

**ITShare Company**

01122339966-01122339955

Mostafa@iteshare.com

# Microsoft Web Applications

## Course 40410-A: JavaScript, HTML and CSS Web Development

### Module 1: Creating Web Pages using Hypertext Markup Language (HTML)

- Introduction to HTML
- Introduction to Document Structure and Layout
- How to work with Basic HTML Elements
- How to format HTML Elements
- Introduction to Hyperlinks
- How to Configure your Development Environment
- How to use various HTML Elements on a web page
- How to create a basic HTML document using the core HTML elements

### Module 2: Creating Sophisticated Document Structures using HTML

- How to Capture User Input
- Introduction to HTML5 Semantic Elements
- How to work with Scalable Vector Graphics (SVG)
- How to Implement Media Controls
- How to add additional HTML features to a web page
- How to use advanced features of HTML such as forms

### Module 3: Designing Web Pages Elements using Cascading Style Sheets (CSS)

- Introduction to CSS

- How to work with Style Definitions
- How to work with Style Inheritance
- How to work with Box Model
- How to work with Fonts
- How to Style a web page using CSS

#### **Module 4: Implementing Program Flow using JavaScript (JS)**

- Introduction to JavaScript
- Working with JavaScript Types
- Introduction to JavaScript Callbacks
- Advanced JavaScript
- How to work with HTML5 APIs
- How to implement an image carousel in HTML5
- How to use JavaScript to add a logical flow to a web page

#### **Module 5: Adding External Libraries to Enhance HTML Applications**

- Introduction to jQuery
- Introduction to AngularJS
- Introduction to Bootstrap
- Introduction to TypeScript
- Introduction to Sass
- How to modernize a web page
- Introduction to various external libraries that may enhance the HTML, CSS, and JavaScript content on a web page

## Course 20483 : Programming in C#

### Module 1: Review of Visual C# Syntax

- Overview of Writing Application by Using Visual C#
- Data Types, Operators, and Expressions
- Visual C# Programming Language Constructs
- Implementing Insert Functionality for the Students List
- Implementing Delete Functionality for the Students List
- Displaying a Student's Age

### Module 2: Creating Methods, Handling Exceptions, and Monitoring Applications

- Creating and Invoking Methods
- Creating Overloaded Methods and Using Optional and Output Parameters
- Handling Exceptions
- Monitoring Applications
- Refactoring the Enrolment Code
- Validating Student Information
- Saving Changes to the Class List

### Module 3: Basic types and constructs of Visual C#

- Implementing Structs and Enums
- Organizing Data into Collections
- Handling Events
- Adding Navigation Logic to the Grades Prototype Application
- Creating Data Types to Store User and Grade Information
- Displaying User and Grade Information

### Module 4: Creating Classes and Implementing Type-Safe Collections

- Creating Classes
- Defining and Implementing Interfaces
- Implementing Type-Safe Collections
- Implementing the Teacher, Student, and Grade Structs as Classes

- Adding Data Validation to the Grade Class
- Displaying Students in Name Order
- Enabling Teachers to Modify Class and Grade Data

## Module 5: Creating a Class Hierarchy by Using Inheritance

- Creating Class Hierarchies
- Extending .NET Framework Classes
- Refactoring Common Functionality into the User Class
- Implementing Password Complexity by Using an Abstract Method
- Creating the ClassFullException Custom Exception

## Module 6: Reading and Writing Local Data

- Reading and Writing Files
- Serializing and Deserializing Data
- Performing I/O by Using Streams
- Serializing Data for the Grades Report as XML
- Previewing the Grades Report
- Persisting the Serialized Grade Data to a File

## Module 7: Accessing a Database

- Creating and Using Entity Data Models
- Querying Data by Using LINQ
- Creating an Entity Data Model from The School of Fine Arts Database
- Updating Student and Grade Data by Using the Entity Framework
- Extending the Entity Data Model to Validate Data

## Module 8: Accessing Remote Data

- Accessing Data Across the Web
- Accessing Data by Using OData Connected Services
- Creating a WCF Data Service for the SchoolGrades Database
- Integrating the Data Service into the Application
- Retrieving Student Photographs Over the Web (If Time Permits)

## Module 9: Designing the User Interface for a Graphical Application

- Using XAML to Design a User Interface

- Binding Controls to Data
- Customizing the Appearance of Student Photographs
- Styling the Logon View
- Animating the StudentPhoto Control (If Time Permits)

## **Module 10: Improving Application Performance and Responsiveness**

- Implementing Multitasking
- Performing Operations Asynchronously
- Synchronizing Concurrent Access to Data
- Ensuring That the UI Remains Responsive When Retrieving Teacher Data
- Providing Visual Feedback During Long-Running Operations

## **Module 11: Integrating with Unmanaged Code**

- Creating and Using Dynamic Objects
- Managing the Lifetime of Objects and Controlling Unmanaged Resources
- Generating the Grades Report by Using Word
- Controlling the Lifetime of Word Objects by Implementing the Dispose Pattern

## **Module 12: Creating Reusable Types and Assemblies**

- Examining Object Metadata
- Creating and Using Custom Attributes
- Generating Managed Code
- Versioning, Signing, and Deploying Assemblies
- Creating and Applying the IncludeInReport attribute
- Updating the Report
- Storing the Grades.Utilities Assembly Centrally (If Time Permits)

## **Module 13: Encrypting and Decrypting Data**

- Implementing Symmetric Encryption
- Implementing Asymmetric Encryption
- Encrypting the Grades Report
- Encrypting the Grades Report

# Introduction to SQL Programming

## Executing a Simple Query

- Connect to the SQL Database using SQL Server Management Studio
- Query a Database using the SELECT Statement
- Save a Query Script to a File
- Modify and Execute a Saved Query

## Performing a Conditional Search

- Use the WHERE Clause to Search and Filter Data using One or More Conditions
- Use the WHERE Clause to Search for a Range of Values with BETWEEN and IN Clauses

## How to Handle NULL Values in Data

- Search Text Data Based on Patterns using the LIKE Operator

## Working with Functions

- Format and Manipulate Dates using Date Functions
- Summarize Data using Aggregate Functions Like COUNT, SUM, AVG, MIN, MAX
- Use Functions to Manipulate String Values

## Organizing Data

- Sort Data using the ORDER BY Clause
- Rank Data with the RANK and DENSE\_RANK Clauses (Additional Self-Study Topic)
- Group Data with the GROUP BY Clause
- Filter Grouped Data using the HAVING Clause
- Summarize Grouped Data using ROLLUP and CUBE Clauses (Additional Self-Study Topic)
- Use PIVOT and UNPIVOT Operators (Additional Self-Study Topic)

## Retrieving Data from Multiple Tables

- Combine the Results of Two Queries with the UNION Statement (Additional Self-Study Topic)
- Compare the Results of Two Queries with the INTERSECT and EXCEPT Statements (Additional Self-Study Topic)
- Query Data from Multiple Tables using the JOIN Clause

## Exporting Query Results

- Save the Query Results (Additional Self-Study Topic)
- Exporting an XML File (Additional Self-Study Topic)

## 20486-D: Developing ASP.NET Core MVC Web Applications

### Module 1: Exploring ASP.NET Core MVC

- Overview of Microsoft Web Technologies
- Overview of ASP.NET 4.x
- Introduction to ASP.NET Core MVC
- Lab : Exploring ASP.NET Core MVC
- Exploring a Razor Pages Application
- Exploring a Web API Application
- Exploring an MVC Application

### Module 2: Designing ASP.NET Core MVC Web Applications

- Planning in the Project Design Phase
- Designing Models, Controllers and Views
- Lab : Designing ASP.NET Core MVC Web Applications
- Planning Model Classes
- Planning Controllers
- Planning Views
- Architecting and MVC Web Application

### Module 3: Configure Middlewares and Services in ASP.NET Core

- Configuring Middlewares
- Configuring Services
- Lab : Configuring Middleware and Services in ASP.NET Core
- Working with Static Files
- Creating custom middleware
- Using dependency injection
- Injecting a service to a controller

### Module 4: Developing Controllers

- Writing Controllers and Actions
- Configuring Routes

- Writing Action Filters
- Lab : Developing Controllers
- Adding controllers and actions to an MVC application
- Configuring routes by using the routing table
- Configuring routes using attributes
- Adding an action filter

## Module 5: Developing Views

- Creating Views with Razor Syntax
- Using HTML Helpers and Tag Helpers
- Reusing Code in Views
- Lab : Developing Views
- Adding Views to an MVC Application
- Adding a partial view
- Adding a view component

## Module 6: Developing Models

- Creating MVC Models
- Working with Forms
- Validate MVC Application
- Lab : Developing Models
- Adding a model
- Working with Forms
- Add Validation

## Module 7: Using Entity Framework Core in ASP.NET Core

- Introduction to Entity Framework Core
- Working with Entity Framework Core
- Use Entity Framework Core to connect to Microsoft SQL Server
- Lab : Using Entity Framework Core in ASP.NET Core
- Adding Entity Framework Core
- Use Entity Framework Core to retrieve and store data
- Use Entity Framework Core to connect to Microsoft SQL Server

## Module 8: Using Layouts, CSS and JavaScript in ASP.NET Core MVC

- Using Layouts



- Using CSS and JavaScript
- Using jQuery
- Lab : Using Layouts, CSS and JavaScript in ASP.NET Core
- Applying a layout and link views to it
- Using CSS
- Using JavaScript
- Using jQuery

## Module 9: Client-Side Development

- Applying Styles
- Using Task Runners
- Responsive design
- Lab : Client-Side Development
- Use gulp to run tasks
- Styling using Sass
- Using Bootstrap

## Module 10: Testing and Troubleshooting

- Testing MVC Applications
- Implementing an Exception Handling Strategy
- Logging MVC Applications
- Testing a Model
- Testing a controller using a fake repository
- Implementing a repository in MVC project
- Add exception handling
- Add logging

## Module 11: Managing Security

- Authentication in ASP.NET Core
- Authorization in ASP.NET Core
- Defending from Attacks
- Lab : Managing Security
- Use Identity
- Add Authorization
- Avoid the Cross-Site Request Forgery Attack

## Module 12: Performance and Communication

- Implementing a Caching Strategy
- Managing State
- Two-way communication
- Lab: Implementing a Caching Strategy
- Lab: Managing state
- Lab: Two-Way communication

## Module 13: Implementing Web APIs

- Introducing Web APIs
- Developing a Web API
- Calling a Web API
- Lab: Adding Actions and Call Them Using Microsoft Edge
- Lab: Calling a Web API using server-side code
- Lab: Calling a Web API using jQuery

## Module 14: Hosting & Deployment

- On-premise hosting and deployment
- Deployment to Microsoft Azure
- Microsoft Azure Fundamentals
- Lab: Deploying a Web Application to Microsoft Azure
- Lab: Upload an Image to Azure Blob Storage