

Index

Contents

| | |
|--|-----------|
| <i>Immersive Course Structure .NET Core with Azure</i> | <i>2</i> |
| <i>Programming Foundation.....</i> | <i>3</i> |
| <i>.NET Core 6.0 and C# 10.0.....</i> | <i>3</i> |
| <i>Git</i> | <i>6</i> |
| <i>RDBMS & SQL Server.....</i> | <i>6</i> |
| <i>LINQ and Entity Framework Core.....</i> | <i>8</i> |
| <i>ASP.NET Core MVC.....</i> | <i>9</i> |
| <i>Azure DevOps</i> | <i>10</i> |
| <i>Cloud Computing & Azure Fundamentals.....</i> | <i>10</i> |
| <i>Cloud Networking & Storage</i> | <i>11</i> |
| <i>Azure PaaS Services</i> | <i>12</i> |

IMMERSIVE COURSE STRUCTURE .NET CORE WITH AZURE

.NET with Azure LOT provides exposure to the entire spectrum of .NET Core with Azure technologies. It focuses on Desktop as well as Cloud application development using .NET Technologies. The following table lists 40 Days Immersive course structure for .NET Core with Azure.

| Sr. No. | Course | Duration | Remarks |
|-------------------------|--|----------|---|
| 1 | Power Skills | 5 | |
| 2 | Programming foundation | 3 | |
| 3 | .NET Core + C# 10.0 | 6 | |
| 4 | Git | 1 | |
| 5 | RDBMS and SQL Server | 3 | |
| 6 | Module Test | 0.5 | Module Test |
| 7 | LINQ and Entity Framework Core | 2.5 | (Overview on ADO.NET Can be provided for 1 or 2Hrs.) |
| 8 | ASP.NET Core MVC | 4 | |
| 9 | Azure DevOps | 1 | |
| 10 | Cloud Computing & Microsoft Azure Fundamentals | 2 | |
| 11 | Cloud Networking and Storage | 2 | |
| 12 | Azure PaaS Services | 3 | |
| 13 | Sprint Implementation | 4 | Project use case to be implemented using Azure Function App + Azure Logic App + Azure Service Bus |
| 14 | Module Test + Sprint Evaluation | 1 | |
| 15 | L1 Preparation | 1 | |
| 16 | L1 Test | 1 | |
| Total Training Duration | | 40 | |

.NET Core with Azure Curriculum

Programming Foundation

Program Duration: 3 Days.

Table of contents

- Introduction to program development with pseudocode
- Good Programming Practices
- Algorithm Analysis and Design
- C# language fundamentals
- Data Structure and Algorithm

.NET Core 6.0 and C# 10.0

Program Duration: 6 Days.

Table of contents

- Introduction to .NET Core
 - .NET Core - Overview
 - Characteristics of .NET Core
 - The .NET Core Platform
 - .NET CORE architecture and Advantages
 - Build and run Cross platform apps
 - .NET Core - Environment Setup
 - .NET Core - Code Execution
 - Middleware
 - .NET Core - Modularity
 - .NET Core - Project Files
 - IIS Publishing & different cross platform deployments
 - Microservices using .NET Core
 - .NET Core - Windows Runtime and Extension SDKs.
 - .NET Core - Create .NET Standard Library .
 - What is .NET Framework
 - Comparison between .NET Framework & .NET Core
- Introduction to C#
 - Features of C#
 - C# Compilation and Execution
 - General Structure of a C# Program
- Data Types and Arrays in C#
 - Data Types in C#
 - Value Types and Reference Types
 - Boxing and UnBoxing
 - Single Dimensional, Multi-Dimensional & Jagged arrays
 - Nullable Types
 - Implicitly Typed Local variables
 - Var vs dynamic

- Is and as operator
- Ref vs out keywords
- The 'object' base class in .net
- Equals() vs ==
- String vs StringBuilder
- Various String class methods
- Default parameters, named parameters
- Parse() vs TryParse() vs Convert Class methods
- Debugging in C#
 - Various Types of .NET Projects
 - Tracing, Debugging, Build
 - Compile Options
 - Debugging Exception
 - Using watch and output window
 - What are Diagnostics?
 - Debug and Trace Classes
 - Creating multiple projects within one solution
 - Customizing Visual Studio Settings - Extensions, NUGet Package, Environmental Settings
- OOP with C#
 - Structures and enums
 - The architecture of a class in C#
 - Instance, Class & Reference variables
 - Access Modifier
 - Abstract Classes
 - Constructors, Destructors, The GC
 - .NET Base class library
 - Inheritance in C#
 - Method Overloading
 - Method Overriding
 - Operator Overloading
 - Method Hiding
 - Access modifiers : private, public, protected, internal, protected internal, new
 - Anonymous types
 - Abstract classes
 - Sealed classes
 - Creating Interfaces
 - Implementing Interface inheritance
 - Declaring properties within Interfaces
 - Namespaces
 - Creating and using Generic classes
 - Indexers & Properties
 - Auto Implemented properties
 - Static Classes
 - Property Accessors
 - Partial types
 - Extension methods
 - Object Initializer
- Evaluating Regular Expressions in C#
 - RegEx Class

- Forming Regular Expression
 - Methods for Regular Expression
- Exception Handling
 - Exceptions in C#
 - Exception class hierarchy
 - Try block
 - Multiple catch blocks
 - Finally block
 - Purpose of throw keyword
 - Purpose of inner exception
 - Creating Custom Exception
- Garbage Collection in C#
 - Role of a Garbage Collector
 - Garbage Collection Algorithm
 - Finalize vs Dispose
- Collections & Generics
 - System.Collections Namespace
 - Collection Interfaces
 - Collection Classes
 - The collection API
 - IEnumerable, IEnumerator, IComparer interfaces
 - Working with Generics
 - Creating Generic class, Generic Methods, Interfaces, Delegates
 - Collection Initializers
 - Iterators
 - Constraints
- Anonymous Types, Delegates, Events & Lambda
 - Extension Methods
 - Anonymous Type
 - Var and Dynamic
 - Introduction to Delegates
 - Events in C#
 - Anonymous Methods
 - Lambda Expression
- File I/O and Serialization
 - Using StreamReader, StreamWriter
 - Using BinaryReader, BinaryWriter
 - Using File, FileInfo, Directory, DirectoryInfo
 - Serialization modes: Binary, SOAP, XML
 - JSON serialization
- Reflection and Attributes
 - What is Reflection?
 - Using Reflection to read type information
 - Attributes.
 - Pre-defined Attributes
 - Custom Attributes.
 - Using Reflection to read custom attributes
-
- Threading, Parallel and Async programming with C#
 - Task Parallel Library

- Threads Vs. Tasks
- Thread state
- Task Based Asynchronous Model
- Async and Await
- Using Locks
- New Features in C# 10.0
 - Record structs
 - Improvements of structure types
 - Interpolated string handlers
 - global using directives
 - File-scoped namespace declaration
 - Extended property patterns
 - Improvements on lambda expressions
 - Allow const interpolated strings
 - Record types can seal ToString()
 - Improved definite assignment
 - Allow both assignment and declaration in the same deconstruction
 - Allow AsyncMethodBuilder attribute on methods
 - CallerArgumentExpression attribute
 - Enhanced #line pragma
 - Warning wave 6
- Packaging and Deployment
 - File System Editor
 - Registry Editor
 - File Types Editor
 - User Interface Editor
 - Custom Actions
 - Launch Condition Editor
 - Creating Uninstall Shortcut

Git

Table of contents

Program Duration: 1 Day.

- Getting Started with Git
 - Install the Git Tools
 - Clone an Existing Repository
 - Add Files to a Repository
 - Edit Files in a Git Repository
 - Create and Merge Branches
 - Rewrite History in a Git Repository
 - Resolve Merge Conflicts

RDBMS & SQL Server

Program Duration: 3 Days.

Table of contents

- Introduction to RDBMS
 - Introduction to databases
 - Data Models in Database
 - Properties of RDBMS
 - Normalization
 - CODD's Relational Database Rules
 - Data Integrity
 - T-SQL Language
- Working with Data Types, Tables & Data Integrity covering DDL, DML, DCL statements
 - Working with Data Types (Only Basics of Data Types)
 - Working with Schema
 - Working with Tables
 - Implementing Data Integrity
- Beginning with Transact-SQL
 - Transact-SQL
 - System Functions
 - Advanced T-SQL Queries`
 - Advanced T-SQL Statements
 - Other T-SQL Statements
 - Set Operators
 - Transact-SQL
 - System Functions
 - Advanced T-SQL Queries
 - Advanced T-SQL Statements
 - Other T-SQL Statements
- Working with Joins and Subqueries
 - What are Joins?
 - Types of joins
 - Subqueries
- Database Objects: Indexes and Views
 - Introduction to Index in SQL Server
 - Introduction to Views in SQL Server
- Stored Procedures
 - Stored Procedure
 - Implementing Stored Procedure
 - Exception handling using TRY-CATCH
- NoSQL Database
 - Brief History of NoSQL Databases
 - NoSQL Database Features
 - Types of NoSQL Database
 - Difference between RDBMS and NoSQL
 - Why NoSQL?
 - When should NoSQL be Used?
 - Demo
- Azure SQL Database:
 - Introduction to Azure SQL Database
 - Demo

LINQ and Entity Framework Core

Program Duration: 2.5 Days.

Table of contents

- Language Integrated Query
 - Introduction , LINQ Syntax
 - Introduction to System.Linq.Queryable
 - Query Operators
 - Select, from, Where
 - ofType
 - OrderBy
 - ThenBy
 - GroupBy, into
 - Select
 - SelectMany
 - Take, TakeWhile
 - First
 - FirstOrDefault
 - Single
 - SingleOrDefault
 - Aggregate functions Sum, Min, Max, Average, Count
 - Distinct
 - Intersect
 - Except
 - Join
 - LINQ projection
 - Deferred execution vs immediate execution
 - Let keyword
 - LINQ to Object
 - LINQ to DataTable
- Entity Framework Core
 - Overview of ORM Products
 - Entity Framework introduction
 - Using Database first Approach
 - Using Code First approach
 - Implementing Repository Pattern
 - Introduction & Benefits
 - Repository Pattern implementation
 - Setting up Entities in EFCore
 - Using LINQ to Entities to perform CRUD operations
 - SQL Query Logging
 - Migration & Database Update
 - Eager Loading Vs Explicit Loading Vs Lazy Loading
 - Raw SQL And Stored Procedures

ASP.NET Core MVC

Program Duration: 4 Days.

Table of contents

- ASP.NET Core Introduction
 - .NET Core Theory
 - Introduction to ASP.NET Core
 - Setting ASP.NET Core Development Environment
- Setup and Project Overview
 - Creating an ASP.NET Core Project
 - Project File and Program File
 - Startup Class and Service
 - Startup File, .Net Core Pipeline and Middleware
 - Routing in MVC and Endpoints
 - Launchsettings AppSettings and WWWRoot
 - Static Content
 - Shared Folder as well as ViewImports and ViewStart
- Dependency Injection
 - Understanding DI
 - Dependency Chains & Dependency Methods
- Controllers and Routing
 - Introduction to Controller
 - Creating Controller
 - Introduction to Routing
 - Transfer Data
 - Conventional Routing
 - Attribute Based Routing
 - URL Routing
 - Endpoint Routing
 - Areas
 - Controller Actions
 - Passing Values to Actions
 - Action Return Types
- Model, Database and CRUD
 - Introduction to Model
 - Creating Our First Model
 - Code First Approach
 - Setting Up Connectionstring
 - Setting up DbContext as well as EntityFrameWork
 - Setting Up the DB Context in Our Startup File
 - Passing Data To The Controller
 - Server Side Validation and Client Side Validation
- ViewBag, ViewData and ViewModel
 - Viewbag and Passing Data From A Controller To A View
 - ViewBag And ViewData
 - ViewModels and Strongly Typed Views
 - ViewModels in Action
 - Add Styling and JS
 - Layout

- Helper Class
 - View Component
- Filters
 - Introduction to Filters
 - Filter Types
- Advanced Features
 - Session in .NET Core
 - TempData
 - Authentication in .NET Core & Implementing Authentication Identity
 - Token Based Authentication -JWT
 - Authorization in .NET Core
 - Web Sockets
 - Deployment modes of .NET core Application
- Microservices Fundamentals
 - ASP.NET Core Microservices
 - Advance
 - Introduction to Docker
 - Choosing Between .NET 6 and .NET Framework for Docker Containers
 - Architecting container and microservice-based applications

Azure DevOps

Table of contents

Program Duration: 1 Day

- Azure DevOps
 - Introduction to Azure DevOps
 - Why Azure DevOps?
 - Components of Azure DevOps
 - Pipelines
 - Boards
 - Artifacts
 - Repos
 - Test Plans

Cloud Computing & Azure Fundamentals

Program Duration: 2 Days.

Table of contents

- Cloud Computing & Microsoft Azure Fundamentals
- Microsoft Azure Portal Overview (Utilization, Cost)
- Overview of Cloud Computing (Benefit, Capex/Opex)

- Overview of Public and Private Cloud & Hybrid
- Core Azure Service
- Availability Zones, Availability Set, Resource Groups, Azure Resource Manager
- Identity, Azure Active Directory, Users & Groups

Cloud Networking & Storage

Program Duration: 2 Days.

Table of contents

- Azure Virtual Machine
 - Create Virtual Machine
 - Manage Virtual Machine
 - Create/Manage Disks
 - Create VM images
 - Create Scale Set
 - Load Balance VMs
- Virtual Network
 - Create Virtual Network (Portal, PowerShell & CLI)
 - Add, Change or Delete a Subnet
 - Connect Virtual Networks
 - Virtual Network Peering
 - Filter Network Traffic
 - Route Network Traffic
 - Route Table
 - Restrict Virtual Network
 - Secure Network Traffic
 - Network Security Groups
 - Application Security Groups
 - Virtual Network Service End Points
 - Monitor Virtual Networks
 - Virtual Network Terminal Access Point (TAP)
- Azure Load Balancer
 - Load Balancer Algorithm
 - Load Balancer Components
- VPN Gateway, Azure Application Gateway
 - Create & Manage VPN Gateway
 - Create Site to Site VPN Connection
- Azure CDN
 - Create an Azure CDN Profile and endpoint
 - Monitor health of Azure CDN Resources
- Azure Storage Services
 - Core Storage Services
 - Azure Blobs
 - Azure Files

- Azure Queues
 - Azure Tables
 - Azure Disks
- Creating Storage Account
- Azure Blob Storage
 - Work with blobs
 - Upload, download & list blobs
 - Encrypt & Decrypt blobs using Azure Key Vault
- Introduction to Azure Data Lake Storage
- Authorize access to Azure Storage
- Choosing Data Storage Technology in Azure
- Structured Data, Semi Structure Data, Unstructured Data
- Accessing Azure Storage using Azure SDK & C#.NET
- Azure API Management
 - API Management Features
 - Create an Instance

Azure PaaS Services

Program Duration: 3 Days.

Table of contents

- Azure Web App
 - What is App Service
 - App Service Plan and Comparison between them
 - App Service Environments
 - Creating ASP.NET Web APP
 - Deploying App Using Visual Studio & Kudu
 - Run App in Staged Environments using deployment slots
 - Working with Configurations
 - Accessing Configuration using ASP.NET Application
 - Host API with CORS
 - Creating and Using Web Jobs
- Azure Function App
 - Overview of Serverless Computing & Benefits
 - Create Function App using C# in Azure Portal
 - Deploying Function App using Visual Studio
 - Bindings & Triggers and their Types
 - Implementing Various Triggers
 - Accessing Functions using Keys
 - Durable Functions
 - Serverless Comparison
 - Hosting Plan

- Azure Service Bus
 - Service Bus Messaging
 - Queues, Topics, and Subscriptions
 - Create Service Bus Queue
 - Create Service Bus Topics & Subscriptions
 - Send and Receive Messages - Queues
- Azure Logic App
 - Introduction to Logic App
 - Single-tenant versus Multi-Tenant
 - Creating Logic App
 - Schedule Based Workflows
 - Approval Based Workflows
 - Creating Azure Storage and Azure Function Work Flow
 - Deploy Logic Apps
 - Automated Logic App Deployment
- Azure Database Services & Azure SQL
 - Database Workloads in Azure
 - OLAP in Azure Introduction
 - Non-Relational Databases in Azure
 - Azure SQL Introduction
 - Migrate to Azure SQL
 - SQL Managed Instances
 - Elastic Pools
 - Instance Pools
 - Create SQL Database
 - Configure Firewall
 - Configuring Security
 - Logins, user accounts, roles, and permissions
- Containers, ACR & ACI
 - Introduction to Containerization of Web Apps
 - Comparing On-Premises, Virtual Machines & Containerized deployments
 - Introduction to ACI & Features
 - Docker Hub Vs ACR