Index

Contents

IMMERSIVE LEARNING - COURSE STRUCTURE .NET LOT

.NET LOT provides exposure to the entire spectrum of .NET technologies. It focuses on Desktop as well as Web application development using .NET Technologies. The following table lists the proposed course structure for .NET LOT.

Sr. No.	Course	Duration	Remarks
1	Discover	-	Online
2	Soft Skills Part 1	1	Soft Skills Part 1
3	.NET Framework 4.6 + C# 7.0	11	
4	Git	1	
5	Soft Skills Part 2	1	Soft Skills Part 2 (Saturday)
6	.NET Framework 4.6 + C# 7.0 Test	0.5	Module Test (MCQ + Coding)
7	RDBMS and SQL Server	5	
8	ADO.NET	2	
9	LinQ and Entity Framework Core	2.5	
10	Soft Skills Part 3	1	Soft Skills Part 3
11	SQL Server + ADO.NET + LINQ and EF MCQ Test & Sprint 1 Implementation	4	Project use case to be implemented using Console Application and LINQ and Entity Framework Core
12	Sprint 1 Evaluation	1	
13	ASP.NET Web Forms	4	
14	Soft Skills Part 4	1	Soft Skills Part 4 (Evaluation)
15	ASP.NET Core Web API	2.5	
16	ASP.NET Core MVC	4	
17	ASP.NET + MVC + Web API + WCF MCQ Test & Sprint 2 Implementation	4.5	Project use case to be implemented using ASP.NET Core API Service and consuming the same in ASP.NET and ASP.NET Core MVC
18	Sprint 2 Evaluation	1	
19	L1 Test	1	
	Total Training Duration	48	

.NET Curriculum
.NET Framework 4.6 and C# 7.0

Program Duration: 11 Days.

Table of contents

- 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, pubic, 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
    - Working with Generics
    - Creating Generic class, Generic Methods, Interfaces, Delegates
    - Collection Initializers
    - Iterators
    - Constraints
- Delegates and Events
    - Introduction to Delegates
    - Implementing Delegates
    - Single & Multicasting Delegate
    - Creating callback method using Delegate
    - Events in C#
    - Generic delegates and generic Delegates, Generic Classes
    - Action, Func and Predicate Delegate
- File I/O and Serialization
    - Persisting object state to a stream
    - Various classes used for File handling
    - Using StreamReader, StreamWritter
    - Using BinaryReader, BinaryWriter
    - Using File, FileInfo, Directory, DirectoryInfo
    - Serialization modes: Binary, SOAP, XML
    - JSON serialization using DataContractJsonSerializer
    - Performing various serializations,
    - Runtime serialization (Deep)
    - Marking a class serializable
    - Serialization and inherited classes
    - Customizing Serialization by Using attributes
    - Implementing ISerializable interface
- Assemblies, Reflection & Attribute based programming
    - Introduction to .NET Reflection
    - Obtaining details about types from the assembly(Private,Public,Shared and Satellite)
    - Obtaining details about methods, properties and fields
    - Creating Custom Attributes
    - Retrieving Attribute Details with Reflection
- Threading, Parallel and Async programming with C#
    - Appdomain vs Process vs Thread
    - Process vs Thread
    - Creating and running a thread
    - Thread.Sleep() method
    - Parallelization Overview
    - Task Parallel Library
    - Threads Vs. Tasks
    - Parallel Extensions in .NET 4.5
    - Task Based Asynchronous Model in .NET 4.5
    - Async and Await
    - Using Locks
- New Features in C# 6.0
    - Using Static
    - String Interpolation

    - Dictionary Initializers
    - Auto property Initializers & Getter only auto properties
    - nameOf Operator
    - Await in catch/finally
    - Null conditional operator and Null Propagation
    - Expression bodied members
    - Using static with Extension Methods
    - Exception filters
- New Features in C# 7.0
    - Out Variable Enhancements
    - Pattern Matching
    - Tuples
    - Local or Nested Functions
    - ref Returns and Locals
    - Expression Bodied Members
    - Throw Expressions
    - Literal Improvements
    - Generalized async Return Types
- Encrypting and Decrypting
    - Implementing Symmetric & Asymmetric Encryption
- Utilizing Class Libraries & Console Project
    - The .NET Console Application Project Type
    - References and Importing Namespaces
    - Instancing Classes
- IOC Container
    - Dependency Injection – Only Concept and Simple Demo

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: 5 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
- Introduction to SQL Server 2012
    - What is SQL Server?
    - SQL Server Components
    - SQL Server Authentication Modes
    - SQL Server Services
    - SQL Server Version History
    - SQL Server Editions
- 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
- Implementing Triggers
    - Introduction to Triggers
    - Defining Triggers
- Transactions
    - Introduction to Transactions
- SQL Server Profiler
    - SQL Server Profiler
    - Working with SQL Server Profiler in SQL Server Management Studio

ADO.NET 4.5

Program Duration: 2 Days.

Table of contents

- ADO.NET Architecture
    - .NET Data Providers
    - DB Connectivity Architectures in .NET
    - Elements of .NET Data Providers
    - Introduction to SQL Server
    - Namespaces in ADO.NET
    - Using server explorer window
    - Connection class
    - Command class
    - Direct Command execution against database
    - Using Parameters in command
    - Performing CRUD operations
    - Connected Vs disconnected Architecture
    - Data reader class
    - The dataset and dataset Architecture
    - Comparison ADO & ADO.NET on Disconnected Data architecture
    - Implementing Disconnected Data Architecture
    - Performing CRUD operations in disconnected architecture
    - ReadXml(),ReadXmlSchema(), WriteXml(), WriteXmlSchema() usage.
    - Implementing Data Access Layer
    - Guidelines for Designing DAL

LINQ and Entity Framework Core

Program Duration: 2.5 Days.

Table of contents

- Language Integrated Query
    - Introduction , LINQ Syntax
    - 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 Model First approach
    - Using Code First approach
    - 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 Web Forms 4.5

Program Duration: 4 Days.

Table of contents

- Introduction to ASP.NET 4.5
    - Basic ASP.NET concepts, ASP.NET Framework
    - Web Forms and their features
    - Life Cycle of a Web Form
    - Advantages of ASP.NET over ASP
    - Concept of Code Behind, Web Configuration File
    - Page Life Cycle
    - ASP.NET 4.5 features
- Validation in ASP.NET 4.5
    - Validation Controls
    - Custom Validator
    - Validation Summary
    - Validation Group
    - Validation: Best Practices and Guidelines
- Using JavaScript
    - Overview
    - Validation
    - Model Popup
- Master Pages
    - Why Master Pages
    - Working with Master Pages

    - Coding a Master Page and Content Page
    - Nesting of Master Pages
    - Best Practices & Guidelines
- Tracing
    - Introduction to Tracing
- Data Driven Pages
    - Using DataSource controls provided in ASP.NET
    - Using GridView, FormView, Repeater and DetailsView Controls
- State Management & Caching
    - HTTP Basics
    - State Management
    - Client-side State Management - ViewState, Hidden Fields, QueryString, Cookies
    - Server-side State Management - Session Object, Application Object
    - Global.asax
    - Caching,Types of Caching,Caching Methods(Add,Insert)
    - State Management: Best Practices and Guidelines
- Deploying ASP.NET Applications
    - What to Deploy?
    - Steps to take before Deploying
    - Methods to Deploy
    - Copy Web Site Dialog
    - Deploying a Pre-Compiled Application
    - Deployment: Best Practices and Guidelines
- ASP.NET Identity

ASP.NET Core Web API

Program Duration: 2.5 Days.

Table of contents

- Introduction to .Net Core WebAPI
    - Introduction to Web Service with Demo
    - Introduction to WCF Service with Demo
    - Introduction to Web API
    - Difference between Web Service, WCF Service and Web API
    - Web API features
    - HTTP Web Services
    - Web API Introduction
    - Middleware
    - Web API Routing
    - Configuring WebApi
    - Web API Parameter Biding
    - Action Return Type
    - WebApi Filters
    - Content Negotiation
    - Create CRUD WebApi
    - Consume WebApi
    - MSTest / XUnit
        - Basic Unit Test

- Create CRUD WebApi
  - Working with swagger
  - Postman Utility

ASP.NET Core MVC

Program Duration: 4 Days.

Table of contents

        - Validation
            - Fluent Validation
- Using AJAX In ASP.NET MVC
        - Introduction to Ajax
        - AJAX in ASP.NET MVC
        - AJAX Helpers
        - Partial Page rendering
        - AJAX using jQuery
- Using ASP.NET Identity in ASP.NET MVC
- Consume WebAPI
        - JWT Token Authentication
        - Consume Methods
    - Code Commit
    - Deployment of Application
    - Consume WCF Service