

Serialization

PLINQ

SOA

Design Patterns

WCF

Web Services

WPF

NET 4.0

MVC

Reflection

LINQ



Microsoft

Visual Studio
2010

Advanced

Contents for .Net Professionals

Learn new and stay updated

Design Patterns, OOPS Principles, WCF, WPF, MVC & LINQ

MIRACLE
TECHNOLOGIES

Ist Floor, B-12, Behind DELITE
Showroom, Adjacent to Metro Station,
Near Gol Chakkar Sector-2, Noida

Ph. : +91-120-3058446
+91-120-3058447
Mobile : 9311305845/6/7

Mail ID :
mcts@miracleindia.com
www.miracleindia.com

Design Patterns

Design patterns is a reusable, high quality solution to a given requirement, task or recurring problem. Further it does not comprise of a complete solution that may be instantly converted to a code component, rather it provides a framework for how to solve a problem. Because design patterns consist of proven reusable architectural concepts, they are reliable and they speed up software development process.

Types of Design patterns

1) Creational Patterns

Creational patterns are ones that create objects for you, rather than having you instantiate objects directly. This gives your program more flexibility in deciding which objects need to be created for a given case.

- Ø Factory
- Ø Abstract Factory
- Ø Singleton
- Ø Prototype
- Ø Builder

2) Structural Patterns

These concern class and object composition. They use inheritance to compose interfaces and define ways to compose objects to obtain new functionality.

- Ø Façade
- Ø Bridge
- Ø Proxy
- Ø Adaptor

3) Behavioural Patterns

Most of these design patterns are specifically concerned with communication between objects

- Ø Observer
- Ø Command
- Ø Iterator
- Ø Mediator

SOLID Principle

Initial	Stands for	Concept
S	SRP	Single responsibility principle the notion that an object should have only a single responsibility.
O	OCP	Open/closed principle the notion that "software entities ... should be open for extension, but closed for modification".
L	LSP	Liskov substitution principle
I	ISP	Interface segregation principle "many client specific interfaces are better than one general purpose interface.
D	DIP	Dependency inversion principle one should "Depend upon Abstractions. Do not depend upon concretions

Serialization

- What is Serialization
- Real time Use of Serialization
- Types of Serialization
 - XML
 - Binary
 - SOAP
- Implement ISerializable interface
- Understand different attributes

Reflection

- Real time use of Reflection
- Create and understand custom Attributes
- Read and use assemblies at run time
- Road Map
 - The System.Reflection Namespace
 - The System.Type Class
 - Using System.Object.GetType()
 - Using System.Type.GetType()
 - Using the typeof () C# Operator
- Type Properties
- Type Methods
- Reflecting on Methods
- Reflecting on Fields and Properties
- Reflecting on Implemented Interfaces
- Reflecting on Method Parameters and Return Values
- Reflecting on Constructor
- Assembly Class
- Dynamically Loading an Assembly
- Late Binding
- Reflection Emit

Course Outline

WCF

Module 1: Getting Started with Windows Communication Foundation

This module explains how to build a simple WCF service and client.

Lessons

- Designing an Application to Be Part of a Service Oriented Architecture
- Overview of WCF Architecture
- Using a Language-Level Interface As a Service Contract
- Implementing a Simple WCF Service in Visual Studio 2010
- Consuming a simple WCF service in Visual Studio 2010

Lab : Creating a Simple Service

- Creating a Simple WCF Service
- Calling the Simple WCF Service

After completing this module, students will be able to:

- Explain how to design an application as part of a Service Oriented Architecture (SOA).
- Describe the main parts of the WCF architecture.
- Create a simple service contract for a WCF service.
- Implement a simple WCF service in Visual Studio 2010.
- Consume a simple WCF service in Visual Studio 2010.

Module 2: Configuring and Hosting WCF Services

This module explains how to create and configure a WCF service as a managed application and select an appropriate hosting option.

Lessons

- Programmatically Configuring a Managed Application to Host a WCF Service
- Programmatically Configuring a Managed Application to Call a WCF Service
- Selecting a Hosting Option for a WCF Service
- Deploying a WCF Service

Lab : Configure and Host a WCF Service

- Creating a Programmatically Configured Managed Application to Host a Service
- Calling a Service Hosted in a Managed Application by Using Programmatic Configuration
- Defining Service Settings by Using External Configuration
- Employing Different Hosting Options for a Service

After completing this module, students will be able to:

- Create a programmatically-configured managed application that hosts a WCF service.
- Call a WCF service hosted in a managed application by using programmatic configuration.
- Define WCF service settings by using external configuration.

- Select the best hosting option for a WCF service.
- Deploy a WCF service onto a remote host.

Module 3: Endpoints and Behaviors

This module explains how to expose a WCF service over different endpoints and add run-time functionality by using behaviors.

Lessons

- Exposing WCF Services Over Different Endpoints
- Adding Behaviors to Services and Endpoints
- Interoperating with Non-WCF Web services

Lab : Changing Service Endpoints and Behaviors

- Exposing Services by Using Different Bindings
- Adding Metadata Exchange to a Service
- Creating WCF Clients and Services That Interoperate with Non-WCF Web Services

After completing this module, students will be able to:

- Expose WCF services by using different bindings.
- Add behaviors to services and endpoints.
- Create WCF clients and services that interoperate with different types of Web services.

Module 4: Debugging and Diagnostics

This module explains how to improve debugging capabilities by examining messages and service activity.

Lessons

- Logging Messages
- Activity Tracing

Lab : Message Logging and Activity

- Generating Logging Information for a Service
- Enabling End-to-End Tracing for a Service

After completing this module, students will be able to:

- Log WCF messages.
- Trace WCF service activity.

Module 5: Designing and Defining Contracts

This module explains how to define service, operation, and data contracts to meet application requirements.

Lessons

- Designing a Coherent and Cohesive WCF Service Interface
- Defining a Service Contract
- Defining Operations on a Service
- Defining a Data Contract
- Defining a Message Contract
- Defining a Fault Contract

Lab : Contracts for Services and Data

- Defining and Implementing a One-Way Operation Contract
- Passing Complex Data with a Data Contract
- Defining and Implementing a Callback Contract

After completing this module, students will be able to:

- Design a coherent and cohesive service contract.
- Define a service contract.
- Define operations on a service.
- Define a data contract.

Module 6: Handling Errors

This module explains how to add error handling to a WCF application.

Lessons

- Relating .NET Exceptions to Service-Level Faults
- Using Faults in a Service
- Handling Faults and Exceptions on Clients

Lab : Error Handling

- Handling Unexpected Errors in a WCF Service
- Add Fault Handling to a WCF Service and the Service Contract

After completing this module, students will be able to:

- Explain how .NET exceptions relate to service-level faults.
- Define fault information in a service contract.
- Handle service exceptions on clients.

Module 7: Improving WCF Service Quality

This module explains how to address service quality issues such as performance, availability, concurrency, and instance management.

Lessons

- Managing WCF Service Instances
- Managing Concurrency Issues
- Improving WCF Service Quality

Lab : Improving WCF Service Quality

- Managing WCF Service Instances
- Managing Concurrency Issues
- Throttling Access to a WCF Service
- Passing Bulk Data Between a WCF Client and Service

After completing this module, students will be able to:

- Manage WCF service instances.
- Manage concurrency issues.
- Improve WCF service performance.

Module 8: Implementing WCF Security

This module explains how to implement security in a WCF application.

Lessons

- Overview of Security in WCF
- Applying Overall Security Requirements to a Binding
- Specifying Required Client and Service Credentials
- Working With Security Information

Lab : Protecting a Service

- Applying Security for Internal Network Communication
- Applying Security for Internet Communication

After completing this module, students will be able to:

- Explain the process for implementing security in WCF.
- Apply overall security requirements to a binding.
- Specify required client and service credentials.
- Work with security information.

Module 9: Implementing Transactions

This module explains how to protect data integrity through correct use of transactions.

Lessons

- Overview of Transactions in a Service-Oriented Application
- Creating Transactional Service Operations
- Enabling the Flow of Transactions from Client to Service

Lab : Implementing Transactions for a Service

- Controlling the Flow of a Transaction from Client to Service
- Forcing a Transaction to Start When a Service Operation Is Called

After completing this module, students will be able to:

- Explain how transactions work in a service-oriented application.
- Create transactional service operations.
- Control transaction flow from client to service.

Module 10: Sessions and Instancing

This module explains how to protect data integrity through correct use of transactions.

Lessons

- Overview of Instancing Mode.
- Working with Instance.
- Use Sessions.

Lab : Implementing Instance and Session in WCF.

After completing this module, students will be able to:

- Explain how use instance mode in application.
- Create session object in WCF.

Module 11: Concurrency

This module explains how to Implement Concurrency in WCF.

Lessons

- Concurrency in WCF
- Synchronization in WCF

Lab : Implementing Concurrency.

Module 12: Security - Infrastructure and User Level

This module explains how to Implement Security in WCF.

Lessons

- Transport Level Security
- Message Level Security
- Authentication
- Client and Service Credentials
- Authorization

Lab : Implementing Transport and Message Security.

Course Outline

WPF

Module 1: Creating an Application by Using WPF

This module explains how to build a WPF application.

Lessons

- Overview of WPF
- Creating a Simple WPF Application
- Handling Events and Commands
- Navigating Between Pages

Lab : Creating a WPF Application

- Creating a Stand-Alone WPF Application
- Handling Events and Commands
- Navigating Between Pages
- Creating an XBAP Application

After completing this module, students will be able to:

- Explain WPF concepts and features.
- Create a simple WPF application.
- Handle events and commands.
- Navigate between pages in a WPF application.

Module 2: Building User Interfaces

This module explains how to build a user interface in a WPF application.

Lessons

- Defining Page Layout
- Building User Interfaces by Using Content Controls
- Building User Interfaces by Using Items Controls
- Hosting Windows Forms Controls

Lab : Building User Interfaces

- Defining Page Layout and Adding Content
- Enhancing the User Interface by Using Items Controls
- Integrating Windows Forms Controls

After completing this module, students will be able to:

- Define layout.
- Use content controls.
- Use items controls.
- Host Windows Forms controls.

Module 3: Customizing Appearance

This module explains how to customize the appearance of a WPF application.

Lessons

- Sharing Logical Resources in an Application
- Creating Consistent User Interfaces by Using Styles
- Changing the Appearance of Controls by Using Control Templates
- Enhancing User Interfaces by Using Triggers and Animations

Lab : Customizing the Appearance of a WPF Application

- Sharing Logical Resources in an Application
- Creating Consistent User Interfaces by Using Styles
- Changing the Appearance of Controls by Using Control Templates
- Enhancing the User Interface by Using Triggers and Animations

After completing this module, students will be able to:

- Share logical resources throughout an application.
- Create a consistent user interface appearance by using styles.

- Change the appearance of controls using templates.

Module 4: Data Binding

This module explains how to bind user interface controls to data sources.

Lessons

- Overview of Data Binding
- Creating a Data Binding
- Implementing Property Change Notification
- Converting Data
- Validating Data

Lab : Data Binding

- Creating Data Bindings
- Implementing Property Change Notification
- Converting Data
- Validating Data

After completing this module, students will be able to:

- Explain WPF data binding concepts and terminology.
- Create a binding between a data source and a control.
- Implement property change notifications.
- Convert data between the binding source and the binding target.
- Validate data entered by the user.

Module 5: Data Binding to Collections

This module explains how to bind user interface controls to collections.

Lessons

- Binding to Collections of Objects
- Presenting Data by Using Collection Views
- Presenting Data by Using Data Templates

Lab : Data Binding to Collections

- Presenting Data by Using Collection Views
- Presenting Data by Using Data Templates

After completing this module, students will be able to:

- Bind to a collection of objects.
- Sort, filter, and group collections by using collection views.
- Fine-tune data display by using data templates.

Module 6: Creating New Controls

This module explains how to create new controls in a WPF application.

Lessons

- Overview of Control Authoring
- Creating Controls

Lab : Creating New Controls

- Implementing a Custom Control

After completing this module, students will be able to:

- Explain scenarios and options for creating new controls.
- Create user controls.
- Create custom controls.

Module 7: Managing Documents

This module explains how to manage documents in a WPF application.

Lessons

- Creating and Viewing Flow Documents
- Creating and Viewing Fixed Documents

Lab : Managing Documents

- Creating and Displaying Flow Documents

After completing this module, students will be able to:

- Create and view flow documents.
- Create and view fixed documents.

Module 8: Graphics and Multimedia

This module explains how to add graphics and multimedia support to a WPF application.

Lessons

- Creating 2-D graphics
- Displaying Images
- Creating 3-D Graphics
- Manipulating the 3-D Environment
- Adding Multimedia

Lab : Graphics and Multimedia

- Displaying 2-D graphics
- Displaying Images
- Displaying 3-D Graphics
- Playing Video Clips

After completing this module, students will be able to:

- Display 2-D graphics.
- Add images in a WPF application.
- Display 3-D graphics.
- Use additional 3-D graphics capabilities.
- Add multimedia content to a WPF application.

Module 9: Event Handling and Deployment

This module explains how to handle events in WPF and Deploy WPF application.

Lessons

- Overview of Events.
- Event Handling
- Deployment
- Dependency Property

Lab : Graphics and Multimedia

- Use Events in WPF
- Use Dependency Properties.
- Deployment

Course Outline

ASP.NET MVC

Introduction
History of Web Forms
Design Goals
The MVC Design Pattern
Web Forms versus MVC
Creating an Application
Routing
Routing Demo

Controllers

Controller Demo
Conventions
Views and Typed Views
View Helpers
Updates
Action Filters
Testing
Summary

ASP.NET MVC Controllers

Introduction

The Key
Routes and Controllers
Request Processing
IController Demo
DefaultControllerFactory
Building a Custom Factory
Using a Container
Controller Execution
Actions
Selector Filters
Action Parameters
Action Filters
Custom Action Filter Demo
Action Results
Summary

ASP.NET MVC Views

Introduction
Views and Web Forms
HTTP Refresher
Conventions
Master Pages
View Content
Listing Movies
HTML Helpers
Movie Edit View
Custom Helpers
View Data
Partial Views
MVContrib
Paging Movies
MVC Futures

Validation

Security

ASP.NET MVC Models

Introduction
What is the Model?
The Right Model
Model Conventions
Building a Simple Model
Model Binders
Edit the Model
Using the Form Collection
FormCollection Demo
Including and Excluding Properties
Using Parameters
Model Parameters
Models and the Entity Framework
Using the Object Context
Display a list of Movies
Data Transfer Objects
Create View
Adding Search
Deletes
Edits
Custom Model Binder
Summary

Course Outline

LINQ

Language Extensions

Implicitly typed variables
Extension methods
Object initialization syntax

- Anonymous types
- Lambda expressions

Introduction to LINQ

- LINQ expressions
- Using via extension methods
- Filtering
- Sorting
- Aggregation
- Skip and Take operators
- Joins

Deferred Execution

- Benefits and drawbacks
- IEnumerable vs IQueryable
- Using across tiers

Data Projection

- Single result value
- Existing types
- Anonymous types
- Grouping

LINQ to XML

- New XML classes
- Generating XML
- Querying XML
- Using data projection
- Combining with XPath

LINQ to SQL

- Attributes and mapping
- Creating a DataContext
- Deferred loading
- Saving changes
- Inserts and deletes
- Transactions
- Concurrency
- Handling exceptions

LINQ to Entities

- ADO.NET Entity Framework
- Defining an Entity Data Model (EDM)
- Database-first vs. Model-first
- Object Services
- Change tracking
- Using EntityClient
- Using Stored Procedures
- Plain-Old CLR Object support (POCO) [.NET 4.0 only]
- N-tier and service-based applications

Parallel LINQ (PLINQ) [.NET 4.0 only]

- Introduction to the Task Parallel Library (TPL)
- Using PLINQ with in-memory objects
- PLINQ and XML