Development II in Microsoft Dynamics AX 2012

Varighed: 2 Days      Kursus Kode: M80304

Beskrivelse:
This two-day instructor-led course introduces students to the tools available in Microsoft Dynamics AX 2012 development environment.

Målgruppe:
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.

Agenda:
- After completing this course, students will be able to:
  - 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.
  - Retrieve data from the database using a select statement.
  - Create, update and delete data in the database.
  - Use and build queries using kernel classes.
  - 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.
  - 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.

- 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.
- 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.
- 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.
- 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.
- 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.
- 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.

Forudsætninger:

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.
Indhold:

Module 1: Introduction to X++

This module introduces the students to the characteristics of X++.

Lessons

- Introduction
- Characteristics of X++
- Development Tools
- Reverse Engineering
- Best Practices
- Introduction
- Classes
- Method Access Control
- Inheritance
- Objects
- Scoping and Parameters in X++
- Methods
- Referencing Object Methods
- Method Types
- Table as Classes
- Eventing
- Introduction
- Retrieving Data
- Data Manipulation
- Queries
- Introduction
- Exceptions
- Try and Catch Exceptions
- Throwing Exceptions
- Optimistic Concurrency Exceptions
- Introduction
- Permissions
- Security Policies
- Code Access Security
- Display Method Authorization

Lab: Create an Infolog Tree

Lab: Create a Dialog Box

Lab: Use X++ Control Statements

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

Module 3: Classes and Objects

This module discusses how to use classes and objects within Microsoft Dynamics AX 2012.

Lessons

- Introduction
- Characteristics of X++
- Development Tools
- Reverse Engineering
- Best Practices

Lab: Print to the Screen

Lab: Debug the Job

Lab: Create a Data Model

Lab: Create an XML Developer Document

Lab: Retrieving Data

Lab: Update

Lab: Create Query Using X++

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.
- 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
### Module 2: X++ Control Statements

This module explains how to use control statements in X++.

**Lessons**
- **Introduction**
- **Introduction to Variables**
- **Operators**
- **Conditional Statements**
- **Loops**
- **Built-in Functions**
- **Communication Tools**

#### Lab: Create a Times Table Using a While Loop

#### Lab: Create a Times Table Using a Do...while Loop

#### Lab: Create a Times Table Using a for Statement

#### Lab: Create a YesNo Box

---

### Module 5: Exception Handling

This module discusses the mechanisms built in Microsoft Dynamics AX 2012 to help with exception handling.

**Lessons**
- **Introduction**
- **Characteristics of X++**
- **Development Tools**
- **Reverse Engineering**
- **Best Practices**
- **Introduction**
- **Classes**
- **Method Access Control**
- **Inheritance**
- **Objects**
- **Scoping and Parameters in X++**
- **Methods**
- **Referencing Object Methods**
- **Method Types**
- **Table as Classes**
- **Eventing**
- **Introduction**
- **Retrieving Data**
- **Data Manipulation**
- **Queries**
- **Introduction**
- **Exceptions**
- **Try and Catch Exceptions**
- **Throwing Exceptions**
- **Optimistic Concurrency Exceptions**
- **Introduction**
- **Permissions**
- **Security Policies**
- **Code Access Security**
- **Display Method Authorization**

#### Lab: Create a New Class

#### Lab: Allow Access to Methods

#### Lab: Instantiating a Class

#### Lab: Use Method Parameters

#### Lab: Create a Run Method

#### Lab: 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.
- Examine the exception handling

#### Lab: Handle an Exception

**After completing this module, students will be able to:**
- Use output commands to display data and messages to the user.
- Retrieve data from the database using a select statement.
- Create, update and delete data in the database.
- Use and build queries using kernel classes.
- 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.
Module 4: Accessing the Database
This module explains how to develop modifications that interact with the database.

- 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

This module introduces some more advanced security features of Microsoft Dynamics AX 2012.

Lessons
- Introduction
- Characteristics of X++
- Development Tools
- Reverse Engineering
- Best Practices
- Introduction
- Classes
- Method Access Control
- Inheritance
- Objects
- Scoping and Parameters in X++
- Methods
- Referencing Object Methods
- Method Types
- Table as Classes
- Eventing
- Introduction
- Retrieving Data
- Data Manipulation
- Queries
- Introduction
- Exceptions
- Try and Catch Exceptions
- Throwing Exceptions
- Optimistic Concurrency Exceptions
- Introduction
- Permissions
- Security Policies
- Code Access Security
- Display Method Authorization

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

Flere Informationer:
For yderligere informationer eller booking af kursus, kontakt os på tlf.nr.: 44 88 18 00
training@globalknowledge.dk
www.globalknowledge.dk
Global Knowledge, Stamholmen 110, 2650 Hvidovre