Developing Data Access Solutions with Microsoft Visual Studio 2010

Code:	10265
Length:	5 days
URL:	View Online

In this course the student will learn to develop and optimize data access solutions using Microsoft .NET Framework 4 technologies. This course also helps prepare the student for Exam 70-516.

Skills Gained

After completing this course, students will be able to:

- Evaluate a variety of business cases, and then select an appropriate combination of data access technologies and tools most appropriate to each case. Describe the roles of Entity Framework, WCF Data Services, and ADO.NET for building and maintaining applications. UseLINQ on top of these technologies to improve productivity and the quality of their applications.

- Use the tools provided with the Entity Framework to map the conceptual model used by the business logic of an application to the logical data model provided by a database.

- Query an Entity Data Model (EDM) by using common methods such as LINQ to Entities, Entity SQL, and the classes in the EntityClient namespace.

- Perform data modification tasks on data in an EDM.

- Explainthe function of the Object Services model implemented by the Entity Framework and the support provided by the Object Services API for addressing the issues faced by enterprise applications that have to handle multiple concurrent users simultaneously accessing the same data .

- Describe best practices for designing and building a scalable, optimized data access layer by using Object Services.

- Customize and extend entities with their own business logic and use advanced mappings to shape the data model to their business and application requirements.

- Reuse existing business classes in a data access layer built by using the Entity Framework.

- Address the architectural issues that can arise when building an n-tier enterprise application by using the Entity Framework.

- Build extensible solutions that can update data in an n-tier enterprise application by using the Entity Framework.

- Access offline data or data that that has limited availability in client applications.

- Design, develop, and consume a simple WCF Data Service.

- Use WCF Data Services to update and delete data and to handle multi-user concerns.

- Develop high performance, scalable ADO.NET applications that can query and update data.

- Explainhow LINQ to SQLenables development against a logical model which abstracts the low-level details of querying ADO.NET tables and result sets.

Who Can Benefit

This course is intendedfor professional .NET software developers who use Microsoft Visual Studio in a team-based, medium-sized to large development environment.They willhave experience implementing data access and data binding within their Web and/or Windows client applications and are interested in learning to optimize data access code in their applications by using the Entity Framework, LINQ, and ADO.NET. Members of the audience are experienced users of Microsoft Visual Studio 2008 SP1 or newer releases of the Visual Studio product. The audience has some experience using Visual Studio 2010 for either Windows client or Web application development.Typically, this audience has the following knowledge/experience:

- Experience developing n-tier applications that access various data sources

- Experience implementing data binding within their applications

- Some experience using LINQ and ADO.NET

- A conceptual understanding of the Entity Framework

Prerequisites

Before attending this course, students must have:

- An understanding of the problem-solving techniques that apply to software development, including the following principles of software development:

-
  - Modern software development models

  - Typical phases of a software development lifecycle

  - Concepts of event-driven programming

  - Concepts of object-oriented programming

  - Creating use-case diagrams

  - Designing and building a user interface

  - Developing a structured application

- A basic understanding of the following scripting techniques and some hands-on experience writing scripts:

-
  - Web scripting techniques

  - Macro scripting techniques

  - Windows scripting techniques

- A general understanding of the purpose, function, and features of following .NET Framework topics:

-
  - Common Language Runtime

  - .NET Framework class library

  - Common Type System

  - Component interoperation

- Cross-language interoperability
- Assemblies in the Common Language Runtime
- Application domains
- Runtime hosts supported by the .NET Framework

- Experience using Visual Studio 2008 in the following task areas:
  -
    - Declaring and initializing typed variables using the Camel case naming convention
    - Using arithmetic, relational, and logical operators in code statements
    - Using branching statements to control code execution
    - Using looping statements to iterate through collections or repeat steps until a specified condition is met
    - Creating classes and methods to establish the basic structure of an application
    - Using methods and events to implement the programming logic of an application
    - Identifying syntax and logic errors
    - Accessing and managing data from a data source

- Experience in object oriented design and development as follows:
  -
    - Creating and accessing classes and class properties
    - Creating and accessing methods and overloaded methods
    - Implementing inheritance, base classes, and abstract classes
    - Declaring, raising, and handling events
    - Responding to and throwing exceptions
    - Implementing interfaces and polymorphism
    - Implementing shared and static members
    - Implementing generics
    - Creating components and class libraries

- Experience in N-Tier application design and development as follows:
  -
    - Managing a software development process
    - Controlling input at the user interface level in Windows client and Web applications
    - Debugging, tracing, and profiling .NET applications
    - Monitoring and logging .NET applications
    - Implementing basic testing best practices
    - Performing basic data access tasks with LINQ
      -
        - Basics of LINQ to XML
        - Basics of LINQ to Entities
        - Basics of LINQ to SQL
    - Implementing basic security best practices in .NET Applications
      -
        - Basics of Code Access Security
        - Basics of Role-Based Security

- Basics of Cryptography Services
- Implementing basic service calls
-
    - Basics of creating and consuming XML Web Services
    - Basics of creating and consuming WCF Services
- Using .NET Configuration Files
- Deploying .NET Framework Applications using ClickOnce and the MS Installer
- Data access experience in Windows client application development as follows:
-
    - Connect to a data source
    - Implement data binding
    - Implement data validation at the UI layer
- Data access experience in Web application development as follows:
-
    - Connect to a data source
    - Implement dynamic data
    - Implement data validation at the UI layer

Course Details

Outline

Module 1: Architecture and Data Access Technologies

This module describes the commonly used data access technologies and scenarios in which you are likely to use them.
Lessons

- Data Access Technologies
- Data Access Scenarios

Lab : Analyzing Data Access Scenarios

After completing this module, students will be able to:
- Describe the key data access technologies available to .NET Framework developers.
- Assign appropriate data access technologies to common data access scenarios.

Module 2: Building Entity Data Models

This module introduces the concepts of data modeling, and in particular, Entity Data Models (EDMs). It explains how you can use EDMs to decouple the conceptual data structure in your applications from the logical data structure in the data store.
Lessons
- Introduction to Entity Data Models
- Modifying the Entity Data Model
- Customizing the Entity Data Model

Lab : Using Entity Data Models

After completing this module, students will be able to:

- Describe and create an Entity Data Model.

- Modify an Entity Data Model by using the Entity Designer.

- Customize a model to meet their business requirements.

Module 3: Querying Entity Data

This module explains how toquery an entity data model by using common methods such as LINQ to Entities, Entity SQL, and the classes in the EntityClient namespace.
Lessons

- Retrieving Data by Using LINQ to Entities

- Retrieving Data by Using Entity SQL

- Retrieving Data by Using EntityClient Provider

- Retrieving Data by Using Stored Procedures

- Unit Testing YourData Access Code

Lab : Querying Entity Data

After completing this module, students will be able to:

- Retrieve data by using LINQ to Entities.

- Retrieve data by using the Entity SQL language.

- Retrieve data by using the EntityClient Provider.

- Retrieve data by using stored procedures in the entity model.

- Create unit tests for their data access code.

Module 4: Creating, Updating, and Deleting Entity Data
This module introduces you to the ways that the Entity Framework enables you to modify the data in your database. You apply changes to the entities managed by the ObjectContext class. The ObjectContext class is responsible for tracking all changes to entities and then persisting these changes to the database on request.
Lessons

- Understanding Change Tracking in the Entity Framework

- Modifying Data in an Entity Data Model

Lab : Creating, Updating, and Deleting Entity Data

After completing this module, students will be able to:

- Describe how the Entity Framework implements change tracking.

- Describe how to modify data in the entity model, and persist the changes to the database.

Module 5: Handling Multi-User Scenarios by Using Object Services

This moduleintroduces the concurrency model that the Entity Framework uses to address the issues faced by applications that must support multiple users who access the same data simultaneously. It also describes how the Entity Framework can make use of transactions to ensure data integrity.
Lessons

- Handling Concurrency in the Entity Framework

- Transactional Support in the Entity Framework

Lab : Handling Multi-User Scenarios by Using Object Services

After completing this module, students will be able to:

- Describe the optimistic concurrency model that the Entity Framework uses.
- Manage transactions in applications that use the Entity Framework.

Module 6: Building Optimized Solutions by Using Object Services

This module explainsbest practices for designing and building a scalable, optimized data access layer by using Object Services. The module introduces several techniques that can beused to optimize the performance of queries that execute against the conceptual model.
Lessons

- The Stages of Query Execution
- Change Tracking and Object Materialization
- Using Compiled Queries
- Using Design-Time Generated Entity Framework Views
- Monitoring Performance
- Performing Asynchronous Data Modifications

Lab : Building Optimized Solutions by Using Object Services

After completing this module, students will be able to:

- Explain how the Entity Framework executes queries.
- Understand the impact of tracking and object materialization on query performance.
- Describe how to use compiled queries.
- Describe how to use design-time generated views.
- Describe how to monitor query performance.
- Describe how to perform asynchronous data modifications.

Module 7: Customizing Entities and Building Custom Entity Classes

This module describes how to customize and extend entities with your own business logic.
Lessons

- Overriding Generated Classes
- Using Templates to Customize Entities
- Creating and Using Custom Entity Classes

Lab : Customizing Entities and Building Custom Entity Classes

After completing this module, students will be able to:

- Use partial classes and methods to add business logic to generated code.
- Create and use templates to customize code generation.
- Modify existing business classes to take advantage of entity functionality.

Module 8: Using POCO Classes with the Entity Framework

This module introduces the ways in which you can define custom entity classes in your Entity Framework application. By default, Microsoft Visual Studio generates a set of entity classes for you from the Entity Data Model (EDM). Instead of these generated classes, you may want to use an existing set of "plain old" CLR objects (POCO) business classes in your application You can also extend the generated entity classes to

add custom business functionality to your entity objects.
Lessons

- Requirements for POCO Classes

- POCO Classes and Lazy Loading

- POCO Classes and Change Tracking

- Extending Entity Types

Lab : Using POCO Classes with the Entity Framework

After completing this module, students will be able to:

- List the requirements that their POCO classes must meet.

- Create POCO entities that support automatic lazy loading.

- Create POCO entities that support automatic change tracking.

- Describe the options for using interfaces and inheritance to create custom entity objects.

Module 9: Building an N-Tier Solution by Using the Entity Framework

This module explains how toaddress the architectural issues that can arise when building an N-Tier enterprise application by using the Entity Framework.
Lessons

- Designing an N-Tier Solution

- Defining Operations and Implementing Data Transport Structures

- Protecting Data and Operations

Lab : Building an N-Tier Solution by Using the Entity Framework

After completing this module, students will be able to:

- Describe the issues and strategies that are relevant to building n-tier applications.

- Understand the key components that they must create in order to implement an n-tier application.

- Describe how to protect operations and data in an n-tier application.

Module 10: Handling Updates in an N-Tier Solution by Using the Entity Framework

This module describes how you can handle data modifications in an n-tier solution. The module describes the different strategies for handling modifications that you should use for the different alternative formats for transporting data between tiers: data transfer objects (DTOs), self-tracking entities (STEs), and simple entities (SEs). The module also describes how to manage the exceptions that can occur during the data modification process.
Lessons

- Tracking Entities and Persisting Changes

- Managing Exceptions in an N-Tier Solution

Lab : Handling Updates in an N-Tier Solution by Using the Entity Framework

After completing this module, students will be able to:

- Describe strategies that they can adopt for tracking changes in the client application, and persisting those changes in the

  database.

- Describe how to trap and handle update and concurrency exceptions in the an n-tier solution.

Module 11: Building Occasionally Connected Solutions

This module describes how toaccess offline or occasionally connected data in client applications.
Lessons

- Offline Data Caching by Using XML

- Using the Sync Framework

Lab : Building Occasionally Connected Solutions

After completing this module, students will be able to:

- Cache data in local XML files by using LINQ to XML.

- Implement an occasionally connected application by using the Microsoft Sync Framework.

Module 12: Querying Data by Using WCF Data Services
Windows Communication Foundation (WCF)Data Services enable you to create highly flexible data services that can be used to provide access to data across the Internet or a corporate network. You can access these services by using REST-style URIs, and they can be easily consumed by a wide variety of applications. As WCF Data Services are build on top of standard Internet protocols such as HTTP and the Atom Publishing Protocol, they are an ideal choice for delivering data to AJAX applications and Rich Interactive Applications built using technologies such as Microsoft Silverlight.
Lessons

- Introduction toWCF Data Services

- Creating a WCF Data Service

- Consuming a WCF Data Service

- Protecting Data and Operations in a WCF Data Service

Lab : Creating and Using WCF Data Services

After completing this module, students will be able to:

- Describe the purpose and features of WCF Data Services.

- Expose data by using a WCF Data Service.

- Implement a client application that can consume a WCF Data Service.

- Grant and restrict access to resources exposed by a WCF Data Service.

Module 13: Updating Data by Using WCF Data Services
This module describes how to use WCF Data Services to create, update, and delete data. WCF Data Services use standard internet protocols such as HTTP and the Atom Publishing Protocol to enable update access to data across the Internet or a corporate network.
Lessons

- Creating, Updating, and Deleting Data in a WCF Data Service

- Preventing Unauthorized Updates and Improving Performance

- Using WCF Data Services with Nonrelational Data

Lab : Updating Data by Using WCF Data Services

After completing this module, students will be able to:

- Create, update, and delete entities by using a WCF Data Service.

- Control access to data modification functionality, and improve performance by batching commands together.

- Use WCF Data Services to access and modify nonrelational data.

Module 14: Using ADO.NET

ADO.NET is a highly flexible framework for building applications that require access to data held in a data source.This module introduces ADO.NET and explains how you can use it to develop scalable, high-performance, data-driven applications.
Lessons

- Retrieving and Modifying Data by Using ADO.NET Commands

- Retrieving and Modifying Data by Using DataSets

- Managing Transactions and Concurrency in Multiuser Scenarios

Lab : Using ADO.NET

After completing this module, students will be able to:

- Retrieve and update databy using ADO.NET commands and stored procedures.

- Retrieve and update databy using DataSet objects.

- Implement transactions and handle concurrency exceptions.

Module 15: Using LINQ to SQL

ADO.NET provides a mechanism that enables you to build applications that can query and maintain data that is held in a variety of sources in a database-agnostic manner. However, building applications by using ADO.NET requires that you are familiar with the Structured Query Language (SQL) language and features of the database management system that you are connecting to. Language-Integrated Query (LINQ) to SQL provides a higher-level abstraction for managing data that is held in a Microsoft SQL Server database, and is an ideal stepping stone for migrating ADO.NET applications toward the ADO.NET Entity Framework. This module introduces LINQ to SQL and explains how you can use it to abstract the low-level details of ADO.NET queries by developing against a logical data model.
Lessons

- Implementing a Logical Data Model by Using LINQ to SQL

- Managing Performance and Handling Concurrency

Lab : Using LINQ to SQL

After completing this module, students will be able to:

- Design a logical data model by using LINQ to SQL.

- Manage performance by using LINQ to SQL, and handle concurrency.

Schedule (as of 4)

Date   **Location**