

# 10265: Developing Data Access Solutions with Microsoft® Visual Studio® 2010

Duration: 5 Days

Method: Instructor-Led

## Course Description

In this course, experienced developers who know the basics of data access (CRUD) in Windows® client and Web application environments will learn to optimize their designs and develop better performing data access code by using the ADO.NET Entity Framework, LINQ, WCF Data Services, the Sync Framework, and ADO.NET.

## Target Audience

This course is intended for:

- Professional .NET software developers who use Microsoft® Visual Studio® in a team-based, medium-sized to large development environment.
  - They will have 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.
- 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.

## Prerequisites

To attend this course, participants 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



## Prerequisites, *Continued*

- *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



## Prerequisites, *Continued*

- *Experience in N-Tier application design and development as follows: (Continued)*
  - 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 Objectives

Upon successful completion of this course, participants 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. Use LINQ 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.
- Explain the 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 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.
- Explain how LINQ to SQL enables development against a logical model which abstracts the low-level details of querying ADO.NET tables and result sets.



## Course Content

### Module 1: Architecture and Data Access Technologies

- Data Access Technologies
- Data Access Scenarios

### Module 2: Building Entity Data Models

- Introduction to Entity Data Models
- Modifying the Entity Data Model
- Customizing the Entity Data Model

### Module 3: Querying Entity Data

- 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 Your Data Access Code

### Module 4: Creating, Updating, and Deleting Entity Data

- Understanding Change Tracking in the Entity Framework
- Modifying Data in an Entity Data Model

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

- Handling Concurrency in the Entity Framework
- Transactional Support in the Entity Framework

### Module 6: Building Optimized Solutions by Using Object Services

- 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

### Module 7: Customizing Entities and Building Custom Entity Classes

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

### Module 8: Using POCO Classes with the Entity Framework

- Requirements for POCO Classes
- POCO Classes and Lazy Loading
- POCO Classes and Change Tracking
- Extending Entity Types



## Course Content, *Continued*

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

- Designing an N-Tier Solution
- Defining Operations and Implementing Data Transport Structures
- Protecting Data and Operations

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

- Tracking Entities and Persisting Changes
- Managing Exceptions in an N-Tier Solution

### Module 11: Building Occasionally Connected Solutions

- Offline Data Caching by Using XML
- Using the Sync Framework

### Module 12: Querying Data by Using WCF Data Services

- Introduction to WCF Data Services
- Creating a WCF Data Service
- Consuming a WCF Data Service
- Protecting Data and Operations in a WCF Data Service

### Module 13: Updating Data by Using WCF Data Services

- Creating, Updating, and Deleting Data in a WCF Data Service
- Preventing Unauthorized Updates and Improving Performance
- Using WCF Data Services with Nonrelational Data

### Module 14: Using ADO.NET

- Retrieving and Modifying Data by Using ADO.NET Commands
- Retrieving and Modifying Data by Using DataSets
- Managing Transactions and Concurrency in Multiuser Scenarios

### Module 15: Using LINQ to SQL

- Implementing a Logical Data Model by Using LINQ to SQL
- Managing Performance and Handling Concurrency

## LABS INCLUDED

