ADO.NET Using C# (VS 2013)

This three day, intensive course teaches the essential elements of ADO.NET such that, at the end of the course, the programmer is able to utilize its tremendous database manipulation powers to build effective database applications. The course includes a major case study demonstrating the use of ADO.NET in a realistic setting. It is current to .NET 4.5.1, Visual Studio® 2013 and SQL Server® 2012.

Course Objectives:

- Understand the architecture and main classes of ADO.NET.
- Gain fluency in programming ADO.NET using C#.
- Gain a thorough understanding of the use of disconnected DataSets for building highly scalable applications.
- Acquire a working knowledge of the tight coupling of XML with ADO.NET.
- Learn how to use additional features in ADO.NET, including asynchronous operations, multiple active result sets, and bulk copy.
- Acquire a working knowledge of LINQ and the Entity Framework
- Implement a realistic case study that ties together many concepts of ADO.NET in a practical demonstration.

Audience: C# .NET developers needing to communicate with datasources.

Prerequisites: A basic knowledge of SQL and of programming the .NET Framework using C#. The student should also understand the fundamentals of XML. To get full benefit from the examples in the course the student should be able to write simple Windows Forms applications. A working knowledge of SQL Server is also desirable.

Number of Days: 3 days

1 Introduction to ADO.NET

Microsoft Data Access Technologies
ODBC
OLE DB
ActiveX Data Objects (ADO)
Accessing SQL Server before ADO.NET
ADO.NET
ADO.NET Architecture
.NET Data Providers
Programming with ADO.NET Interfaces
.NET Namespaces
Connected Data Access
SQL Express LocalDB
SqlLocalDB Utility
Visual Studio Server Explorer
Queries
SQL Server Management Studio
ADO.NET Class Libraries
Connecting to an OLE DB Data Provider
Using Commands
Creating a Command Object
ExecuteNonQuery
Using a Data Reader
Disconnected Datasets
Data Adapters
Buy Computer
Model
Component
Part
PartConfiguration
System
SystemId as Identity Column
SystemDetails
StatusCode
Relationships

Stored Procedure

2 ADO.NET Connections

ADO.NET Block Diagram
.NET Data Providers
Namespaces for .NET Data Providers
BasicConnect (version 1)
Using Interfaces
IDbConnection Properties
Connection String
SQL Server Connection String
OLE DB Connection String
SQL Server Security
IDbConnection Methods
BasicConnection (version 2)
Connection Life Cycle
BasicConnect (version 3)
Database Application Front-ends
Connection Pooling
Pool Settings for SQL Server
Connection Events
ADO.NET Exception Handling

3 ADO.NET Commands

Command Objects
Creating Commands
Executing Commands
Dynamic Queries
Parameterized Queries
Command Types
Stored Procedures
Testing the Stored Procedure
Stored Procedures in ADO.NET
Batch Queries
Transactions

4 DataReaders and Connected Access

DataReaders
Using a DataReader
Closing a DataReader
IDataRecord
Type-Safe Accessors
GetOrdinal()
Null Data
Testing for Null
ExecuteReader Options
Returning Multiple Result Sets
DataReader Multiple Results Sets
Obtaining Schema Information

5 DataSets and Disconnected Access

DataSet
DataSet Architecture
Why DataSet?
DataSet Components
DataAdapter
Data Access Class
Retrieving the Data
Filling a DataSet
Accessing a DataSet
Updating a DataSet Scenario
Adding a New Row
Searching and Updating a Row
Deleting a Row
Row Versions
Row State
BeginEdit and CancelEdit
DataTable Events
Updating a Database
Insert Command
Update Command
Delete Command
Exception Handling
Command Builders

6 More About DataSets

Filtering DataSets
Using a Single DataTable
Multiple Tables
DataSet Architecture
Schema in the DataSet
Relations
Navigating a DataSet
Using Parent/Child Relation
Inferring Schema
AddWithKey
Adding a Primary Key
TableMappings
Identity Columns
Creating a Dataset Manually
Manual DataSet Code

7 XML and ADO.NET

ADO.NET and XML
Rendering XML from a DataSet
XmlWriteMode
Reading XML into a DataSet
DataSets and XML Schema

ModelSchema.xsd
Reading XML Schema
XmlReadMode
Writing Data as Attributes
XML Data in DataTables
Typed DataSets
Table Adapter
Using a Typed DataSet
Synchronizing DataSets and XML
Using XmlDataDocument
Windows Client Code
Web Client Code
XML Serialization
Default Constructor

8 Concurrency and Transactions
DataSets and Concurrency
Handling Concurrency Violations
Pessimistic Concurrency
Transactions
Programming ADO.NET Transactions
ADO.NET Transaction Code
Using ADO.NET Transactions
DataBase Transactions
Transaction in Stored Procedure
Testing the Stored Procedure
SQL Server Error

9 Additional Features
AcmePub Database
Connected Database Access
Long Database Operations
Asynchronous Operations
Multiple Active Result Sets
Bulk Copy

10 LINQ and Entity Framework
Language Integrated Query (LINQ)
LINQ to ADO.NET
Bridging Objects and Data
Object Relational Designer
IntelliSense
Basic LINQ Query Operators
Obtaining a Data Source
Filtering
Ordering
Aggregation
Obtaining Lists and Arrays
Deferred Execution
Modifying a Data Source
Performing Inserts via LINQ to SQL
Performing Deletes via LINQ to SQL
Performing Updates via LINQ to SQL
LINQ to DataSet
Using the Typed DataSet
ADO.NET Entity Framework
Exploring the EDM
AcmePub Tables
AcmePub Entity Data Model
XML Representation of Model
Entity Data Model Concepts
Conceptual Model
Storage Model
Mappings
Querying the EDM
Class Diagram
Context Class
List of Categories
List of Books
Entity Framework in a Class Library
Data Access Class Library
Client Code

11 Appendix A – Acme Computer Case Study

12 Appendix B – SQL Server 2012 Express
SQL Server Express
SQL Server 2012 Express LocalDB
AttachDBFileName
Database
Moving from LocalDB to SQL Server

13 Appendix C – Learning Resources