

1. What is asp.net life cycle?

Ans: Life Cycle Events, like:

PreInit

-The properties like IsPostBack have been set at this time.

-This event will be used when we want to:

-Set master page dynamically.

-Set theme dynamically.

-Read or set profile property values.

-This event is also preferred if want to create any dynamic controls.

Init

-Raised after all the controls have been initialized with their default values and any skin settings have been applied.

-Fired for individual controls first and then for page.

LoadViewState

-Fires only if IsPostBack is true.

-Values stored in HiddenField with id as _ViewState decoded and stored into corresponding controls.

LoadPostData

Some controls like:

-Fires only if IsPostBack is true.

-Some controls like Textbox are implemented from

IPostBackDataHandler and this fires only for such controls.

-In this event page processes postback data included in the request object pass it to the respective controls.

PreLoad

-Used only if want to inject logic before actual page load starts.

Load

-Used normally to perform tasks which are common to all requests, such as setting up a database query.

Control events

-This event is fired when IsPostBack is true.

-Use these events to handle specific control events, such as a Button control's Click event or a TextBox control's TextChanged event.

PreRender - Raised after the page object has created all the controls that are required for rendering which includes child controls and composite controls.

-Use the event to make final changes to the contents of the page or its controls before the values are stored into the viewstate and the rendering stage begins.

-Mainly used when we want to inject custom JavaScript logic.

SaveViewState

-All the control values that support viewstate are encoded and stored into the viewstate.

Render - Generates output (HTML) to be rendered at the client side.

-We can add custom HTML to the output if we want here.

Unload

-Fired for individual controls first and then for page.

-Used to perform cleanup work like closing open files and database connections.

2. How the request is handled by IIS ?

Ans: We give an URL to an aspx page in the browser address bar and press enter. What happens next? We get the response in terms of rendered HTML but how?

-We are requesting something from the browser, which means indirectly we are requesting something from the Web Server that means IIS. IIS, based on the file extension, decides which ISAPI extension can serve the request. And in case of ASP.Net (.aspx) it will be aspnet_isapi_dll so the request is passed to it for processing.

-When the first request comes to the website, An application domain is created by the ApplicationManager class where exactly the website runs, and which creates an isolation between 2 web applications. Within the application domain an instance of the HostingEnvironment class is created which provides access information about the application such as the name of the folder where the application is stored.

-Next ASP.Net creates core objects like HttpContext, HttpRequest, HttpResponse.

-Finally the application starts by creating an instance of the HttpApplication Class (which can be reused for multiple requests to maximize performance).

3. What is Difference between Session and Cookies ?

Ans: The basic and main difference between cookie and session is that cookies are stored in the user's browser but sessions can't store in user's browser.

This specifies which is best used for.

- A cookie can keep all the information in the client's browser until deleted. If a person has a login and password, this can be set as a cookie in their browser so they do not have to re-login to your website every time they visit. You can store almost anything in a browser cookie.

- Sessions are not reliant on the user allowing a cookie. They work like a token in the browser which allowing access and passing information while the user has opened his browser. The problem in sessions is when you close the browser the session will automatically lost. So, if you had a site requiring a login, this couldn't be saved as a session but it can be saved as a cookie, and the user has to re-login every time they visit. cookies are nothing but a small piece

of information on the client machine. before we create a cookies we should check whether the cookies are allowed at the browser side. They are limited in a size 4k. (They are 2 types of cookies peristant cookie , and session cookies)

Sessions cookies are stored in a server memory during the client browser session. When the browser is closed the session cookies are lost.

4. Advantages and disadvantages of Session?

Ans: Following are the basic advantages and disadvantages of using session. I have described in details with each type of session at later point of time.

Advantages:

- It helps maintain user state and data all over the application.

- It is easy to implement and we can store any kind of object.

- Stores client data separately.

- Session is secure and transparent from the user.

Disadvantages:

- Performance overhead in case of large volumes of data/user, because session data is stored in server memory.

-Overhead involved in serializing and de-serializing session data, because in the case of StateServer and SQLServer session modes, we need to serialize the objects before storing them. Besides these, there are many advantages and disadvantages of session that are based on the session type. I have discussed all of them in the respective sections below.

5. What is state management?

Ans: State management is the process by which you maintain state and page information over multiple requests for the same or different pages.

6. Http is stateless, What does this mean?

Ans: Stateless protocol is a communications protocol that treats each request as an independent transaction that is unrelated to any previous request so that the communication consists of independent pairs of requests and responses.

7. What is Session?

Ans: We know that Http is stateless, means when we open a webpage and fill some information and then move to next page then the data which we have entered will lost. It happed do to Http protocol stateless nature. So here session come into existence, Session provide us the way of storing data in server

memory. So you can store your page data into server memory and retrieve it back during page postbacks.

8. What are the Advantage and disadvantage of Session?

Ans: Advantages:

- Session provide us the way of maintain user state/data.

- It is very easy to implement.

- One big advantage of session is that we can store any kind of object in it. :eg, datatabe, dataset.. etc

- By using session we don't need to worry about data collesp, because it store every client data separately.

- Session is secure and transparent from the user.

Disadvantages:

- Performance overhead in case of large volumes of data/user, because session data is stored in server memory.

- Overhead involved in serializing and de-serializing session data, because in the case of StateServer and

- SQLServer session modes, we need to serialize the objects before storing them.

9. What is Session ID in Asp.net?

Ans: Asp.Net use 120 bit identifier to track each session. This is secure enough and can't be

reverse engineered. When client communicate with server, only session id is transmitted, between them. When client request for data, ASP.NET looks on to session ID and retrieves corresponding data.

10. By default where the sessions ID's are stored ?

Ans: By default, the unique identifier for a session is stored in a non-expiring session cookie in the browser. You can specify that session identifiers not be stored in a cookie by setting the cookieless attribute to true in the sessionState configuration element. We can also configure our application to store it in the url by specifying a "cookieless" session The ASP Session cookie has this format:
ASPSESSIONIDACSSDCCC=APHELKLDNMKNIOJ
ONJACDHFN

11. Where does session stored if cookie is disabled on client's machine?

Ans: If you want to disable the use of cookies in your ASP.NET application and still make use of session state, you can configure your application to store the session identifier in the URL instead of a cookie by setting the cookieless attribute of the sessionState configuration element to true, or to UseUri, in the Web.config file for your application.

The following code example shows a Web.config file that configures session state to use cookieless session identifiers.

Code:

```
<configuration>
  <system.web>
    <sessionState
      cookieless="true"
      regenerateExpiredSessionId="true"
      timeout="30" />
  </system.web>
</configuration>
```

12. Can you describe all the property set in web.config under session state?

Ans:

Code:

```
<configuration>
  <sessionstate
    mode="inproc"
    cookieless="false"
    timeout="20"
    sqlconnectionstring="data
source=127.0.0.1;user
id=<user id>;password=<password>"
    server="127.0.0.1"
    port="42424"
  />
```

</configuration>

Mode: The mode setting supports three options: inproc, sqlserver, and stateserver. As stated earlier, ASP.NET supports two modes: in process and out of process. There are also two options for out-of-process state management: memory based (stateserver), and SQL Server based (sqlserver). We'll discuss implementing these options shortly.

Cookieless: The cookieless option for ASP.NET is configured with this simple Boolean setting.

Timeout: This option controls the length of time a session is considered valid. The session timeout is a sliding value; on each request the timeout period is set to the current time plus the timeout value

Sqlconnectionstring: The sqlconnectionstring identifies the database connection string that names the database used for mode sqlserver.

Server: In the out-of-process mode stateserver, it names the server that is running the required Windows

NT service: ASPState.

Port: The port setting, which accompanies the server setting, identifies the port number that corresponds to the server setting for mode stateserver.

13. What are Session Events?

Ans: There are two types of session events available in ASP.NET:

Session_Start

Session_End

You can handle both these events in the global.asax file of your web application. When a new session initiates, the session_start event is raised, and the Session_End event raised when a session is abandoned or expires.

14. How you can disable session?

Ans: If we set session Mode="off" in web.config, session will be disabled in the application. For this, we need to configure web.config the following way:

Code:

<configuration>

<sessionstate Mode="off"/>

</configuration>

15. If I have more than one version of one assemblies, then how will I use old version (how/where to specify version number?) in my application?

Ans: The version number is stored in the following format: The assembly manifest can then contain a reference to which version number we want to use.

16. How do you create threading in .NET? What is the namespace for that?

Ans: System.Threading;

//create new thread using the thread class's constructor

Thread myThread = new Thread(new ThreadStart (someFunction));

17. What do you mean by Serialize and MarshalByRef?

Ans:

-Serialization is the act of saving the state of an object so that it can be recreated (i.e deserialized) at a later date.

-The MarshalByRef class is part of the System.Runtime.Remoting namespace and enables us to access and use objects that reside in different application domains. It is the base class for objects that need to communicate across application domains. MarshalByRef objects are accessed directly within their own application domain by using a proxy to communicate. With MarshalByValue the a copy of the entire object is passed across the application domain

18. What is the difference between Array and LinkedList?

Ans: An array is a collection of the same type. The size of the array is fixed in its declaration. A linked list is similar to an array but it doesn't have a limited size.

19. What is Asynchronous call and how it can be implemented using delegates?

Ans: A synchronous call will wait for a method to complete before program flow is resumed. With an asynchronous call the program flow continues whilst the method executes.

```
//create object  
SomeFunction objFunc = new SomeFunction();
```

```
//create delegate  
SomeDelegate objDel = new  
SomeDelegate(objFunc.FunctionA);
```

```
//invoke the method asynchronously (use  
interface IAsyncResult)  
IAsyncResult asynchCall =  
SomeDelegate.Invoke();
```

20. How to create events for a control? What is custom events? How to create it?

Ans: An event is a mechanism used in a class that can be used to provide a notification when something interesting happens. (typical events in a windows application include: change text in

textbox, double click or click a button, select an item in dropdown box).

A custom event is an event created by the user that other developers can use. For example assuming that we have a CashTransaction class and we have a bank balance property in that class. We may want to set-up an event that provides a notification when the bank balance drops below a certain amount. In order to produce an event the process would be roughly as follows:

Create the class for the event derived from EventArgs.

Create a delegate with a return type of void. Create a class containing the method that will activate the event.

Create a class with methods to handle the event.

21. If you want to write your own dot net language, what steps you will you take care?

Ans: We will need to ensure that the high level code is compiled to MSIL (Microsoft intermediate language) so that it can be interpreted by the CLR.

22. Describe the difference between inline and code behind - which is best in a loosely coupled solution?

Ans: The term 'code behind' refers to application code that is not embedded within

the ASPX page and is separated out into a separate file which is then referenced from the ASPX page. Inline code is the traditional ASP architectural model where business logic code was embedded within the ASP page. Separating the business logic code from the presentation layer offers several

Advantages:

- It allows graphic designers and web developers to work on the presentation layer whilst the application developers concentrate on the business logic.

- The codebehind file is compiled as a single dll increasing the efficiency of the application,

- The codebehind model offers a true OO development platform,

- It speeds up development time as it allows developers to fully maximise the features of the .NET framework such as Caching, ViewState, Session, Smart Navigation etc.

- Code is much easier to maintain and susceptible for change.

- The compiler and VS.NET provides much better support for error checking, intellisense and debugging when using the code behind model.

23. How dot net compiled code will become platform independent?

Ans: The raison d'être for .NET was to cater for multiple languages on a single windows platform whereas the aim of Java was to be a

single language on multiple platforms. The only way that .NET can be platform independent is if there is a version of the .NET framework installed on the target machine.

24. Without modifying source code if we compile again, will it be generated MSIL again?

Ans: No.

25. How does you handle this COM components developed in other programming languages in .NET?

Ans: use TlbImp.exe to import the COM types into your .NET project. If no type library for the COM component then use System.Runtime.InteropServices use RegAsm.exe to call a .NET developed component in a COM application.

26. How CCW (Com Callable Wrapper) and RCW (Runtime Callable Wrappers) works?

Ans:

CCW: When a COM application calls a .NET object the CLR creates the CCW as a proxy since the COM application is unable to directly access the .NET object.

RCW: When a .NET application calls a COM object the CLR creates the RCW as a proxy since the .NET application is unable to directly access the .COM object.

27. What are the new three features of COM+ services, which are not there in COM (MTS)?

Ans:

Role based security.

Neutral apartment threading.

New environment called context which defines the execution environment

28. What are the differences between COM architecture and .NET architecture?

Ans: .Net architecture has superseded the old COM architecture providing a flexible rapid application development environment which can be used to create windows, web and console applications and web services. .NET provides a powerful development environment that can be used to create objects in any .NET compliant language.

.NET addresses the previous problems of dll hell with COM by providing strongly named assemblies and side-by-side execution where two assemblies with the same name can run on the same box.

29. Can we copy a COM dll to GAC folder?

Ans: No. It only stores .NET assemblies.

30. What is Shared and Repeatabe Inheritance?

Ans: Shared Inheritance-: Shared Inheritance-: It is multiple times using same class. The mechanism of deriving a new class from an

existing class is called inheritance. Shared inheritance introduces a new opportunity of ambiguity and additional implementation complexity. Assume D inherits from B and C, both of which inherits from A. Here A is shared. Single copy made from both derived classes is called shared inheritance.

31. Can you explain what inheritance is and an example of when you might use it?

Ans: Inheritance is a fundamental feature of any OO language. It allows us to inherit the members and attributes from a base class to a new derived class.

This leads to increased code reusability and also makes applications easier to develop, maintain and extend as the new derived class can contain new features not available in the base class whilst at the same time preserving the attributes inherited from the base class.

32. How can you write a class to restrict that only one object of this class can be created (Singleton class)?

Ans: Use the singleton design pattern.

public sealed class Singleton

{

static readonly Singleton Instance=new Singleton

();

```

static Singleton()
{
}
Singleton()
{
}
public static Singleton Instance
{
    get
    {
        return Instance;
    }
}
}

```

33. What are virtual destructures?

Ans: A constructor cannot be virtual but a destructor may. Use virtual destructors when you want to implement polymorphic tearing down of an object.

34. What is close method? How its different from Finalize and Dispose?

Ans: finalise is the process that allows the garbage collector to clean up any unmanaged resources before it is destroyed.

The finalise method cannot be called directly; it is automatically called by the CLR. In order to allow more control over the release of unmanaged resources the .NET framework

provides a dispose method which unlike finalise can be called directly by code.

Close method is same as dispose. It was added as a convenience.

35. What is Boxing and UnBoxing?

Ans: Boxing is the process of converting a value type to a reference type. More specifically it involves encapsulating a copy of the object and moving it from stack to heap.

Unboxing is the reverse process.

36. What is check/uncheck?

Ans:

checked: used to enable overflow checking for arithmetic and conversion functions.

unchecked: used to disable overflow checking for arithmetic and conversion functions

37. What is the use of base keyword? Tell me a practical example for base keyword's usage?

Ans:

The base keyword is used to access members of the base class from within a derived class:

- * Call a method on the base class that has been overridden by another method.

- * Specify which base-class constructor should be called when creating instances of the derived class. A base class access is permitted only in a constructor, an instance method, or an instance property accessor. It is an error to use the base

keyword from within a static method.

Example: In this example, both the base class, Person, and the derived class, Employee, have a method named GetInfo. By using the base keyword, it is possible to call the GetInfo method on the base

class, from within the derived class.

// keywords_base.cs

// Accessing base class members

```
using System;
```

```
public class Person
```

```
{
```

```
    protected string ssn = "444-55-6666";
```

```
    protected string name = "John L. Malgraine";
```

```
    public virtual void GetInfo()
```

```
{
```

```
    Console.WriteLine("Name: {0}", name);
```

```
    Console.WriteLine("SSN: {0}", ssn);
```

```
}
```

```
}
```

```
class Employee: Person
```

```
{
```

```
    public string id = "ABC567EFG";
```

```
    public override void GetInfo()
```

```
{
```

```
    // Calling the base class GetInfo method:
```

```
    base.GetInfo();
```

```
    Console.WriteLine("Employee ID: {0}", id);
```

```
}
```

```

}
class TestClass {
public static void Main()
{
Employee E = new Employee();
E.GetInfo();
}
}

```

38. Difference Between Query String and Session

Ans:

Querystring:

-Querystring is client side state management technique.

-Querystring data is page specific i.e. can be accessed in that page only.

-Querystring data is visible to user and can be seen in browser url.

-Data is not secured and can be altered hence insensitive data is stored in querystring.

-Querystring has constraint of Maxlength.

Session:

-Session is server side state management technique.

-Session data can be accessed throughout the session.

-Session data is not visible to user.

-Data is secured hence sensitive data such as user information is stored.

-Session does not have such constraint.

39. Difference between Query string and Cookies

Ans:

-cookies is a text file stored on client machine when we surf anything on internet by the server automatically we don't have to create it

-query string is used to transfer data from 1 page to another but this is not safe as it shows in url what data we're sending to any site and see url after question mark that is url

Cookies: - Cookies are little pieces of information that a server stores on a browser.

They are of two types

1. Temporary cookie

2. Persistent cookie

Temporary cookie: - They are also known as session cookies. These are volatile in nature. When the browser is shutdown they are erased.

Persistent cookie: - These may be called as permanent cookies. These are especially saved in files. It may remain for a month or year.

Properties of cookies

Some properties of cookie

Name: - represent the name of cookie.

Name value: - represent a collection of key values of cookie

Domain: - represent the domain associated with a specific cookie.

Path: - the path associated with a cookie.

Expires: - expired time of cookie.

Hashkey: - identifies whether the cookie is a cookie dictionary.

Secure: - specifies whether the cookie is to be sent in an encrypted connection or not

Query string : Query string is the limited way to pass information to the web server while

Transferring from one page to another page.

This information is passed in url of the request.

see below the code sample

```
//Retrieving values from query string
```

```
String name;
```

```
//Retrieving from query string
```

```
name = Request.Param["umar"].ToString();
```

But remember that many browsers impose a limit of 255 characters in query strings. You need to use HTTP-Get method to post a page to server otherwise query string values will not be available.

40. How can we identify that the Page is Post Back?

Ans: Page object has an "IsPostBack" property, which can be checked to know that is the page posted back.

41. What is the lifespan for items stored in ViewState?

Ans: The items stored in ViewState live until the lifetime of the current page expires including the postbacks to the same page

42. What is AutoPostBack?

Ans: If you want a control to postback automatically when an event is raised, you need to set the AutoPostBack property of the control to True

43. Why do you use the App_Code folder in ASP.NET?

Ans: The App_Code folder is automatically present in the project. It stores the files, such as classes, typed data set, text files, and reports. If this folder is not available in the application, you can add this folder. One of the important features of the App_Code folder is that only one dll is created for the complete folder, irrespective of how many files it contains.

44. In which event of page cycle is the ViewState available?

Ans: After the Init() and before the Page_Load().

45. How long the items in ViewState exists?

Ans: They exist for the life of the current page.

46. Where the viewstate is stored after the page postback?

Ans: ViewState is stored in a hidden field on the page at client side. ViewState is transported to the client and back to the server, and is not stored on the server or any other external source.

47. What are the different Session state management options available in ASP.NET?

Ans:

-In-Process

-Out-of-Process.

In-Process stores the session in memory on the web server.

Out-of-Process Session state management stores data in an external server. The external server may be either a SQL Server or a State Server. All objects stored in session are required to be serializable for Out-of-Process state management.

48. What is the difference between web config and machine config?

Ans: Web config file is specific to a web application where as machine config is specific to a machine or server. There can be multiple web config files into an application where as we can have only one machine config file on a server.

49. What are the different types of cookies in ASP.NET?

Ans: Session Cookie – Resides on the client machine for a single session until the user does not log out.

Persistent Cookie – Resides on a user's machine for a period specified for its expiry, such as 10 days, one month, and never.

50. What is the difference between ExecuteScalar and ExecuteNonQuery?

Ans: ExecuteScalar returns output value where as ExecuteNonQuery does not return any value but the number of rows affected by the query. ExecuteScalar used for fetching a single value and ExecuteNonQuery used to execute Insert and Update statements.

1. What is ASP?

ANS: Active Server Pages (ASP), also known as Classic ASP, is a Microsoft's server-side technology, which helps in creating dynamic and user-friendly Web pages. It uses different scripting languages to create dynamic Web pages, which can be run on any type of browser. The Web pages are built by using either VBScript or JavaScript and these Web pages have access to the same services as Windows application, including ADO (ActiveX Data Objects) for database access, SMTP (Simple Mail Transfer Protocol) for e-mail, and

the entire COM (Component Object Model) structure used in the Windows environment. ASP is implemented through a dynamic-link library (asp.dll) that is called by the IIS server when a Web page is requested from the server.

2. What is ASP.NET?

ANS: ASP.NET is a specification developed by Microsoft to create dynamic Web applications, Web sites, and Web services. It is a part of .NET Framework. You can create ASP.NET applications in most of the .NET compatible languages, such as Visual Basic, C#, and J#. The ASP.NET compiles the Web pages and provides much better performance than scripting languages, such as VBScript. The Web Forms support to create powerful forms-based Web pages. You can use ASP.NET Web server controls to create interactive Web applications. With the help of Web server controls, you can easily create a Web application.

3. What is the basic difference between ASP and ASP.NET?

ANS: The basic difference between ASP and ASP.NET is that ASP is interpreted; whereas, ASP.NET is compiled. This implies that since ASP uses VBScript; therefore, when an ASP page is executed, it is interpreted. On the other hand, ASP.NET uses .NET languages, such as

C# and VB.NET, which are compiled to Microsoft Intermediate Language (MSIL).

4. In which event are the controls fully loaded?

ANS: Page load event guarantees that all controls are fully loaded. Controls are also accessed in Page_Init events but you will see that view state is not fully loaded during this event

5. How can we identify that the Page is Post Back?

ANS: Page object has an "IsPostBack" property, which can be checked to know that is the page posted back.

6. What is the lifespan for items stored in ViewState?

ANS: The items stored in ViewState live until the lifetime of the current page expires including the postbacks to the same page.

7. What is the concept of Postback in ASP.NET?

ANS: A postback is a request sent from a client to server from the same page user is already working with.

ASP.NET was introduced with a mechanism to post an HTTP POST request back to the same page. It's basically posting a complete page back to server (i.e. sending all of its data) on same page. So, the whole page is refreshed.

Another concept related to this approach is "Callback" that is also asked sometimes during a technical interview question. Click here to understand Postback Vs Callback in ASP.NET.

8. Difference between ASP.NET WebForms and ASP.NET MVC?

ANS: ASP.NET Web Forms uses Page controller pattern approach for rendering layout. In this approach, every page has its own controller i.e. code-behind file that processes the request. On the other hand, ASP.NET MVC uses Front Controller approach. In this approach a common controller for all pages, processes the requests.

9. Please briefly explain ASP.NET Page life Cycle?

ANS: ASP.NET page passes through a series of steps during its life cycle. Following is the high-level explanation of life cycle stages/steps.

Initialization: Controls raise their Init event in this stage. Objects and variables are initialized for complete lifecycle of request.

LoadViewState: is a post back stage and loads the view state for the controls that enabled its view state property.

LoadPostBackData: is also a post back stage and loads the data posted for the controls and update them.

Load: In this stage page as well as all the controls raise their Load event. Till this stage all the controls are initialized and loaded. In most of the cases, we are coding this event handler. RaisePostBackEvent: is again a postback stage. For example, it's raise against a button click event. We can easily put our code here to perform certain actions.

SaveViewState: Finally, controls state is saved in this stage before Rendering HTML.

Render: This is the stage where HTML is generated for the page.

Dispose: Lastly, all objects associated with the request are cleaned up.

10. What is the difference between custom controls and user controls?

ANS: Custom controls are basically compiled code i.e. DLLs. These can be easily added to toolbox, so it can be easily used across multiple projects using drag and drop approach. These controls are comparatively hard to create. But User Controls (.ascx) are just like pages (.aspx). These are comparatively easy to create but tightly couple with respect to User Interface and code. In order to use across multiple projects, we need to copy and paste to the other project as well.

What is the concept of view state in ASP.NET? As in earlier question, we understood the concept of postback. So, in order to maintain

the state between postbacks, ASP.NET provides a mechanism called view state. Hidden form fields are used to store the state of objects on client side and returned back to server in subsequent request (as postback occurs). Difference between Response.Redirect and Server.Transfer?

In case of Response.Redirect, a new request is generated from client-side for redirected page. It's a kind of additional round trip. As new request is generated from client, so the new URL is visible to user in browser after redirection.

While in case of Server.Transfer, a request is transferred from one page to another without making a round trip from client. For the end user, URL remains the same in browser even after transferring to another page.

Please briefly explain the usage of Global.asax? Global.asax is basically ASP.NET Application file. It's a place to write code for Application-level events such as Application start, Application end, Session start and end, Application error etc. raised by ASP.NET or by HTTP Modules. There is a good list of events that are fired but following are few of the important events in Global.asax:

Application_Init occurs in case of application initialization for the very first time.

Application_Start fires on application start.

Session_Start fires when a new user session starts

Application_Error occurs in case of an unhandled exception generated from application.

Session_End fires when user session ends.

Application_End fires when application ends or time out.

11. What are the different types of Validation controls in ASP.NET?

ANS: In order to validate user input, ASP.NET provides validation server controls. All validation controls inherits from BaseValidator class which contains the common validation properties and methods like ControlToValidate, Enabled, IsValid, EnableClientScript, ValidationGroup, Validate() etc.

ASP.NET provides a range of validation controls: RequiredFieldValidator validates compulsory/required input.

RangeValidator validates the range. Validates that input falls between the given range values. CompareValidator validates or compares the input of a control with another control value or with a fixed value.

RegularExpressionValidator validates input value against a defined regular expression pattern.

CustomValidator allows to customize the validation logic with respect to our application logic.

ValidationSummary displays all errors on page collectively.

12. What are the types of Authentication in ASP.NET?

ANS: There are three types of authentication available in ASP.NET:

Windows Authentication: This authentication method uses built-in windows security features to authenticate user.

Forms Authentication: authenticate against a customized list of users or users in a database.

Passport Authentication: validates against Microsoft Passport service which is basically a centralized authentication service.

13. What are Session state modes in ASP.NET?

ANS: ASP.NET supports different session state storage options:

In-Process is the default approach. It stores session state locally on same web server memory where the application is running.

StateServer mode stores session state in a process other than the one where application is running. Naturally, it has added advantages that session state is accessible from multiple web servers in a Web Farm and also session state will remain preserved even web application is restarted.

SQLServer mode stores session state in SQL Server database. It has the same advantages as that of StateServer.

Custom modes allows to define our custom storage provider.

Off mode disables session storage.

14. What does the Orientation property do in a Menu control?

ANS: Orientation property of the Menu control sets the display of menu on a Web page to vertical or horizontal.

Originally the orientation is set to vertical.

15. Differentiate between: Client-side and server-side validations in Web pages.

ANS: Client-side validations happens at the client's side with the help of JavaScript and VBScript. This happens before the Web page is sent to the server.

- Server-side validations occurs place at the server side.

16. Authentication and authorization.

ANS: Authentication is the process of verifying the identity of a user using some credentials like username and password while authorization determines the parts of the system to which a particular identity has access.

- Authentication is required before authorization.

For e.g. If an employee authenticates himself with his credentials on a system, authorization will determine if he has the control over just publishing the content or also editing it.

17. What does the .WebPart file do?

ANS: It explains the settings of a Web Parts control that can be included to a specified zone on a Web page.

18. How would you enable impersonation in the web.config file?

ANS: In order to enable the impersonation in the web.config file, take the following steps:

- Include the <identity> element in the web.config file.

- Set the impersonate attribute to true as shown below:

```
<identity impersonate = "true" />
```

19. Differentiate between File-based dependency and key-based dependency.

ANS: In file-based dependency, the dependency is on a file saved in a disk while in key-based dependency, you depend on another cached item.

20. Globalization and localization.

ANS: Globalization is a technique to identify the part of a Web application that is different for different languages and separate it out from the

web application while in localization you try to configure a Web application so that it can be supported for a specific language or locale.

21. Differentiate between a page theme and a global theme?

ANS: Page theme applies to a particular web pages of the project. It is stored inside a subfolder of the App_Themes folder.

- Global theme applies to all the web applications on the web server. It is stored inside the Themes folder on a Web server.

22. What are Web server controls in ASP.NET?

ANS: These are the objects on ASP.NET pages that run when the Web page is requested.

- Some of these Web server controls, like button and text box, are similar to the HTML controls.

- Some controls exhibit complex behavior like the controls used to connect to data sources and display data.

23. Differentiate between a HyperLink control and a LinkButton control.

ANS: A HyperLink control does not have the Click and Command events while the LinkButton control has them, which can be handled in the code-behind file of the Web page.

24. How do Cookies work? Give an example of their abuse.

ANS:

- The server directs the browser to put some files in a cookie. All the cookies are then sent for the domain in each request.

- An example of cookie abuse could be a case where a large cookie is stored affecting the network traffic.

25. What are Custom User Controls in ASP.NET?

ANS:

- These are the controls defined by developers and work similar to other web server controls.

- They are a mixture of custom behavior and predefined behavior.

26. What is Role-based security?

ANS:

- Used in almost all organization, the Role-based security assigns certain privileges to each role.

- Each user is assigned a particular role from the list.

- Privileges as per role restrict the user's actions on the system and ensure that a user is able to do only what he is permitted to do on the system.

28. What are the HTML server controls in ASP.NET?

ANS:

- HTML server controls are similar to the standard HTML elements like those used in HTML pages.

- They expose properties and events for programmatic use.

- To make these controls programmatically accessible, we specify that the HTML controls act as a server control by adding the `runat="server"` attribute.

27. What are the various types of Cookies in ASP.NET?

ANS: There exist two types of cookies in ASP.NET

- Session Cookie – It resides on the machine of the client for a single session and works until the user logs out of the session.

- Persistent Cookie – It resides on the machine of a user for a specified period. This period can be set up manually by the user.

28. How would you turn off cookies on one page of your website?

ANS: This can be done by using the `Cookie.Discard` property.

- It Gets or sets the discard flag set by the server.

- When set to true, this property instructs the client application not to save the Cookie on the hard disk of the user at the end of the session.

29. How would you create a permanent cookie?

ANS: Permanent cookies are stored on the hard disk and are available until a specified expiration date is reached.

- To create a cookie that never expires set its Expires property equal to DateTime.MaxValue.

30. Explain Culture and UICulture values.

ANS:

- Culture value determines the functions like Date and Currency used to format data and numbers in a Web page.

- UICulture value determines the resources like strings or images loaded in a Web application for a Web page.