DOTNET

UNIT 2

Introduction to C#

C# is pronounced as "C-Sharp". It is an object-oriented programming language provided by Microsoft that runs on .Net Framework.
By the help of C# programming language, we can develop different types of secured and robust applications:

- Window applications
- Web applications
- Distributed applications
- Web service applications
- Database applications etc.

C# is approved as a standard by ECMA and ISO. C# is designed for CLI (Common Language Infrastructure). CLI is a specification that describes executable code and runtime environment.
C# programming language is influenced by C++, Java, Eiffel, Modula-3, Pascal etc. languages.

C# Features

C# is object oriented programming language. It provides a lot of **features** that are given below.

1. Simple
2. Modern programming language
3. Object oriented
4. Type safe
5. Interoperability

6. Scalable and Updateable
7. Component oriented
8. Structured programming language
9. Rich Library

1) Simple

C# is a simple language in the sense that it provides structured approach (to break the problem into parts), rich set of library functions, data types etc.

2) Modern Programming Language

C# programming is based upon the current trend and it is very powerful and simple for building scalable, interoperable and robust applications.

3) Object Oriented

C# is object oriented programming language. OOPs makes development and maintenance easier where as in Procedure-oriented programming language it is not easy to manage if code grows as project size grow.

4) Type Safe

C# type safe code can only access the memory location that it has permission to execute. Therefore it improves a security of the program.

5) Interoperability

Interoperability process enables the C# programs to do almost anything that a native C++ application can do.

6) Scalable and Updateable

C# is automatic scalable and updateable programming language. For updating our application we delete the old files and update them with new ones.

7) Component Oriented

C# is component oriented programming language. It is the predominant software development methodology used to develop more robust and highly scalable applications.

8) Structured Programming Language

C# is a structured programming language in the sense that we can break the program into parts using functions. So, it is easy to understand and modify.

9) Rich Library

C# provides a lot of inbuilt functions that makes the development fast.

Introduction to ASP.NET

It is a web framework designed and developed by Microsoft. It is used to develop websites, web applications and web services. It provides fantastic integration of HTML, CSS and JavaScript. It was first released in January 2002. It is built on the Common Language Runtime (CLR) and allows programmers to write code using any supported .NET language.
ASP.NET is a part of Microsoft .NET Framework. The following image shows the component stack.

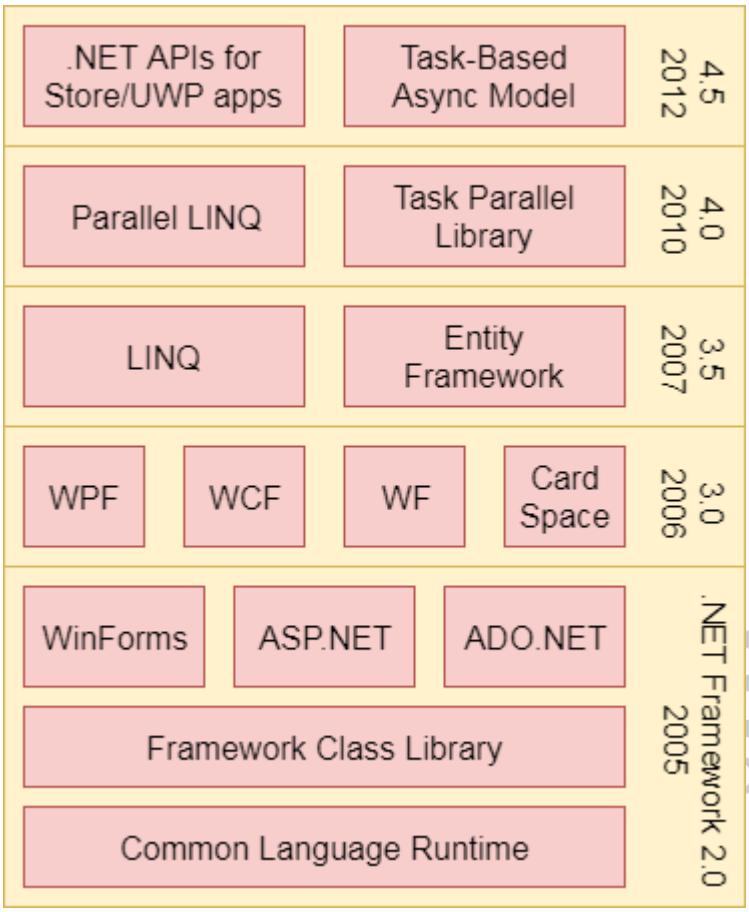

Fig: .NET framework components

ASP.NET provides three development styles for creating web applications:

1. Web Forms
2. ASP.NET MVC
3. ASP.NET Web Pages

- **WEB FORMS**

It is an event driven development framework. It is used to develop application with powerful data access. It provides server side controls and events to create web application. It is part of the ASP.NET framework.

- **ASP.NET MVC**

It gives us a MVC (Model View Controller), patterns-based way to build dynamic websites. It enables a clean separation of concerns and that gives you full control over markup for enjoyable, agile development. It also provides many features that enable fast development for creating outstanding applications.

- **ASP.NET Web Pages**

It is used to create dynamic web pages. It provides fast and lightweight way to combine server code with HTML. It helps to add video, link to the social sites. It also provides other features like you can create beautiful sites that conform to the latest web standards.
All these are stable and well equipped frameworks. We can create web applications with any of them. These are also based on the .NET Framework and share core functionalities of .NET and ASP.NET.
We can use any development style to create application. The selection of style is depends on the skills and experience of the programmer.
Although each framework is independent to other, we can combine and use any of that at any level of our application. For example, to develop client interaction module, we can use MVC and for data control, we can use Web Forms.

ASP.NET Page Lifecycle

In ASP.NET, a web page has execution lifecycle that includes various phases. These phases include initialization, instantiation, restoring and maintaining state etc. it is required to understand the page lifecycle so that we can put custom code at any stage to perform our business logic.

Page Lifecycle stages

The following table contains the lifecycle stages of ASP.NET web page.

Stage	Description
Page request	This stage occurs before the lifecycle begins. When a page is requested by the

	user, ASP.NET parses and compiles that page.
Start	In this stage, page properties such as Request and response are set. It also determines the Request type.
Initialization	In this stage, each control's UniqueID property is set. Master page is applied to the page.
Load	During this phase, if page request is postback, control properties are loaded with information.
Postback event handling	In this stage, event handler is called if page request is postback. After that, the Validate method of all validator controls is called.
Rendering	Before rendering, view state is saved for the page and all controls. During the rendering stage, the page calls the Render method for each control, providing a text writer that writes its output to the OutputStream object of the page's Response property.

Unload	At this stage the requested page has been fully rendered and is ready to terminate.at this stage all properties are unloaded and cleanup is performed.

A requested page first loaded into the server memory after that processes and sent to the bowser. At last it is unloaded from the server memory. ASP.NET provides methods and events at each stage of the page lifecycle that we can use in our application. In the following table, we are tabled events.

ASP.NET Web Forms

Web Forms are web pages built on the ASP.NET Technology. It executes on the server and generates output to the browser. It is compatible to any browser to any language supported by .NET common language runtime. It is flexible and allows us to create and add custom controls.

We can use Visual Studio to create ASP.NET Web Forms. It is an IDE (Integrated Development Environment) that allows us to drag and drop server controls to the web forms. It also allows us to set properties, events and methods for the controls. To write business logic, we can choose any .NET language like: Visual Basic or Visual C#.

Web Forms are made up of two components: the visual portion (the ASPX file), and the code behind the form, which resides in a separate class file.

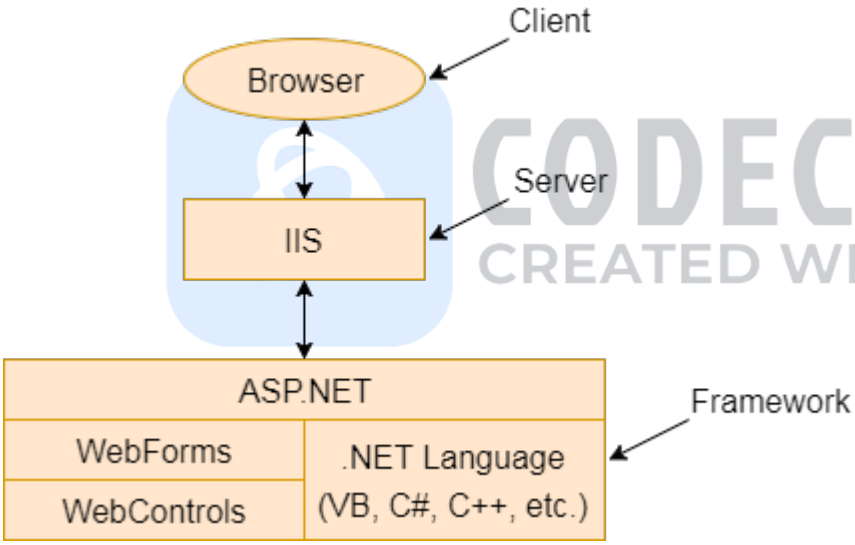

Fig: This diagram shows the components of the ASP.NET

The main purpose of Web Forms is to overcome the limitations of ASP and separate view from the application logic

ASP.NET Web Forms Server Controls

ASP.NET provides web forms controls that are used to create HTML components. These controls are categories as server and client based. The following table contains the server controls for the web forms.

Control Name	Applicable Events	Description
Label	None	It is used to display text on the HTML page.

TextBox	TextChanged	It is used to create a text input in the form.
Button	Click, Command	It is used to create a button.
LinkButton	Click, Command	It is used to create a button that looks similar to the hyperlink.
ImageButton	Click	It is used to create an imagesButton. Here, an image works as a Button.
Hyperlink	None	It is used to create a hyperlink control that responds to a click event.
DropDownList	SelectedIndexChanged	It is used to create a dropdown list control.
ListBox	SelectedIndexCnhaged	It is used to create a ListBox control like the HTML control.
DataGrid	CancelCommand, EditCommand, DeleteCommand,	It used to create a frid that is used to show data. We can also perform paging,

	ItemCommand, SelectedIndexChanged, PageIndexChanged, SortCommand, UpdateCommand, ItemCreated, ItemDataBound	sorting, and formatting very easily with this control.
DataList	CancelCommand, EditCommand, DeleteCommand, ItemCommand, SelectedIndexChanged, UpdateCommand, ItemCreated, ItemDataBound	It is used to create datalist that is non-tabular and used to show data.
CheckBox	CheckChanged	It is used to create checkbox.
CheckBoxList	SelectedIndexChanged	It is used to create a group of check boxes that all work together.
RadioButton	CheckChanged	It is used to create radio button.

RadioButtonList	SelectedIndexChanged	It is used to create a group of radio button controls that all work together.
Image	None	It is used to show image within the page.
Panel	None	It is used to create a panel that works as a container.
PlaceHolder	None	It is used to set placeholder for the control.
Calendar	SelectionChanged, VisibleMonthChanged, DayRender	It is used to create a calendar. We can set the default date, move forward and backward etc.
AdRotator	AdCreated	It allows us to specify a list of ads to display. Each time the user re-displays the page.
Table	None	It is used to create table.
XML	None	It is used to display XML documents within the

		HTML.
Literal	None	It is like a label in that it displays a literal, but allows us to create new literals at runtime and place them into this contro

.