msdn training

Introduction

Contents

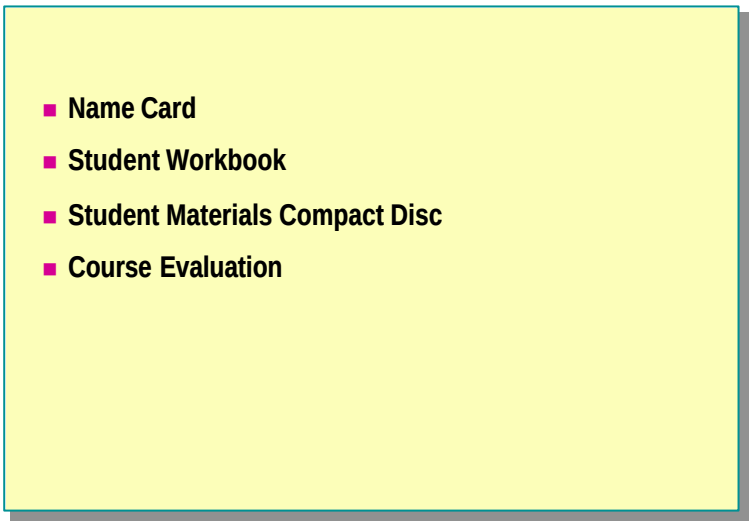
This course is based on the prerelease Beta 1 version of Microsoft® Visual Studio .NET. Content in the final release of the course may be different from the content included in this prerelease version. All labs in the course are to be completed with the Beta 1 version of Visual Studio .NET.

Microsoft

Introduction

- Name
- Company Affiliation
- Title/Function
- Job Responsibility
- Programming Experience
- C, C++, Visual Basic, or Java Experience
- Expectations for the Course

Course Materials



The following materials are included with your kit:

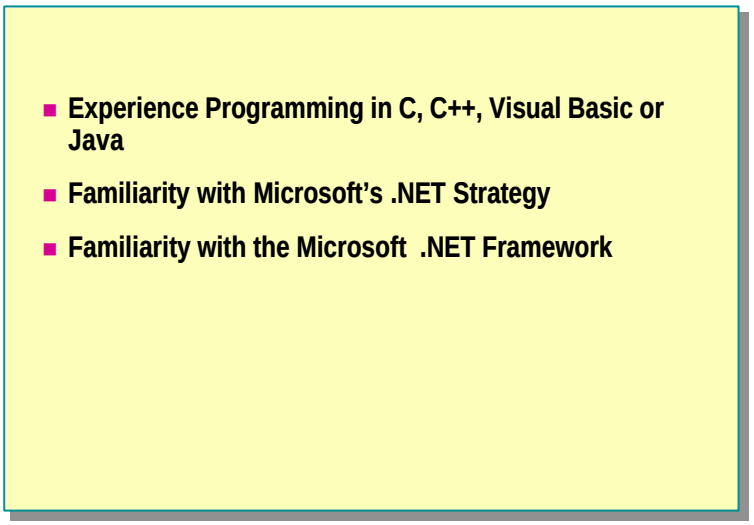
- *Name card.* Write your name on both sides of the name card.
- *Student workbook.* The student workbook contains the material covered in class, in addition to the hands-on lab exercises.
- *Student Materials compact disc.* The Student Materials compact disc contains the Web page that provides you with links to resources pertaining to this course, including additional readings, review and lab answers, lab files, multimedia presentations, and course-related Web sites.

  Note To open the Web page, insert the Student Materials compact disc into the CD-ROM drive, and then in the root directory of the compact disc, double-click **Autorun.exe** or **Default.htm**.

- *Course evaluation.* At the conclusion of this course, please complete the course evaluation to provide feedback on the instructor, course, and software product. Your comments will help us improve future courses.

  To provide additional comments or inquire about the Microsoft Certified Professional program, send e-mail to mcp@msprograms.com.

Prerequisites



This course requires that you meet the following prerequisites:

- Experience programming in C, C++, Microsoft Visual Basic®, Java, or another programming language
- Familiarity with Microsoft's .NET strategy as described on Microsoft's .NET Web site (http://www.microsoft.com/net/)
- Familiarity with the .NET Frameworks as described in MSDN® Magazine (http://msdn.microsoft.com/msdnmag/issues/0900/Framework/Framework.asp and http://msdn.microsoft.com/msdnmag/issues/1000/Framework2/Framework2.asp)

Course Outline

- Module 1: Overview of the Microsoft .NET Platform
- Module 2: Overview of C#
- Module 3: Using Value-Type Variables
- Module 4: Statements and Exceptions
- Module 5: Methods and Parameters

Module 1, “Overview of the Microsoft .NET Platform,” describes the rationale and features that provide the foundation for the .NET platform, including the .NET components. The purpose of this module is to build an understanding of the .NET platform for which you will be developing C# code. After completing this module, you will be able to describe the components of the .NET platform.

Module 2, “Overview of C#,” describes the basic structure of a C# application. This module provides a simple working example for you to analyze to learn how to use the **Console** class to perform some basic input and output operations and to learn best practices for handling errors and documenting your code. After completing this module, you will be able to compile, run, and debug a C# application.

Module 3, “ Using Value-Type Variables,” describes how to use value-type variables in C#. This module explains how to specify the type of data that variables will hold, how to name variables according to standard naming conventions, how to assign values to variables, and how to convert existing variables from one data type to another. After completing this module, you will be able to use value-type variables in C#.

Module 4, “Statements and Exceptions,” explains how to use some common statements in C#. This module also describes how to implement exception handling in C#. After completing this module, you will be able to throw and catch errors.

Module 5, “Methods and Parameters,” describes how to create static methods that take parameters and return values, how to pass parameters to methods in different ways, and how to declare and use overloaded methods. After completing this module, you will be able to use methods and parameters.

Course Outline *(continued)*

- **Module 6: Arrays**
- **Module 7: Essentials of Object-Oriented Programming**
- **Module 8: Using Reference-Type Variables**
- **Module 9: Creating and Destroying Objects**
- **Module 10: Inheritance in C#**

Module 6, " Arrays," explains how to group data into arrays. After completing this module, you will be able to create, initialize, and use arrays.

Module 7, "Essentials of Object-Oriented Programming," explains the terminology and concepts required to create and use classes in C#. This module also explains abstraction, encapsulation, inheritance, and polymorphism. After completing this module, you will be able to explain some of the common concepts of object-oriented programming.

Module 8, " Using Reference-Type Variables," describes how to use reference-type variables in C#. This module explains a number of reference types, such as string, that are built into the C# language and the Common Language Runtime. After completing this module, you will be able to use reference-type variables in C#.

Module 9, "Creating and Destroying Objects," explains what happens in the language runtime when an object is created and how to use constructors to initialize objects. This module also explains what happens when an object is destroyed and how the garbage collector reclaims memory. After completing this module, you will be able to create and destroy objects in C#.

Module 10, "Inheritance in C#," explains how to derive a class from a base class. This module also explains how to implement methods in a derived class by defining them as virtual methods in the base class and overriding or hiding them in the derived class, as required. This module explains how to seal a class so that it cannot be derived from and how to implement interfaces and abstract classes. After completing this module, you will be able to use inheritance in C# to derive classes and to define virtual methods.

Course Outline *(continued)*

- **Module 11: Aggregation, Namespaces, and Advanced Scope**
- **Module 12: Operators, Delegates, and Events**
- **Module 13: Properties and Indexers**
- **Module 14: Attributes**

Module 11, " Aggregation, Namespaces, and Advanced Scope," describes how to group classes together into larger, higher-level classes and how to use namespaces to group classes together inside named spaces and to create logical program structures beyond individual classes. This module also explains how to use assemblies to group collaborating source files together into a reusable, versionable, and deployable unit. After completing this module, you will be able to make code accessible at the component or assembly level.

Module 12, " Operators, Delegates, and Events," explains how to define operators and how to use delegates to decouple a method call from a method implementation. It also explains how to add event specifications to a class. After completing this module, you will be able to implement operators, delegates, and events.

Module 13, "Properties and Indexers," explains how to create properties to encapsulate data within a class and how to define indexers to gain access to classes by using array-like notation. After completing this module, you will be able to use properties to enable field-like access and indexers to enable array-like access.

Module 14, " Attributes," describes the purpose of attributes and the role they play in C# applications. This module explains attribute syntax and how to use some predefined attributes in the .NET environment. After completing this module, you will be able to create custom user-defined attributes and use these custom attributes to query attribute information at run time.

Note The information in this course is based on the Beta 1 prerelease version of Microsoft Visual Studio.NET.

Microsoft Certified Professional Program

The Microsoft Certified Professional program includes the following certifications:

- Microsoft Certified Systems Engineer + Internet (MCSE + Internet)
- Microsoft Certified Systems Engineer (MCSE)
- Microsoft Certified Database Administrator (MCDBA)
- Microsoft Certified Solution Developer (MCSD)
- Microsoft Certified Professional + Site Building (MCP + Site Building)
- Microsoft Certified Professional + Internet (MCP + Internet)
- Microsoft Certified Professional (MCP)
- Microsoft Certified Trainer (MCT)

For More Information See the " Certification" section of the Web page provided on the compact disc or the Microsoft Training and Certification Web site at http://www.microsoft.com/trainingandservices/

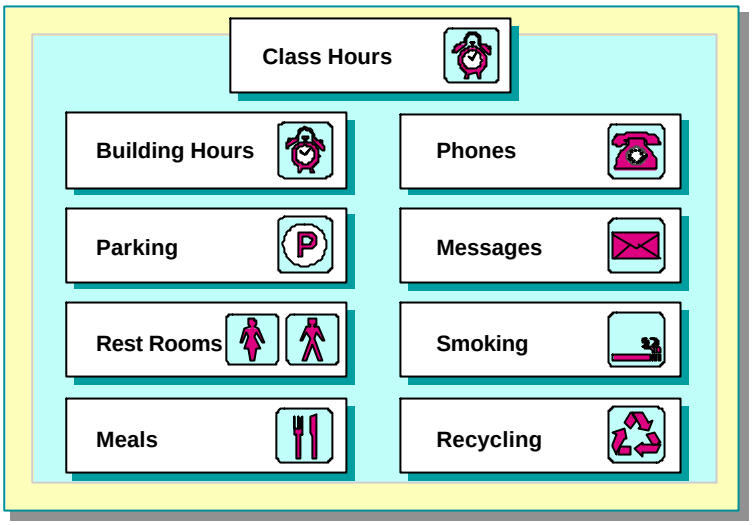
You can also send e-mail to mcp@msprograms.com if you have specific certification questions.

Exam Preparation Guides

To help prepare for the MCP exams, you can use the preparation guides that are available for each exam. Each Exam Preparation Guide contains exam-specific information, such as a list of the topics on which you will be tested. These guides are available on the Microsoft Certified Professional Web site at http://www.microsoft.com/trainingandservices/

Important MSDN Training curriculum helps you to prepare for Microsoft Certified Professional (MCP) exams. However, no one-to-one correlation exists between MSDN Training courses and MCP exams. Passing MCP exams requires real-world experience with the products—MSDN Training courses help get you started.

Facilities


Class Hours
Building Hours
Phones
Parking
Messages
Rest Rooms
Smoking
Meals
Recycling