DAY 1

FullStack development

- front end + middle layer + data layer Web site + web services + database management

online shopping application using .NET

- Database:
  - MySQL + local database (local sql server)
- Business logic :
  - Product catalog logic
  - Shopping cart
  - order proccesing
  - Payment and billing
  - Authentication and Authorization
  - shipment (Delivery)
- Service layer:
  - Product Catalog service
  - Shopping cart service
  - order proccesing service
  - Payment process and billing service
  - Authentication service
  - Authorization service
  - shipment (Delivery) service

light weight , business logic accessed by via HTTP request REST ful web services

- Web Site:
  - web user interface + online web application management
  - web pages + validation + state management, web security, caching, personalization, multilingual web, etc..
  - Responsive UI

Baby step Learning:

- Hello world using Visual Studio.NET
- learn basics of C# Progtamming
- Learn Visual studio .NET Application Development Process

Rules to create project

- Launch VS IDE
- create .NET Framework Console Application
- Start working with Project by modifying or adding C# code

Console Class

- Two static major fucntion
    - WriteLine,ReadLine

```
Console.ReadLine("hello world");
Console.WriteLine();
```

- Everything in C# is defined within namespace
- Each class should be defined in namepsace.
- Class can have static and nonstatic methods.
- Entrypoint function main is alawys methods.
- Command Arguments are always access througth strign[] args.
- public,private,protected and internal aree acccess specifiers in C#.
- Class can have overloaded ctor.
- class expose private data members to outside using member functionss as well as through properties.
- Properties always encapuslate private data members of instance.
- Properties can readonly or writeonly as well.
- It is good practice to wirte new class in seperate file.
- this keyword in C# represent self refrence of an instance.
- value is inbuilt keyword in C#

Day 3

> Agenda

1. C# Language Features
2. .NET Framework
3. Assembly

Official Documentation

- Microsoft official .NET Documentation

No.Stackoverflow--------> this site give lot of readymade problem

- solving samples

C# Language Features

Language syntax

- Basic syntax of C# is similar like C++
- New Keywords in C#
- New Approach while developing Solution
- Loosely Coupled, highly Cohesive
- Common Type System (CTS)

Value Types:

```
primitive types ( inbuilt types)
            int, float, double, ....
            enumeration
            structure
            values are stored on stack
```

Reference Types:

```
- class:
    - concrete class
    - abstract class
- interface:
    - IPrintable
    - ISacnnable
            Delegte:
            Event:
```

- they are always stored on heap
- Heap is used for dynamic memory allocation for objects in .NET
- Heap is managed by Garbage collector
- Garbade collector takes care Automatic Memory Management.

Day 5

Agenda:

- Design Pattern
- Singleton
- Observer : Event Driven Application
- C# reference types:
    - Delegate
    - Events
- Windows based App Development
- ClassFactory
- C# Programming
- Syntax :

- like C++, like Java, like VB
- Language Interoperability(Cross Language Programming)
- we can also pointers *ptr
- acces native code from C# using keyword unsafe
- simplified syntax with new kewords

```
as  is  ref out  params
sealed value interface;
abstract virtual override
region new foreach using
get set  const readonly static
this base
```

- Grammers:
  - Rules, constructs
  - Common Type System
- Value Types:
  - inbuilt (primitive) types,
  - Enum, struct
- Reference Types:
  - class, interface
  - delegate, event
- **Object Orientation:**
  - Abstraction,
  - Encapsulation,
  - Inheritance (Polymorphism),
  - Typing, Concurrency, Modularity, Persistence, Hierachy

> Reusable types: Framework Class Library (FCL)

- a set of classes,
- a set of interfaces,
- a set of structures,
- a set of enumeration,
- a set of attributes

Adopts Design Patterns

```
: using C# language we can apply design patterns while building apps.
```

> What do you means Design Pattern ?

- Pattern:
    - Design pattern are solution to software design problems you find again and again in real world app developement.
    - Patterns are about resualbe designs and interaction object.
    - There are almost 23 Gang of Four pattersn are consisered as Design Patters.

> Categories:

- Creational
    - singleton, class factory,etc.
- Structural
    - MVC, MVP,MVVM etc.
- Behavioural
    - Observer,etc.
- Vehicle:
    - Engine:
        - EME subject
        - two stroke-------Yamaha, Rajdoot, Yezdi, java, Bajaj M80
        - four stroke -----Splendor, pulsar, etc.
        - multi cyclinder engine------ Heavy Vehicles
- Software:
    - millions of softwares been developed in these 60 years.
    - Insurance
    - Telecom
    - Finance
    - Research & Development
    - Production

> Few Computer engg. discrovered common patterns applied in those solutions. They named it as Design Patterns

---

Design Patterns

> Can you write code using standard structure to solve common problem

- MVC (Model---- View------ Controller)
- Singleton (only instacne of class will be created)
- Class Factory ( ) .etc.

- Music:
    - Swar: (syntax)
        - Sa , Re, Ga.....
- Classical Music: (design patterns)
    - Ragas (Patterns)
        - composition of swar
- Bolloywood Song: (building Softwares)
    - has been composed my Musical Director
        - taal + laay + Sur =Melody

Event Drive Mechanism

- Event Delagation Model
- events: underbalance, overbalance
- delegations: HDFCBank, Goverment
- First Observer :
    - Account object (to be observed is balance)
- Events:
    - underbalance, overbalance (notify to external subscribers)
- Subscribers:
    - HDFC Bank , Government (respond to events occured against notification received)