

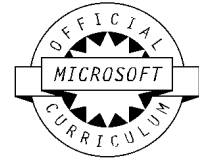
Course Outline

80304- Development II in Microsoft Dynamics AX 2012

Duration: 2 days (12 hours)

Target Audience:

The intended audience is experienced systems consultants typically working for a Microsoft Dynamics partner that is selling, consulting, implementing, and supporting Microsoft Dynamics AX 2012.



Prerequisites:

Before attending this course, students must have:

- Working experience with Microsoft Dynamics AX and some knowledge of technical features of Microsoft Dynamics AX 2012 architecture and development environment.
- Completed Course 80303A, Development I in Microsoft Dynamics AX 2012.

Topics Covered:

➤ Module 1: Introduction to X++

- Introduction
- Characteristics of X++
- Development Tools
- Reverse Engineering
- Best Practices
- Lab : Print to the ScreenLab : Debug the JobLab : Create a Data ModelLab : Create an XML Developer Document

After completing this module, students will be able to:

- Identify key features of developing with X++.
- Describe the basic foundation of object-oriented programming.
- Use the development tools available within Microsoft Dynamics AX 2012.
- Create object and data models from existing application elements by using the Reverse Engineering tool.
- Use best practices to instill good programming habits.

➤ Module 2: X++ Control Statements

- Introduction
- Introduction to Variables
- Operators
- Conditional Statements

- Loops
 - Built-in Functions
 - Communication Tools
 - Lab : Create a Times Table Using a While LoopLab : Create a Times Table Using a Do...while LoopLab : Create a Times Table Using a for StatementLab : Create a YesNo BoxLab : Create a Infolog TreeLab : Create a Dialog BoxLab : Use X++ Control Statements
- After completing this module, students will be able to:

- Declare and use extended data types for variables.
- Use the various operators available in X++.
- Control program flow using conditional statements in X++.
- Repetitively call the same blocks of code by using Loop statements.
- Use standard functions that are built in to the application
- Use output commands to display data and messages to the user.

➤ **Module 3: Classes and Objects**

- Introduction
 - Classes
 - Method Access Control
 - Inheritance
 - Objects
 - Scoping and Parameters in X++
 - Methods
 - Referencing Object Methods
 - Method Types
 - Table as Classes
 - Eventing
 - Lab : Create a New ClassLab : Allow Access to MethodsLab : Instantiating a ClassLab : Use Method ParametersLab : Create a Run MethodLab : Create a Calculator Class
- After completing this module, students will be able to:

- Use the classes within Microsoft Dynamics AX 2012 X++ development
- Control access to methods using Access Control Method Modifiers
- Extend a class using the concept of inheritance
- Describe the differences between an object and a class
- Initialize variables in the appropriate place according to scoping rules
- Call methods within the same class
- Use the different method types available
- Describe the similarities and differences between tables and classes
- Use the eventing publisher and subscriber model when modifying code in the application.

➤ Module 4: Accessing the Database

- Introduction
- Retrieving Data
- Data Manipulation
- Queries
- Lab : Retrieving DataLab : UpdateLab : Create Query Using X++

After completing this module, students will be able to:

- Retrieve data from the database using a select statement.
- Create, update and delete data in the database.
- Use and build queries using kernel classes.

➤ Module 5: Exception Handling

- Introduction
- Exceptions
- Try and Catch Exceptions
- Throwing Exceptions
- Optimistic Concurrency Exceptions
- Lab : Handle an Exception

After completing this module, students will be able to:

- Examine the exception handling mechanism in Microsoft Dynamics AX.
- Use the Try, Catch, and Retry statements.
- Throw an exception from code.
- Identify and create code used to handle optimistic concurrency exceptions.

➤ Module 6: Security for Developers

- Introduction
- Permissions
- Security Policies
- Code Access Security
- Display Method Authorization

After completing this module, students will be able to:

- Set permissions on application elements
- Design and create security policies
- Secure unsafe Application Programming Interfaces (APIs) using the Code Access Security framework
- Authenticate data returned from display methods