

INEW 1072 Course Syllabus

1. Name of Course: Core Foundations of Microsoft .NET 2.0 Development

2. Number of Clock Hours: 24

3. Course Description:

This course provides students with the knowledge and skills to program Microsoft .NET Framework applications by using Microsoft Visual Studio 2005 development system. This course helps students to prepare for Exam 70-536, which is a core requirement for the MCTS certification. MOC 2956. Prerequisites: Visual Basic or C# programming experience.

4. Course Objectives

After completing this course, students will be able to:

- Develop applications that use types and standard contracts.
- Manage common data by using collections.
- Deploy and configure assemblies.
- Monitor and debug applications.
- Read and write files.
- Serialize data.

5. Rationale:

Upon completion of this course, students will have a better understanding of Microsoft .NET, the most popular framework in use today.

6. Required Materials:

Microsoft Official Curriculum, MOC 2856, included.

7. Evaluation

Those who participate in class discussions, complete course lab work, and miss no more than three class meetings will be awarded 2.4 continuing education units.

8. Course Outline

Module 1: Developing Applications by Using Types and Standard Contracts

This module describes the differences between reference types and value types. The module also describes how to create generic data types, implement standard .NET Framework interfaces, use delegates and events, and use attributes and exceptions.

Lessons

- Explaining Value Types and Reference Types
- Working with Generic Types
- Implementing .NET Standard Contracts
- Implementing Delegates and Events
- Working with Attributes and Exceptions

Lab: Developing Applications by Using Types and Standard Contracts

- Using Nullable Types
- Defining a Generic Type
- Implementing Standard .NET Framework Interfaces
- Throwing and Catching Exceptions (If Time Permits)
- Raising and Handling Events (If Time Permits)

After completing this module, students will be able to:

- Manage data in a .NET Framework application by using the .NET Framework 2.0 system types.
- Implement generic types.
- Implement .NET Framework interfaces to cause components to comply with standard contracts.
- Control interactions between .NET Framework application components by using delegates and events.

Module 2: Managing Common Data by Using Collections

This module introduces the object-based collection classes in the System.Collections namespace. The module also describes the generic collection classes in the System.Collections.Generic namespace and outlines the benefits of the generic collections. The module also covers several specialized collections in the System.Collections.Specialized namespace.

Lessons

- Working with Object-Based Collections
- Working with Generic Collections
- Working with Specialized Collections

Lab: Managing Common Data by Using Collections

- Using the Dictionary Generic Collection
- Using the List Generic Collection
- Using the NameValueCollection Specialized Collection

After completing this module, students will be able to:

- Use the object-based collections that are defined in the System.Collections namespace.
- Use the generic collections that are defined in the System.Collections.Generic namespace.
- Use the specialized collections that are defined in the System.Collections.Specialized namespace.

Module 3: Deploying and Configuring Assemblies

This module describes the key features of assembly configuration and installation. It also explains how to install assemblies and how to configure assemblies and the .NET Framework.

Lessons

- Installing and Configuring Assemblies
- Installing Assemblies by Using the Installer
- Configuring Assemblies
- Configuring the .NET Framework

Lab 3: Deploying and Configuring Assemblies

- Managing the Configuration Settings of an Assembly
- Deploying an Application by Using Windows Installer

After completing this module, students will be able to:

- Describe the purpose of deployment and configuration.
- Create a custom Microsoft Windows Installer for the .NET Framework components by using the System.Configuration.Install namespace.
- Embed configuration management functionality in a .NET Framework application.
- Configure the .NET Framework and applications by using configuration files, environment variables, and the .NET Framework Configuration tool.

Module 4: Monitoring and Debugging Applications

This module describes how to use the available classes of the System.Diagnostics namespace to monitor and debug a .NET Framework application.

Lessons

- Working with Application Processes
- Managing Application Performance
- Reading and Writing to an Event Log
- Debugging and Tracing Applications

Lab: Monitoring and Debugging Applications

- Monitoring Application Performance
- Logging Information in a Custom Event Log
- Adding and Configuring Tracing Statements in an Application

After completing this module, students will be able to:

- Manage system processes by using the Process class of the System.Diagnostics namespace.
- Monitor the performance of a .NET Framework application by using the diagnostics functionality of the .NET Framework 2.0.
- Manage the Windows Event Log by using the System.Diagnostics namespace.
- Debug and trace a .NET Framework application by using the System.Diagnostics namespace.

Module 5: Reading and Writing Files

This module describes how to use the classes in the .NET Framework 2.0 Class Library to manipulate files and folders and how to read and write data to and from files and memory.

Lessons

- Managing the File System
- Reading and Writing Data by Using Streams
- Compressing and Protecting Data by Using Streams
- Improving Application Security by Using Isolated Storage

Lab: Reading and Writing to File and Folders

- Archiving Files
- Compressing Files
- Storing and Retrieving User Preferences

After completing this module, students will be able to:

- Access files and folders by using the .NET Framework file system classes.
- Use streams to read and write data held in files and in memory.
- Compress, decompress, encrypt, and decrypt data as it is streamed.
- Create and manage isolated data stores for an application.

Module 6: Serializing Data

This module describes how the .NET Framework Class Library implements serialization and explains how to use the classes provided to serialize and deserialize classes and structures. This module also describes how to customize the serialization mechanism implemented by the .NET Framework.

Lessons

- Serializing and Deserializing Objects by Using Runtime Serialization
- Customizing the Runtime Serialization and Deserialization Processes
- Serializing and Deserializing Objects As XML Data

Lab 6: Serializing Data

- Serializing and Deserializing Data Across a Network by Using Runtime Serialization
- Customizing the Runtime Serialization Process
- Serializing and Deserializing Data as XML

After completing this module, students will be able to:

- Serialize and deserialize an object graph to a stream by using runtime serialization techniques.
- Customize the runtime serialization and deserialization processes.
- Serialize and deserialize an object graph into XML format data.