a specific solution and can be accessed when needed. Most of them contain links to related tips which you can follow if you want to dive deeper into a topic or need a slightly different solution. There is no need to read the tips in a specific order. Feel free to read the book from cover to cover or to just pick the tips that help you in your current project.

Beginning C# Object-Oriented Programming brings you into the modern world of development as you master the fundamentals of programming with C# and learn to develop efficient, reusable, elegant code through the object-oriented programming (OOP) methodology. Take your skills out of the 20th century and into this one with Dan Clark's accessible, quick-paced guide to C# and object-oriented programming, completely updated for .NET 4.0 and C# 4.0. As you

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

develop techniques and best practices for coding in C#, one of the world's most popular contemporary languages, you'll experience modeling a "real world" application through a case study, allowing you to see how both C# and OOP (a methodology you can use with any number of languages) come together to make your code reusable, modern, and efficient. With more than 30 fully hands-on activities, you'll discover how to transform a simple model of an application into a fully-functional C# project, including designing the user interface, implementing the business logic, and integrating with a relational database for data storage. Along the way, you will explore the .NET Framework, the creation of a Windows-based user interface, a web-based user interface, and service-oriented programming, all using Microsoft's industry-leading Visual Studio 2010, C#, Silverlight, the Entity Framework, and more.

Solid Principles Succinctly

SOLID is Not Solid: Five Object-Oriented Principles To Create a Codebase Everyone Will Hate

Mastering C++ Programming

Adaptive Code via C#

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

Invoke TDD principles for end-to-end application development, 2nd Edition

Design Principles and Problems

A Handbook of Agile Software Craftsmanship

This book is the result of a unique collaboration between an artist who has taught basic design and an experienced professional writer of college textbooks. Together we have tried to create a book that prepares a solid foundation for studying all the fine and applied arts and is at the same time readable, interesting, and clear.

Object orientation is a lie. Reusable, flexible components have failed. The SOLID Principles of Object-Oriented Design still cling to these lies, sending developers down so many wrong paths. In less than 70 pages, this book presents five broadsides against each principle, tracing their history, demonstrating their flaws, and taking their advice to an hilarious degree all to prove a point: you can build good software by focusing on the problem at hand, and discussing the code you're writing, not some nebulous set of principles.

Get up to speed with creational, structural, behavioral and concurrent patterns in Delphi to write clear, concise and effective code
Key Features
Delve into the core patterns and components of Delphi in order to master your application's design
Brush up on tricks, techniques, and best practices to solve common design and architectural challenges
Choose the right patterns to improve your program's

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

efficiency and productivity

Book Description Design patterns have proven to be the go-to solution for many common programming scenarios. This book focuses on design patterns applied to the Delphi language. The book will provide you with insights into the language and its capabilities of a runtime library. You'll start by exploring a variety of design patterns and understanding them through real-world examples. This will entail a short explanation of the concept of design patterns and the original set of the 'Gang of Four' patterns, which will help you in structuring your designs efficiently. Next, you'll cover the most important 'anti-patterns' (essentially bad software development practices) to aid you in steering clear of problems during programming. You'll then learn about the eight most important patterns for each creational, structural, and behavioral type. After this, you'll be introduced to the concept of 'concurrency' patterns, which are design patterns specifically related to multithreading and parallel computation. These will enable you to develop and improve an interface between items and harmonize shared memories within threads. Toward the concluding chapters, you'll explore design patterns specific to program design and other categories of patterns that do not fall under the 'design' umbrella. By the end of this book, you'll be able to address common design problems encountered while developing applications and feel confident while building scalable projects. What you will learn

Gain insights into the concept of design patterns

Study modern programming techniques with Delphi

Keep

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

up to date with the latest additions and program design techniques in DelphiGet to grips with various modern multithreading approachesDiscover creational, structural, behavioral, and concurrent patternsDetermine how to break a design problem down into its component partsWho this book is for Hands-On Design Patterns with Delphi is aimed at beginner-level Delphi developers who want to build scalable and robust applications. Basic knowledge of Delphi is a must.

Summary .NET Core in Action shows .NET developers how to build professional software applications with .NET Core. Learn how to convert existing .NET code to work on multiple platforms or how to start new projects with knowledge of the tools and capabilities of .NET Core. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology .NET Core is an open source framework that lets you write and run .NET applications on Linux and Mac, without giving up on Windows. Built for everything from lightweight web apps to industrial-strength distributed systems, it's perfect for deploying .NET servers to any cloud platform, including AWS and GCP. About the Book .NET Core in Action introduces you to cross-platform development with .NET Core. This hands-on guide concentrates on new Core features as you walk through familiar tasks like testing, logging, data access, and networking. As you go, you'll explore modern architectures like microservices and cloud data storage, along with practical matters like performance profiling, localization, and signing assemblies.

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

What's Inside Choosing the right tools Testing, profiling, and debugging Interacting with web services Converting existing projects to .NET Core Creating and using NuGet packages About the Reader All examples are in C#. About the Author Dustin Metzgar is a seasoned developer and architect involved in numerous .NET Core projects. Dustin works for Microsoft. Table of Contents Why .NET Core? Building your first .NET Core applications How to build with .NET Core Unit testing with xUnit Working with relational databases Simplify data access with object-relational mappers Creating a microservice Debugging Performance and profiling Building world-ready applications Multiple frameworks and runtimes Preparing for release appendix A - Frameworks and runtimes appendix B - xUnit command-line options appendix C - What's in the .NET Standard Library? appendix D - NuGet cache locations

Test Driven Development for Embedded C

An An Atypical ASP.NET Core 5 Design Patterns Guide

97 Things Every Programmer Should Know

Beginning C# Object-Oriented Programming

Learning TypeScript

Refactor your legacy code base

.NET Core in Action

Using research in neurobiology, cognitive science and learning theory, this text loads

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

patterns into your brain in a way that lets you put them to work immediately, makes you better at solving software design problems, and improves your ability to speak the language of patterns with others on your team.

Create various design patterns to master the art of solving problems using Java Key Features This book demonstrates the shift from OOP to functional programming and covers reactive and functional patterns in a clear and step-by-step manner All the design patterns come with a practical use case as part of the explanation, which will improve your productivity Tackle all kinds of performance-related issues and streamline your development Book Description Having a knowledge of design patterns enables you, as a developer, to improve your code base, promote code reuse, and make the architecture more robust. As languages evolve, new features take time to fully understand before they are adopted en masse. The mission of this book is to ease the adoption of the latest trends and provide good practices for programmers. We focus on showing you the practical aspects of smarter coding in Java. We'll start off by going over object-oriented (OOP) and functional programming (FP) paradigms, moving on to describe the most frequently used design patterns in their classical format and explain how Java's functional programming features are changing them. You will learn to enhance implementations by mixing OOP and FP, and finally get to know about the reactive programming model, where FP and OOP are used in conjunction with a view to writing better code. Gradually, the book will show you the latest trends in architecture,

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

moving from MVC to microservices and serverless architecture. We will finish off by highlighting the new Java features and best practices. By the end of the book, you will be able to efficiently address common problems faced while developing applications and be comfortable working on scalable and maintainable projects of any size. What you will learn Understand the OOP and FP paradigms Explore the traditional Java design patterns Get to know the new functional features of Java See how design patterns are changed and affected by the new features Discover what reactive programming is and why is it the natural augmentation of FP Work with reactive design patterns and find the best ways to solve common problems using them See the latest trends in architecture and the shift from MVC to serverless applications Use best practices when working with the new features Who this book is for This book is for those who are familiar with Java development and want to be in the driver's seat when it comes to modern development techniques. Basic OOP Java programming experience and elementary familiarity with Java is expected.

This book teaches you all the essential knowledge required to learn and apply time-proven SOLID principles of object-oriented design and important design patterns in ASP.NET Core 1.0 (formerly ASP.NET 5) applications. You will learn to write server-side as well as client-side code that makes use of proven practices and patterns. SOLID is an acronym popularized by Robert Martin used to describe five basic principles of good object-oriented design--Single Responsibility, Open/Closed, Liskov

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Substitution, Interface Segregation and Dependency Inversion. This book covers all five principles and illustrates how they can be used in ASP.NET Core 1.0 applications. Design Patterns are time proven solutions to commonly occurring software design problems. The most well-known catalog of design patterns comes from Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, the so-called as GoF patterns (Gang of Four patterns). This book contains detailed descriptions of how to apply Creational, Structural and Behavioral GoF design patterns along with some Patterns of Enterprise Application Architecture. Popular JavaScript patterns are covered, along with working examples of all these patterns in ASP.NET Core 1.0 and C# are included. What You Will Learn: How to apply SOLID principles to ASP.NET applications How to use Gang of Four (GoF) design patterns in ASP.NET applications Techniques for applying Patterns of Enterprise Application Architecture cataloged by Martin Fowler in ASP.NET applications How to organize code and apply design patterns in JavaScript Who This Book Is For: This book is for ASP.NET developers familiar with ASP.NET Core 1.0, C# and Visual Studio.

With the award-winning book Agile Software Development: Principles, Patterns, and Practices, Robert C. Martin helped bring Agile principles to tens of thousands of Java and C++ programmers. Now .NET programmers have a definitive guide to agile methods with this completely updated volume from Robert C. Martin and Micah Martin, Agile Principles, Patterns, and Practices in C#. This book presents a series of case

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

studies illustrating the fundamentals of Agile development and Agile design, and moves quickly from UML models to real C# code. The introductory chapters lay out the basics of the agile movement, while the later chapters show proven techniques in action. The book includes many source code examples that are also available for download from the authors' Web site. Readers will come away from this book understanding Agile principles, and the fourteen practices of Extreme Programming Spiking, splitting, velocity, and planning iterations and releases Test-driven development, test-first design, and acceptance testing Refactoring with unit testing Pair programming Agile design and design smells The five types of UML diagrams and how to use them effectively Object-oriented package design and design patterns How to put all of it together for a real-world project Whether you are a C# programmer or a Visual Basic or Java programmer learning C#, a software development manager, or a business analyst, Agile Principles, Patterns, and Practices in C# is the first book you should read to understand agile software and how it applies to programming in the .NET Framework.

Learning JavaScript Design Patterns

The Clean Coder

Abstraction, Specification, and Object-Oriented Design

Implementing Design Patterns in C# and .NET 5

Hibernate Tips

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

A SOLID adventure into architectural principles, design patterns, .NET 5, and C#
Head First Design Patterns

The Object-Oriented Thought Process Third Edition Matt Weisfeld An introduction to object-oriented concepts for developers looking to master modern application practices. Object-oriented programming (OOP) is the foundation of modern programming languages, including C++, Java, C#, and Visual Basic .NET. By designing with objects rather than treating the code and data as separate entities, OOP allows objects to fully utilize other objects' services as well as inherit their functionality. OOP promotes code portability and reuse, but requires a shift in thinking to be fully understood. Before jumping into the world of object-oriented programming languages, you must first master The Object-Oriented Thought Process. Written by a developer for developers who want to make the leap to object-oriented technologies as well as managers who simply want to understand what they are managing, The Object-Oriented Thought Process provides a solution-oriented approach to object-oriented programming. Readers will learn to understand object-oriented design with inheritance or composition, object aggregation and association, and the difference between interfaces and implementations. Readers will also become more efficient and better thinkers in terms of object-oriented development. This revised edition focuses on interoperability across various technologies, primarily using XML as the communication mechanism. A more detailed focus is placed on how business objects operate over networks, including client/server architectures and web services. "Programmers who aim to create high quality software—as all programmers should—must learn the varied subtleties of the familiar yet not so familiar beasts called objects and classes. Doing so entails careful study of books such as Matt Weisfeld's The Object-

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Oriented Thought Process.” –Bill McCarty, author of *Java Distributed Objects*, and *Object-Oriented Design in Java* Matt Weisfeld is an associate professor in business and technology at Cuyahoga Community College in Cleveland, Ohio. He has more than 20 years of experience as a professional software developer, project manager, and corporate trainer using C++, Smalltalk, .NET, and Java. He holds a BS in systems analysis, an MS in computer science, and an MBA in project management. Weisfeld has published many articles in major computer trade magazines and professional journals.

The hypothesis of this handbook is that you don't need to understand the full extent of Service Design to improve the user and customer experience. You don't need to understand all the theory to create great services. That's why each principle in this handbook is summarized in a simple rule of thumb. These simple rules of thumb should be enough for smart readers. You might find, under each principle, a little story, an example, or a study. This additional content can help you turn this principle into action.

Summary Dependency Injection Principles, Practices, and Patterns teaches you to use DI to reduce hard-coded dependencies between application components. You'll start by learning what DI is and what types of applications will benefit from it. Then, you'll work through concrete scenarios using C# and the .NET framework to implement DI in your own projects. As you dive into the thoroughly-explained examples, you'll develop a foundation you can apply to any of the many DI libraries for .NET and .NET Core. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology *Dependency Injection (DI)* is a great way to reduce tight coupling between software components. Instead of hard-coding dependencies, such as specifying a database driver, you make those connections

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

through a third party. Central to application frameworks like ASP.NET Core, DI enables you to better manage changes and other complexity in your software. About the Book Dependency Injection Principles, Practices, and Patterns is a revised and expanded edition of the bestselling classic Dependency Injection in .NET. It teaches you DI from the ground up, featuring relevant examples, patterns, and anti-patterns for creating loosely coupled, well-structured applications. The well-annotated code and diagrams use C# examples to illustrate principles that work flawlessly with modern object-oriented languages and DI libraries. What's Inside Refactoring existing code into loosely coupled code DI techniques that work with statically typed OO languages Integration with common .NET frameworks Updated examples illustrating DI in .NET Core About the Reader For intermediate OO developers. About the Authors Mark Seemann is a programmer, software architect, and speaker who has been working with software since 1995, including six years with Microsoft. Steven van Deursen is a seasoned .NET developer and architect, and the author and maintainer of the Simple Injector DI library. Table of Contents PART 1 Putting Dependency Injection on the map The basics of Dependency Injection: What, why, and how Writing tightly coupled code Writing loosely coupled code PART 2 Catalog DI patterns DI anti-patterns Code smells PART 3 Pure DI Application composition Object lifetime Interception Aspect-Oriented Programming by design Tool-based Aspect-Oriented Programming PART 4 DI Containers DI Container introduction The Autofac DI Container The Simple Injector DI Container The Microsoft.Extensions.DependencyInjection DI Container Software -- Software Engineering. Design Patterns

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

Service Design Principles 1-100: 100 Ideas to Improve the User and Customer Experience in Simple and Practical Ways.

Clean Architecture

Roles, Responsibilities, and Collaborations

A JavaScript and jQuery Developer's Guide

Dependency Injection Principles, Practices, and Patterns

Clean Code

Implement design patterns in .NET using the latest versions of the C# and F# languages. This book provides a comprehensive overview of the field of design patterns as they are used in today's developer toolbox. Using the C# programming language, Design Patterns in .NET explores the classic design pattern implementation and discusses the applicability and relevance of specific language features for the purpose of implementing patterns. You will learn by example, reviewing scenarios where patterns are applicable. MVP and patterns expert Dmitri Nesteruk demonstrates possible implementations of patterns, discusses alternatives and pattern inter-relationships, and illustrates the way that a dedicated refactoring tool (ReSharper) can be used to implement design patterns with ease. What You'll Learn Know the latest pattern implementations available in C# and F# Refer to researched and proven variations of patterns Study complete, self-contained examples including many that cover advanced scenarios Use the latest implementations of C# and Visual Studio/ReSharper Who This Book Is For Developers who have some experience in the C# language and want to expand their comprehension of the art of programming by leveraging design

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

approaches to solving modern problems

Presents practical advice on the disciplines, techniques, tools, and practices of computer programming and how to approach software development with a sense of pride, honor, and self-respect.

Tap into the wisdom of experts to learn what every programmer should know, no matter what language you use. With the 97 short and extremely useful tips for programmers in this book, you'll expand your skills by adopting new approaches to old problems, learning appropriate best practices, and honing your craft through sound advice. With contributions from some of the most experienced and respected practitioners in the industry--including Michael Feathers, Pete Goodliffe, Diomidis Spinellis, Cay Horstmann, Verity Stob, and many more--this book contains practical knowledge and principles that you can apply to all kinds of projects. A few of the 97 things you should know: "Code in the Language of the Domain" by Dan North "Write Tests for People" by Gerard Meszaros "Convenience Is Not an -ility" by Gregor Hohpe "Know Your IDE" by Heinz Kabutz "A Message to the Future" by Linda Rising "The Boy Scout Rule" by Robert C. Martin (Uncle Bob) "Beware the Share" by Udi Dahan

"Head First Object Oriented Analysis and Design is a refreshing look at subject of OOAD. What sets this book apart is its focus on learning. The authors have made the content of OOAD accessible, usable for the practitioner." Ivar Jacobson, Ivar Jacobson Consulting "I just finished reading HF OOA&D and I loved it! The thing I liked most about this book was its focus on why we do OOA&D-to write great software!" Kyle Brown, Distinguished Engineer, IBM "Hidden behind the funny pictures and crazy fonts

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

is a serious, intelligent, extremely well-crafted presentation of OO Analysis and Design. As I read the book, I felt like I was looking over the shoulder of an expert designer who was explaining to me what issues were important at each step, and why." Edward Sciore, Associate Professor, Computer Science Department, Boston College Tired of reading Object Oriented Analysis and Design books that only makes sense after you're an expert? You've heard OOA&D can help you write great software every time—software that makes your boss happy, your customers satisfied and gives you more time to do what makes you happy. But how? Head First Object-Oriented Analysis & Design shows you how to analyze, design, and write serious object-oriented software: software that's easy to reuse, maintain, and extend; software that doesn't hurt your head; software that lets you add new features without breaking the old ones. Inside you will learn how to: Use OO principles like encapsulation and delegation to build applications that are flexible Apply the Open-Closed Principle (OCP) and the Single Responsibility Principle (SRP) to promote reuse of your code Leverage the power of design patterns to solve your problems more efficiently Use UML, use cases, and diagrams to ensure that all stakeholders are communicating clearly to help you deliver the right software that meets everyone's needs. By exploiting how your brain works, Head First Object-Oriented Analysis & Design compresses the time it takes to learn and retain complex information. Expect to have fun, expect to learn, expect to be writing great software consistently by the time you're finished reading this!

***A comprehensive guide to building smart and reusable code in Java
Head First Object-Oriented Analysis and Design***

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

A Brain Friendly Guide to OOA&D

Design Patterns and Best Practices in Java

Object Design

Elements of Reusable Object-Oriented Software

Mastering JavaScript Object-Oriented Programming

Another day without Test-Driven Development means more time wasted chasing bugs and watching your code deteriorate. You thought TDD was for someone else, but it's not! It's for you, the embedded C programmer. TDD helps you prevent defects and build software with a long useful life. This is the first book to teach the hows and whys of TDD for C programmers. TDD is a modern programming practice C developers need to know. It's a different way to program---unit tests are written in a tight feedback loop with the production code, assuring your code does what you think. You get valuable feedback every few minutes. You find mistakes before they become bugs. You get early warning of design problems. You get immediate notification of side effect defects. You get to spend more time adding valuable

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

features to your product. James is one of the few experts in applying TDD to embedded C. With his 1.5 decades of training, coaching, and practicing TDD in C, C++, Java, and C# he will lead you from being a novice in TDD to using the techniques that few have mastered. This book is full of code written for embedded C programmers. You don't just see the end product, you see code and tests evolve. James leads you through the thought process and decisions made each step of the way. You'll learn techniques for test-driving code right next to the hardware, and you'll learn design principles and how to apply them to C to keep your code clean and flexible. To run the examples in this book, you will need a C/C++ development environment on your machine, and the GNU GCC tool chain or Microsoft Visual Studio for C++ (some project conversion may be needed).

Written by a world-renowned expert on programming methodology, and the winner of the 2008 Turing Award, this book shows how to build production-quality programs--programs that are reliable, easy to maintain, and

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

quick to modify. Its emphasis is on modular program construction: how to get the modules right and how to organize a program as a collection of modules. The book presents a methodology effective for either an individual programmer, who may be writing a small program or a single module in a larger one; or a software engineer, who may be part of a team developing a complex program comprised of many modules. Both audiences will acquire a solid foundation for object-oriented program design and component-based software development from this methodology. Because each module in a program corresponds to an abstraction, such as a collection of documents or a routine to search the collection for documents of interest, the book first explains the kinds of abstractions most useful to programmers: procedures; iteration abstractions; and, most critically, data abstractions. Indeed, the author treats data abstraction as the central paradigm in object-oriented program design and implementation. The author also shows, with numerous examples, how to develop informal

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

specifications that define these abstractions--specifications that describe what the modules do--and then discusses how to implement the modules so that they do what they are supposed to do with acceptable performance. Other topics discussed include: Encapsulation and the need for an implementation to provide the behavior defined by the specification Tradeoffs between simplicity and performance Techniques to help readers of code understand and reason about it, focusing on such properties as rep invariants and abstraction functions Type hierarchy and its use in defining families of related data abstractions Debugging, testing, and requirements analysis Program design as a top-down, iterative process, and design patterns The Java programming language is used for the book's examples. However, the techniques presented are language independent, and an introduction to key Java concepts is included for programmers who may not be familiar with the language.

Object technology pioneer Wirfs-Brock teams with expert

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

McKean to present a thoroughly updated, modern, and proven method for the design of software. The book is packed with practical design techniques that enable the practitioner to get the job done.

Write code that can adapt to changes. By applying this book's principles, you can create code that accommodates new requirements and unforeseen scenarios without significant rewrites. Gary McLean Hall describes Agile best practices, principles, and patterns for designing and writing code that can evolve more quickly and easily, with fewer errors, because it doesn't impede change. Now revised, updated, and expanded, Adaptive Code, Second Edition adds indispensable practical insights on Kanban, dependency inversion, and creating reusable abstractions. Drawing on over a decade of Agile consulting and development experience, McLean Hall has updated his best-seller with deeper coverage of unit testing, refactoring, pure dependency injection, and more. Master powerful new ways to:

- Write code that enables and complements Scrum, Kanban, or any other Agile framework •*

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

Develop code that can survive major changes in requirements

- *Plan for adaptability by using dependencies, layering, interfaces, and design patterns*
- *Perform unit testing and refactoring in tandem, gaining more value from both*
- *Use the “golden master” technique to make legacy code adaptive*
- *Build SOLID code with single-responsibility, open/closed, and Liskov substitution principles*
- *Create smaller interfaces to support more-diverse client and architectural needs*
- *Leverage dependency injection best practices to improve code adaptability*
- *Apply dependency inversion with the Stairway pattern, and avoid related anti-patterns*

About You This book is for programmers of all skill levels seeking more-practical insight into design patterns, SOLID principles, unit testing, refactoring, and related topics. Most readers will have programmed in C#, Java, C++, or similar object-oriented languages, and will be familiar with core procedural programming techniques.

Principles of Package Design

Build Scalable, Fast, and Reliable .NET Applications Using

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

the Most Common Design Patterns (English Edition)

Beginning SOLID Principles and Design Patterns for ASP.NET Developers

A Code of Conduct for Professional Programmers

Built on Solid Principles

Creating Reusable Software Components

Program Development in Java

Take your C++ coding to the next level by leveraging the latest features and advanced techniques to building high performing, reliable applications. About This Book Get acquainted with the latest features in C++ 17 Take advantage of the myriad of features and possibilities that C++ offers to build real-world applications Write clear and expressive code in C++, and get insights into how to keep your code error-free Who This Book Is For This book is for experienced C++ developers. If you are a novice C++ developer, then it's highly recommended that you get a solid understanding of the C++ language before reading this book What You Will Learn Write modular C++ applications in terms of the existing and newly introduced features Identify

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

***code-smells, clean up, and refactor legacy C++ applications
Leverage the possibilities provided by Cucumber and Google Test/Mock to automate test cases Test frameworks with C++
Get acquainted with the new C++17 features Develop GUI applications in C++ Build portable cross-platform applications using standard C++ features In Detail C++ has come a long way and has now been adopted in several contexts. Its key strengths are its software infrastructure and resource-constrained applications. The C++ 17 release will change the way developers write code, and this book will help you master your developing skills with C++. With real-world, practical examples explaining each concept, the book will begin by introducing you to the latest features in C++ 17. It encourages clean code practices in C++ in general, and demonstrates the GUI app-development options in C++. You'll get tips on avoiding memory leaks using smart-pointers. Next, you'll see how multi-threaded programming can help you achieve concurrency in your applications. Moving on, you'll get an in-depth understanding of the C++ Standard Template Library.***

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

We show you the concepts of implementing TDD and BDD in your C++ programs, and explore template-based generic programming, giving you the expertise to build powerful applications. Finally, we'll round up with debugging techniques and best practices. By the end of the book, you'll have an in-depth understanding of the language and its various facets. Style and approach This straightforward guide will help you level up your skills in C++ programming, be it for enterprise software or for low-latency applications like games. Filled with real-world, practical examples, this book will take you gradually up the steep learning curve that is C++.

A .NET developer's guide to crafting robust, maintainable, and flexible web apps by leveraging C# 9 and .NET 5 features and component-scale and application-scale design patterns
Key Features
Apply software design patterns effectively, starting small and progressing to cloud-scale
Discover modern application architectures such as vertical slice, clean architecture, and event-driven microservices
Explore ASP.NET design patterns, from options to full-stack web development

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

using BlazorBook Description Design patterns are a set of solutions to many of the common problems occurring in software development. Knowledge of these design patterns helps developers and professionals to craft software solutions of any scale. ASP.NET Core 5 Design Patterns starts by exploring basic design patterns, architectural principles, dependency injection, and other ASP.NET Core mechanisms. You'll explore the component scale as you discover patterns oriented toward small chunks of the software, and then move to application-scale patterns and techniques to understand higher-level patterns and how to structure the application as a whole. The book covers a range of significant GoF (Gangs of Four) design patterns such as strategy, singleton, decorator, facade, and composite. The chapters are organized based on scale and topics, allowing you to start small and build on a strong base, the same way that you would develop a program. With the help of use cases, the book will show you how to combine design patterns to display alternate usage and help you feel comfortable working with a variety of design patterns.

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Finally, you'll advance to the client side to connect the dots and make ASP.NET Core a viable full-stack alternative. By the end of the book, you'll be able to mix and match design patterns and have learned how to think about architecture and how it works. What you will learn Apply the SOLID principles for building flexible and maintainable software Get to grips with .NET 5 dependency injection Work with GoF design patterns such as strategy, decorator, and composite Explore the MVC patterns for designing web APIs and web applications using Razor Discover layering techniques and tenets of clean architecture Become familiar with CQRS and vertical slice architecture as an alternative to layering Understand microservices, what they are, and what they are not Build ASP.NET UI from server-side to client-side Blazor Who this book is for This design patterns book is for intermediate-level software and web developers with some knowledge of .NET who want to write flexible, maintainable, and robust code for building scalable web applications. Knowledge of C# programming and an understanding of web concepts like HTTP

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

is necessary.

This comprehensive, pragmatic tutorial on Agile Development and eXtreme programming, written by one of the founding fathers of Agile Development: Teaches software developers and project managers how to get projects done on time, and on budget using the power of Agile Development; Uses real-world case studies to show how to of plan, test, refactor, and pair program using eXtreme programming; Contains a wealth of reusable C++ and Java code; Focuses on solving customer oriented systems problems using UML and Design Patterns. With Learning JavaScript Design Patterns, you'll learn how to write beautiful, structured, and maintainable JavaScript by applying classical and modern design patterns to the language. If you want to keep your code efficient, more manageable, and up-to-date with the latest best practices, this book is for you. Explore many popular design patterns, including Modules, Observers, Facades, and Mediators. Learn how modern architectural patterns—such as MVC, MVP, and MVVM—are useful from the perspective of a modern web

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

application developer. This book also walks experienced JavaScript developers through modern module formats, how to namespace code effectively, and other essential topics. Learn the structure of design patterns and how they are written Understand different pattern categories, including creational, structural, and behavioral Walk through more than 20 classical and modern design patterns in JavaScript Use several options for writing modular code—including the Module pattern, Asynchronous Module Definition (AMD), and CommonJS Discover design patterns implemented in the jQuery library Learn popular design patterns for writing maintainable jQuery plugins "This book should be in every JavaScript developer's hands. It's the go-to book on JavaScript patterns that will be read and referenced many times in the future."—Andrée Hansson, Lead Front-End Developer, presis!

Test-Driven Java Development, Second Edition

Build applications using idiomatic, extensible, and concurrent design patterns in Delphi

Hands-On Design Patterns with Delphi

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Collective Wisdom from the Experts ***Agile Software Development*** ***A Craftsman's Guide to Software Structure and Design*** ***AGILE PRIN PATTS PRACTS C#_1***

This book will teach the concepts of test driven development in Java so you can build clean, maintainable and robust code Key Features Explore the most popular TDD tools and frameworks and become more proficient in building applications Create applications with better code design, fewer bugs, and higher test coverage, enabling you to get them to market quickly Implement test-driven programming methods into your development workflows Book Description Test-driven development (TDD) is a development approach that relies on a test-first procedure that emphasizes writing a test before writing the necessary code, and then refactoring the code to optimize it. The value of performing TDD with Java, one of the longest established programming languages, is to improve the productivity of programmers and the maintainability and performance of code, and develop a deeper understanding of the language and how to employ it effectively. Starting with the basics of TDD and understanding why its adoption is beneficial, this book will take you from the first steps of TDD with Java until you are confident enough to embrace the practice in your day-to-day routine. You'll be guided through setting up tools, frameworks, and the environment you need, and we will dive right into hands-on exercises with the goal of mastering one practice, tool, or framework at a time. You'll learn about the Red-Green-Refactor procedure, how to write

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

unit tests, and how to use them as executable documentation. With this book, you'll also discover how to design simple and easily maintainable code, work with mocks, utilize behavior-driven development, refactor old legacy code, and release a half-finished feature to production with feature toggles. You will finish this book with a deep understanding of the test-driven development methodology and the confidence to apply it to application programming with Java. What you will learn

- Explore the tools and frameworks required for effective TDD development
- Perform the Red-Green-Refactor process efficiently, the pillar around which all other TDD procedures are based
- Master effective unit testing in isolation from the rest of your code
- Design simple and easily maintainable code by implementing different techniques
- Use mocking frameworks and techniques to easily write and quickly execute tests
- Develop an application to implement behavior-driven development in conjunction with unit testing
- Enable and disable features using feature toggles

Who this book is for If you're an experienced Java developer and want to implement more effective methods of programming systems and applications, then this book is for you.

Getting the most out of Python to improve your codebase

Key Features

- Save maintenance costs by learning to fix your legacy codebase
- Learn the principles and techniques of refactoring
- Apply microservices to your legacy systems by implementing practical techniques

Book Description Python is currently used in many different areas such as software construction, systems administration, and data processing. In all of these areas, experienced professionals can find examples of inefficiency, problems, and other perils, as a result of bad code. After reading this book, readers will understand

Where To Download Beginning Solid Principles And Design Patterns For AspNet Developers

these problems, and more importantly, how to correct them. The book begins by describing the basic elements of writing clean code and how it plays an important role in Python programming. You will learn about writing efficient and readable code using the Python standard library and best practices for software design. You will learn to implement the SOLID principles in Python and use decorators to improve your code. The book delves more deeply into object oriented programming in Python and shows you how to use objects with descriptors and generators. It will also show you the design principles of software testing and how to resolve software problems by implementing design patterns in your code. In the final chapter we break down a monolithic application to a microservice one, starting from the code as the basis for a solid platform. By the end of the book, you will be proficient in applying industry approved coding practices to design clean, sustainable and readable Python code.

What you will learn

- Set up tools to effectively work in a development environment
- Explore how the magic methods of Python can help us write better code
- Examine the traits of Python to create advanced object-oriented design
- Understand removal of duplicated code using decorators and descriptors
- Effectively refactor code with the help of unit tests
- Learn to implement the SOLID principles in Python

Who this book is for

This book will appeal to team leads, software architects and senior software engineers who would like to work on their legacy systems to save cost and improve efficiency. A strong understanding of Programming is assumed.

Unleash the true power of JavaScript by mastering Object-Oriented programming principles and patterns

About This Book

Covering all the new Object-Oriented features

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

introduced in ES6, this book shows you how to build large-scale web apps Build apps that promote scalability, maintainability, and reusability Learn popular Object-Oriented programming (OOP) principles and design patterns to build robust apps Implement Object-Oriented concepts in a wide range of front-end architectures Who This Book Is For This book is ideal for you if you are a JavaScript developers who wants to gain expertise in OOP with JavaScript to improve your web development skills and build professional quality web applications. What You Will Learn Master JavaScript's OOP features, including the one's provided by ES6 specification Identify and apply the most common design patterns such as Singleton, Factory, Observer, Model-View-Controller, and Mediator Patterns Understand the SOLID principles and their benefits Use the acquired OOP knowledge to build robust and maintainable code Design applications using a modular architecture based on SOLID principles In Detail ECMAScript 6 introduces several new Object-Oriented features that drastically change the way developers structure their projects. Web developers now have some advanced OOP functionality at their disposal to build large-scale applications in JavaScript. With this book, we'll provide you with a comprehensive overview of OOP principles in JavaScript and how they can be implemented to build sophisticated web applications. Kicking off with a subtle refresher on objects, we'll show you how easy it is to define objects with the new ES6 classes. From there, we'll fly you through some essential OOP principles, forming a base for you to get hands-on with encapsulation. You'll get to work with the different methods of inheritance and we'll show you how to avoid using inheritance with Duck Typing. From there, we'll move on to some advanced patterns for object creation

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

and you'll get a strong idea of how to use interesting patterns to present data to users and to bind data. We'll use the famous promises to work with asynchronous processes and will give you some tips on how to organize your code effectively. You'll find out how to create robust code using SOLID principles and finally, we'll show you how to clearly define the goals of your application architecture to get better, smarter, and more effective coding. This book is your one-way ticket to becoming a JavaScript Jedi who can be counted on to deliver flexible and maintainable code. Style and approach This comprehensive guide on advanced OOP principles and patterns in JavaScript is packed with real-world use cases, and shows you how to implement advanced OOP features to build sophisticated web applications that promote scalability and reusability.

Implement robust applications by applying efficient Design Patterns with .NET 5 and C#

KEY FEATURES ? Detailed theoretical concepts covered, including the use of encapsulation, interfaces, and inheritance. ? Access to solutions applied for software strategy and final product output. ? Simplified demonstration of real applications implementing numerous design patterns.

DESCRIPTION This book covers detailed aspects of Design Patterns and Object-Oriented Programming concepts using the most modern version of the C# language and .NET platform, including many real-world examples and good practice guidelines that help developers in building robust and extensible applications. The book begins with the essential concepts of C# programming and the .NET platform. You get your foundation strong by understanding SOLID Principles and the actual implementation of reliable applications. You will be working on most common Design Patterns such as Abstract Factory, Adapter,

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Composite, Proxy, Command, Strategy, Observer, Factory Method, Singleton, Builder, Interpreter, Mediator, and many other patterns that will help you to create solid enterprise applications. You will also witness the performance of these design patterns in a real software development environment with the help of practical examples. After learning the most common Design Patterns practiced in .NET enterprise applications, the reader will be able to understand and apply good practices of software development based on the object-oriented paradigm to develop complex enterprise applications efficiently and simply. WHAT YOU WILL LEARN ? Fine-tune your knowledge about interfaces, polymorphism, and encapsulation. ? Learn to practice implementing design patterns in enterprise applications. ? Implement rich design patterns: Observer, Strategy, Command, Proxy, and more. ? Get to learn the latest additional design patterns such as Builder, Bridge, and Decorator. ? Includes illustrations, examples, and real use-cases of .NET 5.0 applications. WHO THIS BOOK IS FOR This book is for .NET developers, application developers, and software engineers who want to develop .NET applications with proven techniques and build error-free applications. This book also attracts fresh graduates and entry-level developers as long as basic knowledge about .NET is known to them. TABLE OF CONTENTS 1. C# Fundamentals 2. Introduction to .NET 5 3. Basic Concepts of Object-Oriented Programming 4. Interfaces in C# 5. Encapsulation and Polymorphism in C# 6. SOLID Principles in C# 7. Abstract Factory 8. Abstract Factory 9. Prototype 10. Factory Method 11. Adapter 12. Composite 13. Proxy 14. Command 15. Strategy 16. Observer 17. Good Practices and Additional Design Patterns

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

Design Patterns in Ruby (Adobe Reader)

More than 70 solutions to common Hibernate problems

The Object-Oriented Thought Process

Design Patterns in .NET

Agile coding with design patterns and SOLID principles

Adaptive Code

Looks at the principles and clean code, includes case studies showcasing the practices of writing clean code, and contains a list of heuristics and "smells" accumulated from the process of writing clean code.

Exploit the features of TypeScript to develop and maintain captivating web applications with ease About This Book Learn how to develop modular, scalable, maintainable, and adaptable web applications by taking advantage of TypeScript Create object-oriented JavaScript that adheres to the solid principles efficiently A comprehensive guide that explains the fundamentals of TypeScript with the help of practical examples Who This Book Is For If you are a JavaScript developer aiming to learn TypeScript to build beautiful web applications, then this book is for you. No prior knowledge of TypeScript is required. What You Will Learn Learn the key TypeScript language features and language runtime Develop modular, scalable, maintainable, and adaptable web applications Create object-oriented code that adheres to the solid

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

principles Save time using automation tools like Gulp and Karma
Develop robust applications with testing (Mocha, Chai and SinonJS) Put
your TypeScript skills in practice by developing a single-page web
application framework from scratch Use the JavaScript of tomorrow (ES6
and ES7) today with TypeScript In Detail TypeScript is an open source
and cross-platform typed superset of JavaScript that compiles to plain
JavaScript that runs in any browser or any host. It allows developers
to use the future versions of JavaScript (ECMAScript 6 and 7) today.
TypeScript adds optional static types, classes, and modules to
JavaScript, to enable great tooling and better structuring of large
JavaScript applications. This book is a step-by-step guide that will
get you started with TypeScript with the help of practical examples.
You start off by understanding the basics of TypeScript. Next,
automation tools like Grunt are explained followed by a detailed
description of function, generics, callbacks and promises. After this,
object-oriented features and the memory management functionality of
TypeScript are explained. At the end of this book, you will have
learned enough to implement all the concepts and build a single page
application from scratch. Style and approach This is a step-by-step
guide that covers the fundamentals of TypeScript with practical
examples. Each chapter introduces a set of TypeScript language
features and leads the readers toward the development of a real-world

Where To Download Beginning Solid Principles And Design Patterns For Aspnet Developers

application.

Apply design principles to your classes, preparing them for reuse. You will use package design principles to create packages that are just right in terms of cohesion and coupling, and are user- and maintainer-friendly at the same time. The first part of this book walks you through the five SOLID principles that will help you improve the design of your classes. The second part introduces you to the best practices of package design, and covers both package cohesion principles and package coupling principles. Cohesion principles show you which classes should be put together in a package, when to split packages, and if a combination of classes may be considered a "package" in the first place. Package coupling principles help you choose the right dependencies and prevent wrong directions in the dependency graph of your packages. What You'll LearnApply the SOLID principles of class designDetermine if classes belong in the same packageKnow whether it is safe for packages to depend on each other Who This Book Is For Software developers with a broad range of experience in the field, who are looking for ways to reuse,share, and distribute their code