SYLLABUS – COMPLETE C#, OOPS AND BASIC OF .NET CORE API

PRE-REQUISITE / TARGET AUDIENCE: Any beginner who wants to build career as Web designer can take this course

SYLLABUS - C# AND OOPS

MODULE 01: OBJECT ORIENTED PROGRAMMING CONCEPTS

- Class
- Object
- Encapsulation
- Inheritance
- Polymorphism
- Object Creation and Instantiation

MODULE 02: SOLID PRINCIPLES

- S: Single Responsibility Principle (SRP)
- O: Open closed Principle (OCP)
- L: Liskov substitution Principle (LSP)
- I: Interface Segregation Principle (ISP)
- D: Dependency Inversion Principle (DIP)

MODULE 03: DESGIN PATTERNS

- Singleton
- Factory
- Strategy
- Dependency Injection

MODULE 04: MS.NET FRAMEWORK INTRODUCTION

- The .NET Framework - an Overview
- Framework Components
- Framework Versions
- Types of Applications which can be developed using MS.NET
- Base Class Library, CLR, Managed code, Memory Management/ Garbage Collection, Types of JIT compilers

MODULE 05: C # LANGUAGE SYNTAX

- Why Datatypes Global
- Stack and Heap Memory
- Common Type System
- Reference Type and Value Type
- Datatypes & Variables Declaration
- Implicit and Explicit Casting
- Checked and Unchecked
- Casting between other datatypes
- Boxing and Unboxing
- Enum and Constant
- Operators
- Control Statements
- Working with Arrays
- Working with Methods
- Pass by value and by reference and out parameters

MODULE 06: COLLECTIONS AND GENERICS

- Introducing Collections
- Benefits of Collection Classes .
- Understanding and using commonly used collections.
- Generics Advantages of Generics .
- How Generics Work at Runtime .
- Constraint on Type Parameters .
- Generic Methods, Generic Collections
- Selecting a Collection Class

MODULE 07: EXCEPTION HANDLING

- Defining Exception
- Understandings try and catch keywords
- Using “finally” block “using” statement
- Throwing exceptions Creating User defined/Custom Exception class

MODULE 08: *IO STREAMS*

- What is a streams?
- Types of Stream Standard I/O Streams Console
- Handling text in files
- Dealing with Binary files Serialization / Deserialization

MODULE 09: DELEGATES & EVENTS

- Introduction to Delegates
- Events Declaration, Raising and Handling
- Anonymous Methods

MODULE 10: MULTITHREADING

- Threading Overview
- Scheduling
- Thread States
- Programming Threads
- Methods of Thread Class
- Thread Pool
- Thread Synchronization (Monitor ,Mutex ,Semaphore, Events)
- Parallel Programming using Task Parallel Library
- Asynchronous Programming using async and wait keywords

MODULE 11: BASIC OF ASP.NET CORE WEB API

- Difference between .NET Core & .NET framework
- What is .NET Core?
- Why .Net Core?
- Characteristics of .NET core
- Project Structure in .NET core
- Dependency Injection in .NET Core
- Middleware
- Custom Middleware
- Exception Handling
- Logging in .NET core
- Routing
- Filters

MODULE 12 : BASIC OF MICROSERVICES ARCHITECTURE

- Single Responsibility Principle in Microservices
- Microservices Architecture
- Monolithic VS Microservices
- Saga in Microservices

MODULE 13: RABBIT MQ

- What is RabbitMq?
- Architecture of RabbitMq
- How Microservices communicate using RabbitMq

SYLLABUS – GIT

MODULE 01: INTRODUCTION

- Source Control Management overview
- Why SCM tool required
- SVN vs GIT

MODULE 02: CLONING AND CREATING PROJECT

- Cloning - code base
- Init – create project
- States of GIT
  - Working directory
  - Staging area
  - Repository

MODULE 03: BASIC SNAPSHOTTING

- Staging file
- Status
- Commit
- Diff
- Reset

MODULE 04: GENERAL CONCEPTS

- Local repository
- Remote repository
- What is upstream and downstream?

MODULE 05: BRANCHING AND MERGING

- Branch
- Checkout
- Merge
- Log
- Stashing changes

MODULE 06: SHARING AND UPDATING PROJECTS

- Fetch
- Pull
- Push