→ .NET

1. '.' is a access specifier, and 'NET' stands for Network Enabled Technology
2. It is a Software development framework designed & developed by Microsoft, first version was 1.0 released in 2002
3. It can be seen as a virtual machine for compiling and executing programs written in different languages like C#, VB.NET, etc.
4. It provides a runtime environment and a set of libraries & tools for building & running applications on Windows operating systems.
5.

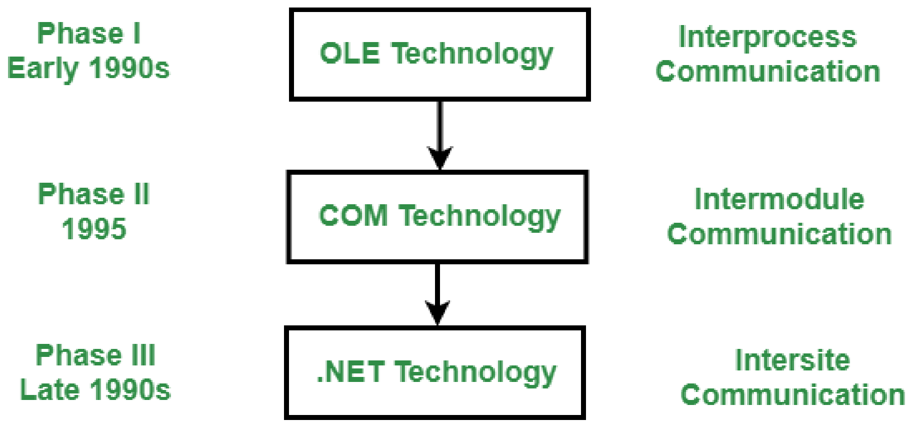
→ Phases of development of .NET Technology

There are three significant phases of development of .NET technology:

a. <u>Object Linking and Embedding (OLE) Technology</u>
    i. Phase 1, early 1990's, interprocess communication
    ii. Is one of the technologies of Microsoft component document
    iii. Main purpose is to link elements from different applications with each other

b. <u>Common Object Model (COM) Technology</u>
    i. Phase 2, 1995, inter-module communication
    ii. This technology of Microsoft windows family of OS enables various software components to communicate
    iii. It is mostly used by developers for various purposes like creating reusable software components, linking components together to build applications, and also taking advantage of Windows services
    iv. The objects of COM can be created with a wide range of programming languages.

c. <u>.NET (Network Enabled Technology)</u>
    i. Phase 3, Late 1990s, Inter-Site COmmunication
    ii. Set of technologies to develop windows and web applications
    iii. .NET is developed by Microsoft and launched in Feb 2002, also known as Microsoft's new Internet strategy
    iv. It was originally called Next Generation Web Services (NGWS)
    v. Considered to be one of the most powerful, popular and very useful Internet Technologies available today



→ .NET Standard

a. It is a specification that defines a common set of APIs that can be used by .NET applications.
b. This makes it possible to write code that can run on different .NET platforms such as .NET Framework, .NET Core and .NET5 and beyond

→ .NET Framework

a. Versions of .NET Framework are
   i. .NET 1.0 (Feb 13, 2002) : Main feature was Common Language Runtime (CLR) and it supported Object-Oriented Development of the Web-Apps
   ii. .NET 1.1 (Apr 2003) : featured security enhancements for ASP.NET, support for IPv6, ODBC
   iii. .NET 2.0 (Nov 2005) : had generic collections, iterators, new features for ASP.NET , nullable types, etc.
   iv. .NET 3.0 (Nov 2006) : featured Windows Presentation Foundation (WPF), WIndows Communication Foundation (WCF), Windows Workflow Foundation (WWF)
   v. .NET 3.5 (Nov 2007) : included support for AJAX, Dynamic Data, LINQ, ASP.NET MVC
   vi. .NET 4.0 (Apr 2010) : Managed Extensibility Framework (MEF), Dynamic Library Runtime (DLR), Task Parallel Library, Razor View Engine
   vi. .NET 4.5 (Aug 2012) : support for Async , better ASP.NET support, support for ZIP compression, improved CLR4.0
   vii. .NET 4.5.1 (Oct 2013) : better performance & debugging, advanced support for Windows Store Application development
   viii. .NET 4.5.2 (Aug 2014) : .NET core was announced and was being developed alongside

ix. .NET 4.6 (Jul 2015) : new JIT compiler , open source packages, support for both TLS 1.1, and TLS 1.2, better event tracing
x. .NET 4.6.1 (Nov 2015) : better performance, better support for Digital signature algorithm, distributed transactions in Azure SQL DB
xi. Final version v4.8.x is not actively being developed, considered a legacy technology, but is still supported

d. This is the original version of the .NET platform, but it is mostly used for building Windows Desktop and Web-Applications
e. .NET Framework is a powerful and versatile development platform that provides a wide range of tools and libraries for building and running applications on Windows OS
f. .NET Framework is used to develop Form-based applications, Web-based Applications and Web Services.
g. It is used to develop both desktop as well as server-based applications
h. There is a variety of programming languages available on .NET platform, VB.NET and C# being the most common ones

i. Components of .NET Framework 3.5

.NET Framework is composed of four main components :

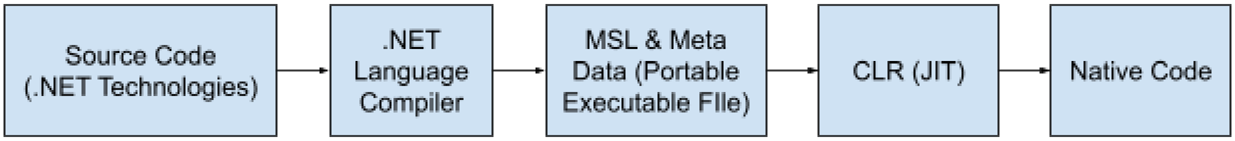
a. Common Language Runtime (CLR)
b. Framework Class Library (FCL)
c. Core Languages (WinForms, ASP.NET, ADO.NET)
d. Other Modules (WCF, WPF, WF, Card Space, LINQ, Entity Framework, Parallel LINQ, Task Parallel Library, etc.)

But, Following lists out the components of .NET Framework :

ii. Common Language Runtime (CLR)

1. Responsible for managing the execution of code written in any of the supported languages
2. Performs memory management, exception handling, garbage collection, debugging, security checking, thread execution, code execution, verification and compilation. Moreover, it provides type safety, interoperability and portability.
3. Code which is directly managed by the CLR is called Managed code. When Managed code is compiled, the compiler converts the source code into a CPU independent Intermediate Language (IL). Then a Just-in-time (JIT) compiler compiles the IL code into native code, which is CPU specific
4. It is a program execution engine that loads and executes the program. It converts the program into native code

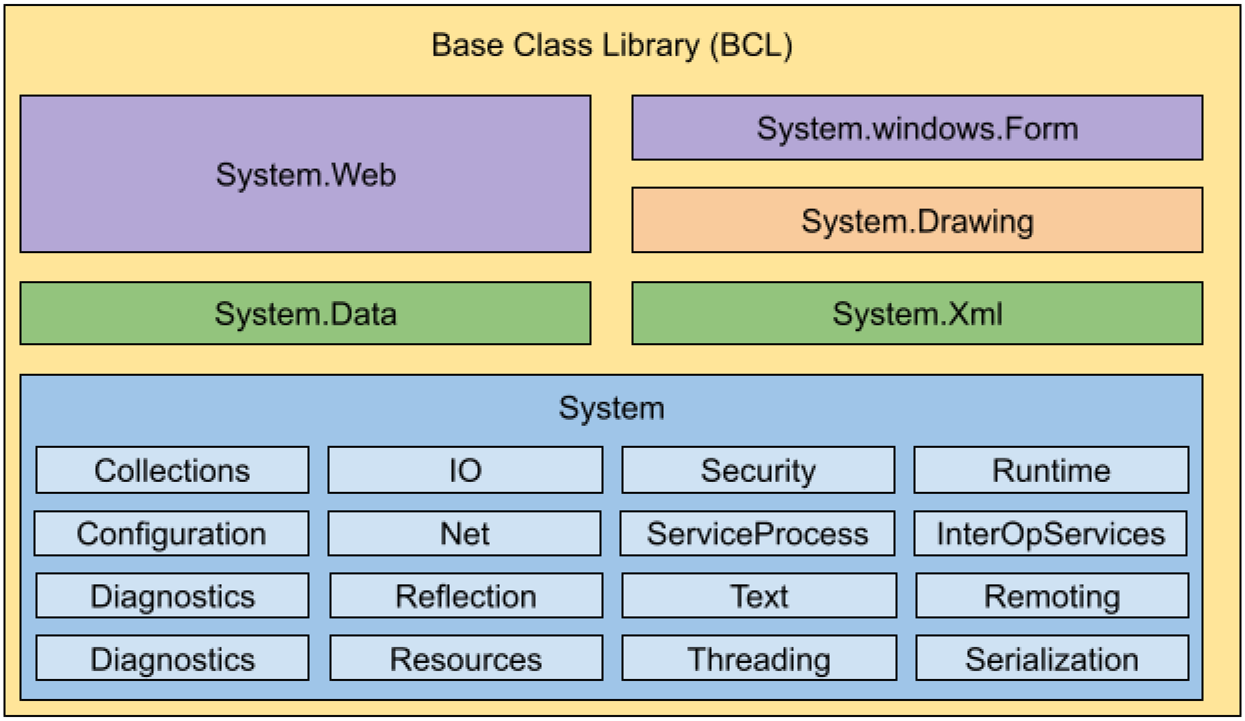
5. It acts as an interface between the framework and operating system


Source Code (.NET Technologies)
.NET Language Compiler
MSL & Meta Data (Portable Executable FIle)
CLR (JIT)
Native Code

Conversion of Source Code into Native Code

iii. Framework Class Library (FCL)

1. Provides a set of pre-built functions and classes that can be used to create a wide range of applications
2. This is integrated with the CLR of the .NET Framework and allows writing programs using .NET supporting programming languages such as C#, Visual C++, VB, etc.
3. Contains a huge library of reusable types, classes, interfaces, structures, and enumerated values, which are collectively called types
4. The Base Class Library (BCL) is the core of the FCL and provides basic functionalities



iv. Common Language Specification

1. It contains the specifications for the .NET supported languages and implementation of language integration

v. Common Type System

   1. Provides guidelines for declaring, using, and managing types at runtime and cross-language

vi. Metadata and Assemblies

   1. Metadata is the binary information describing the program, which is either stored in a portable executable file (PE) or in the memory
   2. Assembly is the logical unit consisting of the assembly manifest, type metadata, IL code and a set of resources like image files

vii. Core Languages (WinForms, ASP.NET and ADO.NET)

   1. Windows Forms
      a. Windows Forms contains graphical representation of any window displayed in the application
      b. It is a smart client technology for the .NET Framework, a set of managed libraries that simplify common application tasks such as reading and writing to the file system
   2. ASP.NET and ASP.NET AJAX
      a. ASP.NET is the web development model and AJAX is an extension of ASP.NET for developing and implementing AJAX functionality
      b. ASP.NET AJAX contains the components that allow the developer to update data on a website without a complete reload of the page
      c. It is used to develop websites, web applications and web services
      d. It provides fantastic integration of HTML, CSS and JavaScript, first released in January 2002
   3. ADO.NET
      a. It is the technology used for working with data and databases
      b. It provides access to data sources like SQL server, OLE DB, XML, etc.
      c. ADO.NET allows connection to data sources for retrieving, manipulating and updating data
      d. It is a module of .NET Framework, which is used to establish a connection between application and data sources, data sources can be such as SQL Server and XML
      e. It consists of classes that can be used to connect , retrieve, insert and delete data

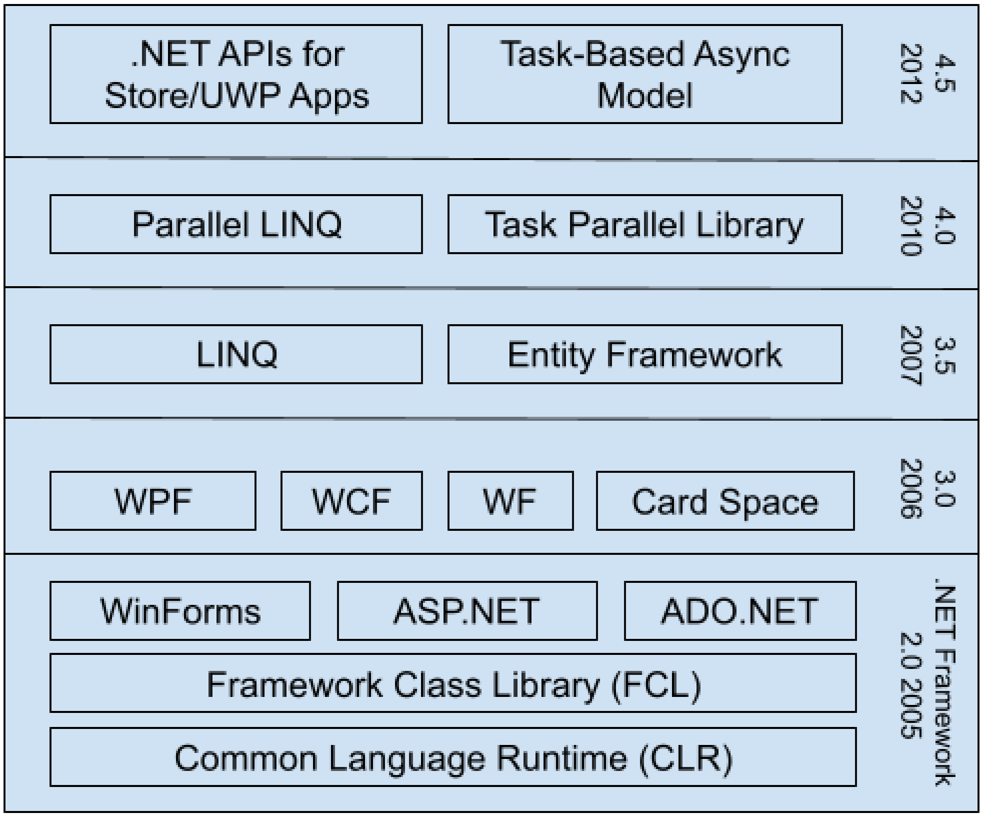
viii.

ix. Other Modules (WCF, WPF, WF, Card Space, LINQ, Entity Framework, Parallel LINQ, Task Parallel Library, etc.)

1. Windows Workflow Foundation (WF)
    a. It helps in building workflow-based applications in windows
    b. It contains activities, workflow runtime, workflow designer and a rules engine
    c. It provides an API, in-process workflow engine, and a rehostable designer to implement long-running processes as workflows within .NET applications
2. Windows Presentation Foundation (WPF)
    a. WPF previously known as ‘Avalon’, was initially released as a part of .NET Framework 3.0 in 2006
    b. Provides a separation between the user interface and the business logic
    c. Helps in developing visually stunning interfaces using documents, media, 2-D & 3-D graphics, animations and more
    d. It is a graphical sub-system by Microsoft for rendering user interfaces in Windows-based applications, it uses DirectX
3. Windows Communication Foundation (WCF)
    a. Technology used for building and executing connected systems
    b. It is a framework for building service-oriented applications
    c. Using WCF, you can send data as asynchronous messages from one service endpoint to another
4. Windows CardSpace
    a. Provides safety for accessing resources and sharing personal information on the internet
5. LINQ
    a. It is a query language introduced in .NET 3.5 framework
    b. It is used to make the query for data sources with C# or Visual Basic Programming languages
    c. It imparts data querying capabilities to .NET languages using a syntax which is similar to the traditional query language SQL
6. Entity Framework
    a. It is an ORM based open-source framework which is used to work with a database using .NET objects
    b. It eliminates a lot of developer effort to handle the database

        c. It is Microsoft’s recommended technology to deal with the database

    7. Parallel LINQ
        a. Parallel LINQ or PLINQ is a parallel implementation of LINQ objects
        b. It combines the simplicity and readability of LINQ and provides the power of parallel programming
        c. It can improve and provide fast speed to execute the LINQ query by using all available computer capabilities



i. Advantages of .NET Framework

i. It supports a variety of programming languages.
    1. Developers can choose the language that best fits their needs and expertise, while still being able to use the same set of libraries and tools provided by the framework
    2. Supports more than 60 programming languages such as C#, F#, VB.NET, J#, VC++, JScript, .NET, APL, COBOL, Perl, Oberon, ML, Pascal ,EIffel, Smalltalk, Python, Cobra, ADA, etc.
    3. Out of more than 60 supported languages, 11 are designed & developed by Microsoft

ii. It supports a variety of application types.

1. It includes libraries and tools for creating applications for desktop, web, mobile, and games, which makes it a versatile choice for developers working on a wide range of projects

iii. It provides a number of features that help to improve the security, reliability and performance of applications. These include features such as type-safety,networking, Code Access Security, Automatic Memory Management, and just-in-time (JIT) compilation, which helps to improve the speed of application execution

iv. It is designed to integrate with the other Microsoft technologies, such as Microsoft SQL Server, Microsoft Sharepoint, and MIcrosoft Office, which can make it easier to build applications that work seamlessly with other Microsoft products.

j. When to use .NET Framework

.NET Framework is to be used for the server application when

i. Application is to be built to run only on WIndows

ii. If application uses .NET Framework technologies which are not available for .NET Core

iii. Applications that are already running on .NET Framework

k. Language Support in .NET Framework

i. The .NET Framework is a language-independent platform that allows interoperability among supported programming languages. The framework supports these programming languages:

1. C#
   a. This is a modern, object-oriented programming language that offers type-safety, scalability support, garbage collection, versioning and other productivity-enhancing features
   b. It is easy to use and can reduce application development time
2. F#
   a. This is an open-source, cross-platform language with object-oriented and imperative programming capabilities.
   b. It is a core functional programming language for .NET
3. Visual Basic
   a. This is a simple language used to build object-oriented apps.
   b. It offers type-safety and uses simple syntax
4. Developers can also use managed C++, IronPython, Visual COBOL, IronRuby and many other languages found in the Visual Studio Languages to code in .NET .

l. .NET Common Language Runtime (CLR)

i.

m. .NET Framework Class Library (FCL)

i.

→ .NET Core

a. Announced in 2014, but officially launched in June 2016
b. .NET Framework has many advantages, but one of the biggest flaws was the ability to work on other platforms rather than WIndows. To fix this, Microsoft re-built .NET Framework as .NET Core
c. It is a cross-platform version of .NET Framework that is introduced to overcome some limitations of the .NET framework, while at the same time, .NET kept getting updates and features that would help it stay advanceand stay preferable by the developers
d. It is lightweight, fast and suitable for building cross-platform applications that can run on WIndows, macOS, and Linux.
e. .NET Core is the recommended choice for new projects
f. It is an open-source platform for building desktop, web, and mobile applications that can be run natively on any operating system
g. It includes tools, libraries and languages that support modern, scalable and high-performance software development
h. Advantages of .NET Core
    i. Ease of Development
        1. It includes many tools that make work easier for developers, e.g. using Visual Studio Suite, developers can write code faster, collaborate efficiently and test and fix their code efficiently
        2. The reusability of code between implementations reduces the cost of development
    ii. High-Performance Applications
        1. .NET Applications provide faster response times and require less computing power
        2. .NET Applications have strong built in

i. When to use .NET Core

.NET Core is to be used for the server application when

i. There is Cross-Platform need
ii. Targeting microservices or using Docker containers
iii. Need for high performance and scalable systems
iv. Side by side need of .NET versions per applications

→ .NET 5 and beyond

a. Microsoft decided to simplify the naming and converge the .NET Core and .NET Framework into a single platform called .NET 5

b. This new platform is fully supported and is the recommended choice for new projects

→ .NET Core vs. .NET Framework

Based on	.NET Core	.NET Framework
Open-Source	.NET Core is open source	Only certain components of .NET Framework are open source
Cross-Platform	Works on principle of “Build Once, Run Anywhere”, it is compatible with various OS	Although it was developed to support softwares & applications on all OS, but .NET Framework is compatible with windows OS
Application Models	.NET Core doesn’t support desktop application development, rather it focuses on web, windows, mobile & windows store including ASP.NET, and Windows Universal Apps	.NET Framework is used for the development of both desktop & web applications as well as it supports Windows Forms & WPF applications
Installation	.NET Core is packaged & installed independently of the underlying OS	.NET Framework is installed as a single package for Windows OS
Support for Microservices & REST services	.NET Core supports development & implementation of microservices, User has to create a REST API for its implementation	.NET Framework doesn’t support the development & implementation of microservices, but it supports the REST API services
Performance & Scalability	.NET Core offers high performance & scalability	.NET Framework is less effective in comparison to .NET Core in terms of performance & scalability of applications
Compatibility	.NET Core is compatible with various OS	.NET Framework is only compatible with Windows OS
Android Development	.NET Core is compatible with open-source mobile	.NET Framework does not support any framework for

	application platforms i.e. Xamarin, through the .NET Standard Library, Developers use Xamarin's tools to configure the mobile apps for specific mobile devices	Mobile Development
Packaging & Shipping	.NET Core is shipped as collection of nugget packages	All the libraries of .NET Framework are packaged and shipped together
Deployment Model	Whenever the updated version of .NET Core is initiated, it is updated instantly on one machine at a time, thereby getting updated in new directories /folders in the existing application without affecting it, making it a flexible deployment model	When updated version of .NET Framework is released, it is first deployed on the Internet Information Server (IIS) only
Support	.NET Core has support for microservices	.NET Framework doesn't support creation and microservices
WCF Services	.NET Core has no support for WCF services	.NET Framework has excellent support for WCF services
REST APIs	Supports REST API	Also supports REST services
CLI Tools	.NET Core provides light-weight editors and command-line tools for all supported platforms	.NET framework is heavy for Command Line Interface and developers prefer to work on the lightweight Command Line Interface
Security	.NET Core does not have features like Code Access Security	Code Access Security feature is present in .NET Framework

→ App Models

The common app models for building software applications in the .NET Framework include

a. Windows Forms :
    i. This model is used to develop form-based applications deployed on end-user services
b. ASP.NET :
    i. This is used to build applications that interact with databases, such as Microsoft SQL Server and Oracle
c. ADO.NET
    i. This model is used to develop web-based applications that can run on browsers

→ ASP Technology

a. ASP and ASP.NET are server side technologies
b. Both Technologies enable computer code to be executed by an Internet Server
c. When a browser requests an ASP or ASP.NET file, the ASP engine reads the file, executes any code in the file, and returns the result to the browser

→ ASP

a. ASP stands for Active Server Pages, also known as Classic ASP
b. It was introduced in 1998 as Microsoft's first server side scripting language
c. Classic ASP Pages have extension .asp and are normally written in VBScript

→ ASP.NET

a. ASP.NET is a web application framework designed and developed by Microsoft
b. It is an open-source and a subset of .NET Framework and successor of the classic ASP(Active Server Pages)
c. With .NET Framework v1.0, ASP.NET was first released in January 2002
d. ASP.NET pages have extension .aspx and are normally written in C#
e. Before .NET and ASP.NET was released, classic ASP was used for developing web applications and services
f. ASP.NET 4.6 is the latest official version of ASP.NET
g. ASP.NET 5 was expected to be an important redesign of ASP.NET, but its development was stopped in favor of ASP.NET Core
h. ASP.NET is a part of .NET Framework used to create dynamic web pages
i. It simplifies the tasks of development, debugging, and deployment of web applications
j. Important functionality of ASP.NET is that developers have access to all of the .NET classes and features
k. Applications developed by ASP.NET are largely component based and built on top of Common Language Runtime (CLR) and can be written in any of the .NET languages

→ Programming Models in ASP.NET

a. ASP.NET provides a number of programming models including the following
    i. ASP.NET MVC :
        1. MVC stands for Model-View-Controller and allows to building of web pages according to Model, View and Controller design pattern
        2. ASP.NET MVC is being merged into the new ASP.NET Core
    ii. ASP.NET Web Pages :
        1. Allows adding dynamic code and data access directly inside HTML
        2. ASP.NET Web Pages is an SPA model (Single Page Application). The SPA model is quite similar to PHP and Classic ASP
        3. ASP.NET Web Pages is being merged into the new ASP.NET Core
    iii. ASP.NET Web Forms :
        1. Allows building modular pages out of components with UI events
    iv. ASP.NET Web API :
        1. Allows developing web APIs on top of .NET Framework

→ ASP.NET Web Forms Model

a. ASP.NET web forms extend the event-driven model of interaction to the web applications
b. The browser submits a web form to the web server and the server returns a full markup page or HTML page in response
c. All client side user activities are forwarded to the server for stateful processing. The server processes the output of the client actions and triggers the reactions
d. HTTP is a stateless protocol and ASP.NET framework helps in storing the information regarding the state of the application, which consists of
    i. Page State
        1. Page State is the state of the client, i.e. the content of the various input fields in the web form
    ii. Session State
        1. The session state is the collective information obtained from various pages, the user visited and worked with i.e. the overall session state
e. E.g. : user adds items to shopping cart, Items are selected from a page, say the items page, and the total collected items and price are shown on a different page, say the cart page. Only HTTP cannot keep track of all the information coming from various pages. ASP.NET session state and server side infrastructure keeps track of the information collected globally over a session.
f. The ASP.NET runtime carries the page state to and from the server across page requests while generating ASP.NET runtime codes, and incorporates the state of the server side components in hidden fields. This way, the server becomes aware of all overall application state and operates in a two-tiered connected way

→ ASP.NET Component Model

a. ASP.NET component model provides various building blocks of ASP.NET pages, which is basically an object model, which describes
    i. Server side counterparts of almost all HTML elements or tags, such as <form> and <input>
    ii. Server controls, which help in developing complex user-interface. E.g. the Calendar control or the GridView Control
b. ASP.NET works on .NET Framework and contains all web-related functionalities
c. An ASP.NET Web Application is made up of pages
d. When a user requests an ASP.NET page, the IIS delegates the processing of the web page to the ASP.NET runtime system
e. The ASP.NET runtime transforms the .aspx page into an instance of a class, which inherits from the base class page of .NET framework. Therefore, each ASP.NET page is an object and all its components i.e. the server-side controls are also objects

I. ASP vs. ASP.NET

S. No.	ASP	ASP.NET
1.	ASP is the interpreted language	ASP.NET is the compiled language
2.	ASP uses ADO (ActiveX Data Object) technology to connect and work with databases	ASP.NET uses ADO.NET to connect and work with databases
3.	ASP is partially object-oriented	ASP.NET is fully object oriented
4.	It does not have a facility to separate design from programming logic	It has an option of code containment
5.	ASP pages have file extension .asp	ASP.NET pages have file extension .aspx
6.	ASP does not have the concept of inheritance	ASP.NET inherits the class written in code behind
7.	ASP pages use scripting language	ASP.NET uses a full-fledged programming language
8.	Error handling is very poor in ASP	Error handling is very good in ASP.NET
9.	Debugging is very difficult because the ASP scripts are interpreted	Debugging is easy in ASP.NET
10.	ASP is not a configurable framework	Web.config file is used to configure ASP.NET Framework
11.	ASP has four built-in classes i.e.,	ASP.NET has more than 2000 built-in

	Request, Response, Session and Application	classes
12.	Custom controls can not be achieved by ASP	Custom Control can be achieved by ASP.NET using @register directive

m. .NET vs. ASP.NET

S.No.	.NET	ASP.NET
1.	.NET is a software framework aimed to develop WIndows, Web, and Server based applications	ASP.NET is a main tool present in the .NET Framework and aimed at simplifying the creation of dynamic web pages
2.	Server side and Client side application development can be done using .NET framework	You can only develop Server side Web Applications using ASP.NET as it is integrated with .NET Framework
3.	Mainly used to make business applications in the Windows platform	It is used to make dynamic Web pages and Websites using .NET languages
4.	Its programming can be done using any language with CIL (Common Intermediate Language) compiler	Its programming can be done using any .NET compliant language

→ ADO.NET

a. ADO stands for ActiveX Data Objects
b. ADO.NET is a set of classes that expose data access services for .NET Framework programmers
c. Provides a rich set of components for creating distributed, data-sharing applications
d. Is an integral part of the .NET Framework, providing access to relational XML, and application data
e. Provides a bridge between the front end controls and backend database
f. ADO.NET objects encapsulate all the data access operations and the controls interact with these objects to display data, thus hiding the details of the movement of data

g. Provides consistent access to data sources such as SQL Server and XML, and to data sources exposed through OLE DB and ODBC
h. ADO.NET classes are found in System.Data.dll , and are integrated with the XML classes found in System.Xml.dll
i. Provides a functionality to developers who wrote managed code similar to the functionality provided to native Component Object Model (COM) developers by ActiveX Data Objects (ADO)
j. It is recommended to use ADO.NET, instead of ADO, for accessing data in your .NET applications
k.

as