Questions:

1. What is ASP.Net?
2. What are the advantages of ASP.Net?
3. What is AppDomain?
4. What is Thread of an App domain?
5. What is Worker Process?
6. When an ASP.NET application starts its life cycle?
7. What is ISAPI?
8. What Kind of HTTP Server Is Needed to Run ISAPI?
9. What ISAPI.DLL does?
10. What Is Inetinfo.exe?
11. Describe the role of inetinfo.exe, aspnet_isapi.dll and aspnet_wp.exe in the page loading process.
12. What is Web Server?
13. What is the default execution engine for ASP.NET Web Application when you run from visual studio IDE?
14. What is IIS (Internet Information Server)?
15. What is IIS Manager?
16. What is HTTP Runtime?
17. What is HTTP Handler?
18. Explain about ASP.NET Web Forms.
19. Explain the life cycle of a WebForm.
20. What's the difference between Response.Write() andResponse.Output.Write()?
21. What methods are fired during the page load?
22. What namespace does the Web page belong in the .NET Framework class hierarchy?
23. Where do you store the information about the user's locale?
24. What's the difference between Codebehind="MyCode.aspx.cs" andSrc="MyCode.aspx.cs"?
25. Explain the differences between Server-side and Client-side code?
26. What type of code (server or client) is found in a Code-Behind class?
27. Describe the difference between inline and code behind.
28. What base class do all Web Forms inherit from?
29. How a Control Manages its State?
30. What is View state?

31. What is the lifespan for items stored in ViewState?
32. In what order do the events of an ASPX page execute. As a developer is it important to understand these events?
33. Explain what is EnableViewState property of a ASP.NET Server Control?
34. How control raises events?
35. What will be the effect when EnableViewState="False" for a control whose events are handled on server?
36. When during the page processing cycle is ViewState available?
37. What is @Page Directive in WebForm?
38. Explain how a web application works.

Questions and Answers:

1. What is ASP.Net?

Answer: ASP.Net is a server side Technology to develop a web based applications. ASP.Net will make use of .Net framework features.

2. What are the advantages of ASP.Net?

Answer: Following are the advantages of ASP.Net:

1) Web application exists in compiled form on the server so the execution speed is faster as compared to the interpreted scripts.
2) ASP.NET makes development simpler and easier to maintain with an event-driven, server-side programming model.
3) Being part of Framework, it has access to all the features of .Net Framework.
4) Content and program logic are separated which reduces the inconveniences of program maintenance.
5) Integrated with ADO.NET.
6) Built-in caching features.

3. What is AppDomain?

Answer: An AppDomain is a logical collection of one or more threads with in a process. If a thread of one App domain wants to access the thread of another app domain, it has to use inter-process communication (IPC). A process can have more than one AppDomain and each AppDomain is independent of other

AppDomains. If a thread in one AppDomain crashes, only other threads of that AppDomain are affected and threads in other AppDomains would continue to run.

4. What is Thread of an App domain?

Answer: Thread is where the instruction will execute.

5. What is Worker Process?

Answer: Worker Process runs the ASP.Net application in IIS. This process is responsible to manage all the request and response that are coming from client system. All the ASP.Net functionality runs under the scope of worker process.

6. When an ASP.NET application starts its life cycle?

Answer: ASP.NET application starts its life cycle when a user requests a page from the server using any browser like internet explorer, Mozilla, Google chrome etc., on web.

7. What is ISAPI?

Answer: ISAPI is nothing but an Internet Server Application Programming Interface. ISAPI is a collection of windows-based web server services which are used to modify and enhance the functionality provided by IIS. ISAPI checks the extension of a file which is requested by the browser and based on that, it loads the httphandler which is responsible for handling the request.

8. What Kind of HTTP Server Is Needed to Run ISAPI?

Answer: To host Web sites, you must have an Internet server that supports the Hypertext Transfer Protocol (HTTP). If you have chosen an ISAPI-compliant Web server (for example, Microsoft Internet Information Server), you can take advantage of server extension DLLs to create small, fast Internet server applications.

9. What ISAPI.DLL does?

Answer: This ISAPI application is going to run within the process address space of IIS which will take over the request of aspx extension from IIS.

10. What Is Inetinfo.exe?

Answer: Inetinfo.exe is the process address space of IIS.

11. Describe the role of inetinfo.exe, aspnet_isapi.dll and aspnet_wp.exe in the page loading process.

Answer: inetinfo.exe is the Microsoft IIS server running, handling ASP.NET requests among other things. When an ASP.NET request is received (usually a file with .aspx extension), the ISAPI filter aspnet_isapi.dll takes care of it by passing the request to the actual worker process aspnet_wp.exe.

12. What is Web Server?

Answer: Web Server is execution engine is responsible for handle all the requests that are coming from clients, process them and provide the responses.

13. What is the default execution engine for ASP.NET Web Application when you run from visual studio IDE?

Answer: When we run our ASP.NET Web Application from visual studio IDE, VS Integrated ASP.NET Engine is responsible to execute all kind of asp.net requests and responses. The process name is "WebDev.WebServer.Exe" which actually take care of all request and response of a web application which is running from Visual Studio IDE.

14. What is IIS (Internet Information Server)?

Answer: It is a windows component, a web server that accepts requests from client browsers and responds with the requested pages.

15. What is IIS Manager?

Answer: A tool to configure and manage IIS. To launch IIS Manager one of the below method can be used.

Control Panel --> Administrative Tool --> IIS Manager

OR

Start --> Run --> inetmgr.exe

16. What is HTTP Runtime?

Answer: The ASP.NET HTTP Runtime is an extensible application runtime that allows developers to change/extend/limit the functionality of ASP.NET. What this means to developers is that if ASP.NET does not provide sufficient functionality in a particular area, or if ASP.NET provides more than required

functionality, then the developer can change the underlying application infrastructure so as to add / remove functionality as required.

17. What is HTTP Handler?

Answer: HTTP Handlers are the logical replacements for ISAPI applications. They can be used to code some complex functionality ranging from downloading and uploading server file to generating images on the fly.

18. Explain about ASP.NET Web Forms.

Answer: Web Forms are an extremely important part of ASP.NET. They are the User Interface (UI) elements which provide the desired look and feel to your web applications. Web Forms provide properties, methods, and events for the controls that are placed onto them.

19. Explain the life cycle of a WebForm.

Answer:

a) Creates an instance of the WebForm class i.e. the dynamic class inherited by framework from the class in code file.
b) For every trip made to the server i.e. for first request or even if the form is submitted (Postback) the new instance the page class is created.
c) Based on the server side tags (runat="server") the controls are instanced and based on tag attributes the control properties are set.
d) Based on the state of the server object for every control, the control renders its part to the overall output of the WebForm rendered to the browser.

20. What's the difference between Response.Write() andResponse.Output.Write()?

Answer: Response.Output.Write() allows you to write formatted output.

21. What methods are fired during the page load?

Answer:

Init() - when the page is instantiated

Load() - when the page is loaded into server memory

PreRender() - the brief moment before the page is displayed to the user as HTML

Unload() - when page finishes loading.

22. What namespace does the Web page belong in the .NET Framework class hierarchy?

Answer: System.Web.UI.Page

23. Where do you store the information about the user's locale?

Answer: System.Web.UI.Page.Culture

24. What's the difference between Codebehind="MyCode.aspx.cs" and Src="MyCode.aspx.cs"?

Answer: Code Behind is relevant to Visual Studio.NET only

25. Explain the differences between Server-side and Client-side code?

Answer: Server-side code executes on the server. Client-side code executes in the client's browser.

26. What type of code (server or client) is found in a Code-Behind class?

Answer: The answer is server-side code since code-behind is executed on the server. However, during the code behind's execution on the server, it can render client-side code such as JavaScript to be processed in the client's browser. But just to be clear, code-behind executes on the server, thus making it server-side code.

27. Describe the difference between inline and code behind.

Answer: Inline code written alongside the html in a page. Code-behind is code written in a separate file and referenced by the .aspx page.

28. What base class do all Web Forms inherit from?

Answer: The Page class.

29. How a Control Manages its State?

Answer:

1. When the page is rendered, every control adds its modified state to the hidden element "__VIEWSTATE".

2. For a given response / page there is only one VIEWSTATE rendered to the browser and all controls add their modified state to it. Ex: The TextBox will add its "ForeColor" and all other properties which are changed programmatically to the "__VIEWSTATE".
3. When the form is submitted from the web browser, the control initializes any one of its properties from the name-value pair submitted on behalf of the HTML representation of the control.

30. What is View state?

Answer: View state is the method that the ASP.NET page framework uses to preserve page and control values between round trips. When the HTML markup for the page is rendered, the current state of the page and values that must be retained during postback are serialized into base64-encoded strings.

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

Answer: Item stored in ViewState exist for the life of the current page. This includes postbacks (to the same page).

32. In what order do the events of an ASPX page execute. As a developer is it important to understand these events?

Answer: This is the order of Page events

a) Page_PreInit
b) Page_Init
c) Page_LoadViewState
d) Page_LoadPostData
e) Page_Load
f) Page_RaisePostDataChanged
g) Page_RaisePostBackEvent
h) Page_PreRender
i) Page_SaveViewState
j) Page_Render
k) Page_Dispose
l) Page_Error (this is caused whenever there is an exception at the page level).

Out of all the Page_Load is the one where your code gets loaded and your magic should be written. page_init occurs only once, i.e. when the page is initially created.

As a developer you need to know these, because your development activity is coding for these only.

33. Explain what is EnableViewState property of a ASP.NET Server Control?

Answer: On a round trip if we are not rendering the output of the same page/WebForm, the EnableViewState property of the controls on the page can be set to false. But if done so any changes in state of the control (dynamically changed properties) during the page life cycle are not added to the hidden element __VIEWSTATE and thus if round trip is made to the same page the old state of the control is not restored.

Example: ForeColor of the TextBox is lost if EnableViewState=False.

34. How control raises events?

Answer: When the form is posted the control compares its name value pair submitted with the old state in __VIEWSTATE, if it's different the control raises an appropriate event on the server. This is done between Init and Load event of Page but event will be raised only after "Load" event of the Page.

35. What will be the effect when EnableViewState="False" for a control whose events are handled on server?

Answer: If EnableViewState="False" for any control whose events are handled on server, those events won't raise any more.

36. When during the page processing cycle is ViewState available?

Answer: After the Init() and before the Page_Load(), or OnLoad() for a control.

37. What is @Page Directive in WebForm?

Answer: @Page - Defines page-specific attributes used by the ASP.NET page parser and compiler and can be included only in .aspx files. Page directives configure the runtime environment that will execute the page.

38. Explain how a web application works.

Answer: A web application resides in the server and serves the client's requests over internet. The client access the web page using browser from his machine. When a client makes a request, it receives the result in the form of HTML which are interpreted and displayed by the browser.