What is the difference between exe and dll.

exe file is a excutable file which runs in a seperate process which is managed by OS,where as a dll file is a dynamic link library which can be used in exe files and other dll files.In .net frame work both are assemblies.

fullname of .exe is Extensible Execute File. .exe use by End User like-Client We can Run the .exe

Process:

- An executing instance of a program is called a process.
- Some operating systems use the term â€˜taskâ€˜ to refer to a program that is being executed.
- A process is always stored in the main memory also termed as the primary memory or random access memory.
- Therefore, a process is termed as an active entity. It disappears if the machine is rebooted.
- Several process may be associated with a same program.
- On a multiprocessor system, multiple processes can be executed in parallel.
- On a uni-processor system, though true parallelism is not achieved, a process scheduling algorithm is applied and the processor is scheduled to execute each process one at a time yielding an illusion of concurrency.
- Example: Executing multiple instances of the â€˜Calculatorâ€™ program. Each of the instances are termed as a process.

Thread:

- A thread is a subset of the process.
- It is termed as a â€˜lightweight processâ€™, since it is similar to a real process but executes within the context of a process and shares the same resources allotted to the process by the kernel.
- Usually, a process has only one thread of control â€“ one set of machine instructions executing at a time.
- A process may also be made up of multiple threads of execution that execute instructions concurrently.

- Multiple threads of control can exploit the true parallelism possible on multiprocessor systems.
- On a uni-processor system, a thread scheduling algorithm is applied and the processor is scheduled to run each thread one at a time.
- All the threads running within a process share the same address space, file descriptors, stack and other process related attributes.
- Since the threads of a process share the same memory, synchronizing the access to the shared data withing the process gains unprecedented importance.

A data type is a value type if it holds the data within its own memory allocation. A reference type contains a pointer to another memory location that holds the data.

Value types include the following:

- All numeric data types
- Boolean, Char, and Date
- All structures, even if their members are reference types
- Enumerations, since their underlying type is always SByte, Short, Integer, Long, Byte, UShort, UInteger, or ULong

Reference types include the following:

- String
- All arrays, even if their elements are value types
- Class types, such as Form
- Delegates

The role of the **Finalize()** method is to ensure that a .NET object can clean up unmanaged resources when garbage collected. However, objects such as database connections or file handlers should be released as soon as possible, instead on relying on garbage collection. For that you should implement **IDisposable** interface, and release your resources in the Dispose() method.

throw ex; will erase your stacktrace. Don't do this unless you mean to clear the stacktrace. Just use

throw;

GetType() is used to find the actual type of a object reference at run-time. This can be different from the type of the variable that references the object, because of inheritance. typeof() creates a Type literal that is of the exact type specified and is determined at compile-time.

Is Postback is normally used on page _load event to detect if the web page is getting generated due to postback requested by a control on the page or if the page is getting loaded for the first time.

If we create a web Page, which consists of one or more Web Controls that are configured to use **AutoPostBack** (Every Web controls will have their own AutoPostBack property), the ASP.Net adds a special JavaScipt function to the rendered HTML Page. This function is named _doPostBack() . When Called, it triggers a PostBack, sending data back to the web Server.

ASP.NET also adds two additional hidden input fields that are used to pass information back to the server. This information consists of ID of the Control that raised the event and any additional information if needed. These fields will empty initially as shown below,

```
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
```

Â 1.**View state** is maintained in page level only.Â Â Â Â Â Â Â -

Â 2. View state of one page is not visible in another page.Â Â Â Â Â Â Â -

Â 3. View state information stored in client only.Â Â Â Â Â Â Â -

Â 4. View state persist the values of particular page in the client

Â Â Â (browser) when post backÂ Â Â Â operation done.
1**. Session** state is maintained in session level.
2. Session state value is available in all pages within a user session.Â Â Â -
Â 3. Session state information stored in server.Â Â Â â€“
4. Session state persist the data of particular user in the server.Â Â Â This data available till user close the browser or session timeÂ Â Â completes.
5. Session state used to persist the user-specific data on the server
Â Â Â Side

Whatâ€™s the use of reponse.output.writer()?

We can write formatted output using Response.Output.Write().

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

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

4. What is the difference between Server.Transfer and Response.Redirect?

In Server.Transfer page processing transfers from one page to the other page without making a round-trip back to the clientâ€™s browser. This provides a faster response with a little less overhead on the server. The clients url history list or current url Server does not update in case of Server.Transfer.

Response.Redirect is used to redirect the userâ€™s browser to another page or site. It performs trip back to the client where the clientâ€™s browser is redirected to the new page. The userâ€™s browser history list is updated to reflect the new address.

6. What are the different validators in ASP.NET?

1. Required field Validator
2. Range Validator
3. Compare Validator
4. Custom Validator
5. Regular expression Validator
6. Summary Validator

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

1. In-Process
2. 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.

12. How you can add an event handler?

Using the Attributes property of server side control

```
1 btnSubmit.Attributes.Add("onMouseOver","JavascriptCode();")
```

13. What is caching?

Caching is a technique used to increase performance by keeping frequently accessed data or files in memory. The request for a cached file/data will be accessed from cache instead of actual location of that file.

14. What are the different types of caching?
ASP.NET has 3 kinds of caching :

1. Output Caching,
2. Fragment Caching,
3. Data Caching.

15. Which type if caching will be used if we want to cache the portion of a page instead of whole page?

Fragment Caching: It caches the portion of the page generated by the request. For that, we can create user controls with the below code:

```
1 <%@ OutputCache Duration="120" VaryByParam="CategoryID;SelectedID"%>
```

16. List the events in page life cycle.

1) Page_PreInit
2) Page_Init
3) Page_InitComplete
4) Page_PreLoad
5) Page_Load
6) Page_LoadComplete
7) Page_PreRender
8)Render

17. Can we have a web application running without web.Config file?

Yes

18. Is it possible to create web application with both webforms and mvc?

Yes. We have to include below mvc assembly references in the web forms application to create hybrid application.

```
1
2 System.Web.Mvc
3 System.Web.Razor
4 System.ComponentModel.DataAnnotations
5
```

19. Can we add code files of different languages in App_Code folder?

No. The code files must be in same language to be kept in App_code folder.

20. What is Protected Configuration?

It is a feature used to secure connection string information.

24. What are the event handlers that we can have in Global.asax file?
Application Events: Application_Start , Application_End, Application_AcquireRequestState, Application_AuthenticateRequest, Application_AuthorizeRequest, Application_BeginRequest, Application_Disposed, Application_EndRequest, Application_Error, Application_PostRequestHandlerExecute, Application_PreRequestHandlerExecute,
Application_PreSendRequestContent, Application_PreSendRequestHeaders, Application_ReleaseRequestState, Application_ResolveRequestCache, Application_UpdateRequestCache

Session Events: Session_Start,Session_End

25. Which protocol is used to call a Web service?
HTTP Protocol

26. Can we have multiple web config files for an asp.net application?
Yes.

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

35. What are the asp.net Security Controls?

- <asp:Login>: Provides a standard login capability that allows the users to enter their credentials

- <asp:LoginName>: Allows you to display the name of the logged-in user
- <asp:LoginStatus>: Displays whether the user is authenticated or not
- <asp:LoginView>: Provides various login views depending on the selected template
- <asp:PasswordRecovery>: email the users their lost password

Â· **36. How do you register JavaScript for webcontrols ?**
We can register javascript for controls using <CONTROL -name>Attribtues.Add(scriptname,scripttext) method.
Â· **37. In which event are the controls fully loaded?**
Â· Page load event.
Â· **38. what is boxing and unboxing?**
Â· Boxing is assigning a value type to reference type variable.
Â· Unboxing is reverse of boxing ie. Assigning reference type variable to value type variable.
Â· **39. Differentiate strong typing and weak typing**
Â· In strong typing, the data types of variable are checked at compile time. On the other hand, in case of weak typing the variable data types are checked at runtime. In case of strong typing, there is no chance of compilation error. Scripts use weak typing and hence issues arises at runtime.

40. How we can force all the validation controls to run?

The *Page.Validate()* method is used to force all the validation controls to run and to perform validation.

41. List all templates of the Repeater control.

- *ItemTemplate*
- *AlternatingItemTemplate*
- *SeparatorTemplate*
- *HeaderTemplate*
- *FooterTemplate*

42. List the major built-in objects in ASP.NET?

- *Application*
- *Request*

- *Response*
- *Server*
- *Session*
- *Context*
- *Trace*

Â· **44. Which data type does the *RangeValidator* control support?**
Â· The data types supported by the *RangeValidator* control are Integer, Double, String, Currency, and Date.
Â· **45. What is the difference between an *HtmlInputCheckBox* control and an *HtmlInputRadioButton* control?**
Â· In *HtmlInputCheckBoxcontrol,* multiple item selection is possible whereas in *HtmlInputRadioButton* controls, we can select only single item from the group of items.
Â· **46. Which namespaces are necessary to create a localized application?**
Â· *System.Globalization*
Â· *System.Resources*
Â· **47. What are the different types of cookies in ASP.NET?**
Â· **Session Cookie** â€“ Resides on the client machine for a single session until the user does not log out.
Â· **Persistent Cookie** â€“ Resides on a userâ€™s machine for a period specified for its expiry, such as 10 days, one month, and never.
Â· **48. What is the file extension of web service?**
Â· Web services have file extension .asmx..
Â· **49. What are the components of ADO.NET?**
Â· The components of ADO.Net are Dataset, Data Reader, Data Adaptor, Command, connection.
Â· **50. What is the difference between ExecuteScalar and ExecuteNonQuery?**
Â· 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.
Â·

4. ASP.Net Page Life Cycle.

   A. There are few events which gets generated during the page execution like: Page_BeginRequest, Page_Init, Page_Load, Page_Prerender, Page_Render, Page_Unload etc

5. Boxing and Unboxing: Terminology, Advantages and Disadvantages.

   A. Converting the Value type data into the Reference type is called as **Boxing**.

```
byte b= 45;
Object o = b.Tostring();
```

   Converting the Reference type data and keep its value to stack is called as the **Unboxing**.

```
Object o=10;
Int i= Convert.ToInt32(o.ToString());
```

The Advantage of boxing and unboxing is that we can convert one type of the object to another type. The disadvantage is that it requires lot of memory and CPU cycles to convert from one type to another type.

7. What is Type Safety?

A. TypeSafe is a way through which the application or framework that the memory will not be leaked to outside environment. E.g. C# is the type safe language where you must have to assign any object before using it. In VB.Net it will take the default value. So C# is the type safe language while VB.Net is not.

8. What is Strong Name?

A. Strong Name (SN) is used to make the dll as the unique not by its name but by its version as:

```
SN -k fileName.dll
```

Now it will have the unique name with respect to the version. This assembly when placed in the GAC, it will treat as the unique with its version number and other details. 2 assemblies with the same name can exist in the GAC but both will have different version. The CLR takes the latest version assembly while running the application.

6. ### 10. What is worker process?

A. Worker process (w3wp.exe) is an executable which is also called as the Application Factory. This is used for the execution of the request and handling of the request for the web pages.

11. CLR and DLR?

A. CLR (Common Language Runtime) is the utility in the .Net framework to run the application. It is the runtime engine which actually executes the application with many responsibilities like taking care of memory management, versioning, CasPol etc.
DLR is new with .Net 4.0 which is the Dynamic Language Runtime and used to run the application on the fly wherever required. CLR runs as statically while DLR runs dynamically.

12. In case more than one dll versions of an installable is installed, which version is invoked by default?

A. By default the CLR will take and invoke the latest version of the dll and execute it accordingly. There could be the same name assemblies exists in the GAC but they will have different versions altogether for their uniqueness.
So while running the application, CLR takes the latest version assembly and use in the application.

13. What are Globalization and localization? How to implement them?

A. Globalization is the concept of developing the application in more than one language while the Localization is used for a particular language. Like if we develop the application in more than one language we need to create the resource files

(.resx) by using *System. Globalization* and when we open the application in a particular language, then the localizations used to convert that application to the selected language.

14. What is assembly, GAC? Where they are physically located?

A. Assembly is the collection of classes, namespaces, methods, properties which may be developed in different language and packed as a dll. So we can say that dll is also called as assembly.
There are 3 types of assemblies-
- Private Assembly, Shared Assembly, and Satellite Assembly.
GAC (Global Assembly Cache)- When the assembly is required by more than one project or application, we need to make the assembly with strong name and keep it in GAC or in Assembly folder by installing the assembly with the GACUtil command.
To make the assembly with strong name:
SN -k MyDll.dll
And to install it in GAC:
GacUtil -i MyDll.dll
GAC assemblies are physically stored in Assembly folder in the system.

15. How to configure HTTPS for a web application?

A. To configure the HTTPS (HTTP with Secure) for the web application, we need to have a client certificate. The client certificates can be purchased from the trusted providers and then we need to install that certificate for our site. By implementing the HTTPS, all the data which is passing will be in encrypted format, while makes the website more secure.

16. What are Inproc and Outproc in session? Where are session data stores in these cases?

A. Inproc and Outproc is the types of Sessions where the session data can be stored in the process memory of the application server(IIS) and in the separate state server.
When the session data is stored in the process memory(AppDomain) of the server(IIS), the session is called as the Inproc server. In this case when the server is restarted, the session data will be lost. So In the Inproc session mode, the session data stores in the memory object in AppDomain in Application Worker Process(AspNet_wp.exe)
When the session data is stored in the separate server like in state server or in Sql Server, the type of session is called as the Outproc session. In this case, if the server where the application is running is restarted, the session will be still remain in the separate servers.
So in the inproc session state, the session data is stored in the Process memory of the Server where the application is running.
In the Outproc session state, the session data is stored in the separate server- may be state server or in sql server.

17. When the View state is saved, and when is it loaded? How to enable/ disable View states?

7. **A.** View State data is stored in the current page in base64 encoded format. It gets loaded with the page and displays the values to the controls after the decoded. Internally it actually saves the check-sum of all the control data where the view state is enabled.so that when the page gets loaded due to any post back, it again finds the check-sum and then decodes the Base64 encoded string and gets back the same data to the controls. We can see the view state base64 encoded string in View Source of the page. It will be like _VIEWETATE="DSDSDF8DGDGDFGFD5FDGGDJFF23BNN457M9UJOG" this.
8. View state won't take the client or server memory to keep the view state data.

18. Difference between GET and POST. Which one is more secure?

A. GET and POST methods are used for the data transfer between the web pages. GET mainly used for small data which is not secure because in case of GET method, the data which we are passing will be visible in the url so we can't keep the secure data which will be visible in the url. There is also limited data which can be passed in case of GET method (max 255 character).
POST is used for transferring the huge data between the pages where we can keep the secure data and can transfer it. In case of using the POST method, the data which is transferring between the pages will not be visible so it is more secure than the GET method. Also there is no limit for POST method to post the data to the next page.
POST is more secure.

19. What are Razor engine? How is it different from ASPX Engines?

A. Razor engine is the new execution engine in the ASP.Net MVC 3 which is mainly used to convert the rezor syntax views to HTML page in the MVC applications. It takes the cshtml pages as the input for the ASP.Net MVC application and then render to the HTML as the output. The ASPX engine takes the aspx code and then renders to the HTML.

20. Pros and cons of JavaScript and AJAX.

A. JavaScript is a scripting language and mainly used for client side validation. We can validate the client side data before sending to the server. So by this we can improve the performance of the application.
Ajax is Asynchronous JavaScript and XML which is used for the Asynchronous calls to the server. It uses the JavaScript/JQuery for making the call and use XML for the Data Transfer. It basically uses the XmlHttpRequest for the asynchronous calls to the server and communicates with the XML data which is platform independent. So Ajax can be used with any technology.

21. In how many different ways JavaScript code can be used/called in an application?

A. JavaScript can be used for Client Side validation, can also be used for calling of server side methods and functions, can be used for calling the web services, WCF service, Web API's, Calling the Controller and Action methods in ASP.Net MVC etc.

22. What needs to be done to call a JavaScript function from code behind?

A. If we want to call the JavaScript function from the code behind, we need to attach the JavaScript to the events in the page_load event as:

```
protected void btnSave_cliekc9object sender, EventArgs e)
{
 btnSave.Attributes.Add("onclick,"JavaScript: retrun Validatedata();");
}
```

Here ValidateData is the JavaScript function which can be used to validate the page data and if validation fails, it will return and will not execute the server side btnSave_click event.

23. Difference between Server Controls and User controls?

A. User controls are used for the re-usability for the controls in the application. By using the Web User Control template, we create the new user controls and then we can use the same control in the various pages. User controls can be created by combining more than one control to extend the functionality of the existing controls. To use the user controls, first we need to register them in the web page where we want to use that control. A separate copy is needed in each page where we want to use the user control. User controls can't be included into the toolbox.

Web Server controls are those controls which can be found in the toolbox and can be directly drag to the application like other controls textbox, button etc. For the web server control, only 1 copy of the control is needed irrespective of the number of web pages. If we want 10 text-boxes to be added in our web page, we need only 1 copy of the textbox in the toolbox and can be dragged 10 times.

24. Difference between Var, object and Dynamic types.

A. var is the keyword introduced with .Net 3.5 and used to store any kind of data like data-set, data table, int, float, char etc. We can keep any kind of data into the var variable.

```
var myVar = new String[] {"hello", "world!!"} ;
```

Here the myVar is the var type variable which is used to store the string array. Like this we can store any type of data into the var.

Object is the type which is used to store the objects of any kind. These objects need to be type cast when used.

Like object myObject = "Hello"

Here the myObject variable of object type is used to keep the string variable. Now when we want this variable value, we need to typecast it like

```
string strvar= (string) myobject;
```

Var is the compile time where the type of the var variable gets defined at the compilation of the program. Once we define the type during compilation, we cant make the changes of the type during run time or at the later stages.

Let's say, if we define like

```
var myVar = new String[] {"hello", "world!!"} ;
```
so here myVar variable will be of string array. Now after this, I can;t make it like:
```
var myVar = 10 ;
```
If we do this, it will throw error.
Dynamic- Itâ€™s a keyword introduces with the .Net 4.0 and used to keep the data similar to the var keyword. The type of Dynamic type can be changed even at run time.
So if we define like:
```
dynamic myVar = new String[] {"hello", "world!!"} ;
```
Then the myVar is of type string array. Now if we do like:
```
dynamic myVar = "Hello" ;
```
Now the myVar type will be used as string type. So we can change the type in case of dynamic.
The difference between the var and dynamic is that the dynamic variable uses the same memory location to store the object and not changes throughout the application.

25. Difference between Functions and methods.

A. In.Net terminology, both function and method are same. In general, we use method in server side code of .Net but in scripting language we use function like JavaScript function.
Here the difference can be function always returns a value whereas method may or may not. It depends upon the return type of the method.

26. Difference between Abstract classes and Interface. Explain with scenario where to implement one?

A. Abstract Class: Collection of the Abstract (Incomplete) and Concrete (complete) members is called as the Abstract class. If there is at least one abstract member in a class, the class must be declared as abstract class.
When there is the similar behavior, we can use the abstract class.
e.g. We want to calculate the area of few shapes. As this is not generic to the application. We have few shapes - like Circle, Ellipse, Parabola, Hyperbola, Triangle etc.
So we can create an abstract class and implement it like below:

```
public abstract class MyAbstractClass
{
 // some other concrete members
 public abstract void Area();// abstract method
}
```

Now in the child class, letâ€™s say i have a circle class and want to calculate the area of the circle:

```
public class Circle: MyAbstractClass
{
 public override void Area()
```

```
 {
  // calculate the area of the circle
 }
}
```

In the similar fashion, we can calculate the area of other shapes.

Interface: Collection of abstract members is called as the Interface. When the behavior is not similar, we need to use the interface. All the members of the interface
must be overridden in the child classes.

e.g. Print functionality of the application can have an interface method like:

```
interface Inf
{
 void Print();
}
```

Now as this is the generic functionality and can be implemented in any of the class so we have taken it as interface. We can implement this functionality into any page like:

```
class MyClass: System.Web.UI.Page, Inf
{
 public void Print()
 {
  // implement details about the Print method
 }
// Here we can implement any kind of print-like print to excel, xml, word all depends on the our decision.
}
```

27. Different forms of Polymorphism. Differences between Abstraction and Polymorphism.

A. Polymorphism is to use the same function in many forms. The polymorphism is of 2 types-

a. Classical polymorphism (Overloading)

b. AdHoc polymorphism (Overriding)

Polymorphism = Poly(many) + Morphism(Forms)

9. **Overloading**: When the runtime (CLR) find the behavior of class members at the compilation of the program, it is called as the Classical polymorphism or Overloading. In this, the method name is same but prototypes (method + parameters) are different and it is implemented in the same class.
10. e.g.

```
public class MyClass
{
  public int  Add(int a, int b)
```

```
  {
   return a+b;
  }
  public int  Add(int a, int b, int c)
  {
   return a+b+c;
  }
}
```

Overriding: When the runtime (CLR) find the behavior of class members at the runtime of the program, it is called as the AdHoc polymorphism or Overriding. In this, the method name as well as the prototype (method + parameters) is same but they are implemented in the different class. We use virtual keyword in the base class method to override in the child class using the override keyword.

e.g.

```
public class MyBaseClass
{
 public virtual void Show(string  message)
 {
  Console.WriteLine(â€œYour message is : â€+ message);
 }
}
public class MyChildClass: MyBaseClass
{
 public override void Show(string message)
 {
  Console.WriteLine(â€œYour new message is : â€+ message);
 }
}
```

11. **Abstraction** is the behavior to get the required functionality. To implement the abstraction, we use access specifiers where if we declare the members as private, it means they will be available only to the current class and if we make them as public, the other classes can also use them. So Abstraction is used to show only the essential features It is also used to hide the unnecessary data which is not relevant but present.
12. Abstract keyword is also used to get the abstraction behavior. We can use Abstract Class and Interface to implement Abstraction.

28. What are Delegates and Events?

A. A **Delegate** is an object, which points to another method in the application. Delegate holds- name of the method,

arguments of the method (if any) and the return type of the method.
See the below points regarding the Delegate:-
Â· delegate keyword is sealed type in System. Multicast namespace.
Â· Delegate works like a function pointer in C language.
Â· Delegate holds the address of the function.
Â· Delegate hides the actual information which is written inside the method definition.
Â· A delegate can hold address of a single function as well as the address of multiple functions.
Â· There are 2 types of delegate-
- Single cast delegate (hold single function) and
- Multicast delegate(hold multiple functions).
Â· Addition and subtraction are allowed for the delegates but NOT multiplication and division. It means, we can add delegates, subtract delegates etc.
e.g. To create a single cast delegate, first we can create a class with a method as:

```
public class DelegateDemo
{
 public void Show(string msg)
 {
  Console.WriteLine(msg);
 }
}
```

Now we can call the method **Show** using the delegate as:

```
public  delegate void MyDelegate(string message); //declare delegate
```

now we need to create the object of the delegate with the address of the method as:

```
DelegateDemo obj = new DelegateDemo();//class object
MyDelegate md= new MyDelegate(obj.Show(â€œHello World!!â€));
md(); // call the delegate
```

We can create the events and event handler by using delegate with the below syntax:

```
public delegate void textChangedEventHandler(Object sender, TextEventArgs e);
```

This event handler will be used to handle the textbox textchanged event.
More details about the delegate and events can be found at the below link:

29. Covariance and Contra-variance.

A. covariance and contravariance are the new features added with the .Net 4.0. They are basically used for the implicit

reference conversion for different .Net types like array, delegate, and generic etc.
You can go to the below link for more details with the examples that how we can use the covariance and contrvariance to implicate reference conversion:
http://blogs.msdn.com/b/csharpfaq/archive/2010/02/16/covariance-and-contravariance-faq.aspx

30. What are Extension methods?

A. Extension methods are special types of methods which are static methods but called as the instance methods. The extension methods are added with the .Net framework 3.5 and with the Visual Studio 2008.
These methods wonâ€™t affect the existing class and the label. These methods are used for the extra behavior which the calls can provide. There is no need to build the class again if we add any extension method to the class.
There are various inbuilt methods added in .Net 3.5 with the introduction of LINQ. We can see the extension methods like Order By when we use the Linq as:
e.g.

```
int[] numbers = { 10, 45, 15, 39, 21, 26 };
var orderedNumbers = numbers.OrderBy(a => a);
```

31. What are Anonymous methods and Lambda Expression?

A. Anonymous methods are those methods which does not have the name. As they donâ€™t have the name, so there is no way to call these methods. These methods are created by using the delegate as below:

```
button1.Click += delegate{listBox1.Items.Add(textBox1.Text)};
```

Lambda Expression: Itâ€™s an easy way to create anonymous functions. It is also an anonymous function which has the capability to contain expressions and statements. We can create the delegate and expression tree types using the lambda expression.
For more details regarding the anonymous method and lambda expression, we can go through the below link:
http://www.codeproject.com/Articles/47887/C-Delegates-Anonymous-Methods-and-Lambda-Expression

32. Multithreading. How to implement Multithreading?

A. Executing more than one processes simultaneously is called as multithreading. To implement the multithreading concept, we need to use the System. Threading .dll assembly and the System. Threading namespace.
To write the thread program, we need to create a class with the method. Now we can create the thread object and then pass the method by using the class object to the method.
After that we need to create the ThreadStart delegate which will call the actual method of the class.
You can go through below link for more explanation and other details regarding the implementation and the code snippet:
http://www.codeproject.com/Articles/1083/Multithreaded-Programming-Using-C

33. Which interface is used to-

a. Convert Boolean values to Visibility values?
b. Compare two integer values?
c. Compare String values?

A. Check the below interfaces which are used in these scenarios:
a. Convert Boolean values to Visibility values?
b. Compare two integer values?- IComparable interface
c. Compare String values? IComparer interface

SQL Server

34. What is the difference between a View and a Cursor?

A. View: It is one of the database object which is also called as virtual table. We can also say that it is a window through which we can see some part of database. View is also called as stored query because we are going to fetch data using View.
View does not contain any data. Itâ€™s just a virtual table which is used to get the records from the base table for which the view is created. View is faster than ad hoc queries because when we create the view and execute it once. Next time onwards it will be available as the compiled format. So whenever the view is called, it will just execute rather than compiling.
Cursor: Cursor is a database object which is also the buffer area and created as a result of any sql statement to hold the intermediate values.
Cursor is used to format the rows individually. By using the cursor, we can process the individual rows. There are 4 types of cursors in Sql Server-
a. Static Cursor
b. Dynamic Cursor
c. Key set cursor
d. Read-only cursor

35. How to execute multiple update on different conditions in a single query?

A. To execute multiple update using a single Sql update statement is the new feature available with the SQL Server 2008. In this, we can update multiple rows using a single update command.

36. Left outer joins and Right Outer joins

A. Joins are used to retrieve data from more than 1 tables using some conditions. There are 3 types of outer joins in SQL Server database-
a. Left Outer Join
b. Right Outer Join
c. Full Join
In order to extract the matched rows from both the tables and unmatched rows from the first table, left Outer join is used. The

syntax for left outer join condition is:
T1.Col1* = T2.Col1
In order to extract the matched rows from both the tables and unmatched rows from the second table, right Outer join is used. The syntax for right outer join condition is:
T1.Col1 = *T2.Col1
In order to extract the matched rows from both the tables and unmatched rows from the first table and then unmatched row from the second table, full join is used. The syntax for full join condition is:
T1.Col1* = *T2.Col1

37. Exception handling.

A. Exception Handling is the way to handle the unexpected error at runtime of the application. From the SQL Server 2005 version, tryâ€¦catch block is also supported to catch the exceptions in SQL Server database. There is various other ways to catch the error like using Global temporary variables @@Error, inbuilt method RaiseError etc.

38. What is Performance Tuning? How do you implement it.

A. Performance Tuning is the process through which we can optimize the SQL Server objects like functions, triggers, stored procedure to achieve high response time to the front end applications. In the performance tuning process we generally check for the below point and optimize the objects processing:
a. Through Query Execution plan, check for the processing time of the query execution.
b. Check the join conditions and break all the condition for executions of the queries individually.
c. Check for the error prone process, conditions in the queries.
d. Check for the loops whether they are terminated if any error occurs.
e. Check for the processes which are taking more time in execution and how to reduce the response time.

39. Difference between Having and Where clauses.

A. When the 'where' clause is not able to evaluate the condition which consists of group functions, Having clause is used. Having clause is always followed by the Group By clause.
'Where' clause is used to filter the records based on the conditions. If there is the requirement to get the group data in the select statement and where clause is not able to get it, we can use the Having clause.
e.g. Display DeptNo, No.of Employees in the department for all the departments where more than 3 employees are working

```
SELECT DEPTNO, COUNT(*) AS TOTAL_EMPLOYEE
FROM EMP
GROUP BY DEPTNO HAVING COUNT(*) >3
```

40. Difference between Temp tables and Tables variables?

A. Temp Table in SQL Server:

a. Temp table are the special type of tables which are used to store the intermediate data of the actual table.
b. Temp tables are only visible to the current sessions of the sql server instance. When the session end, these table data automatically drops.
c. We canâ€™t join the temp tables as they donâ€™t allow the foreign key constraints.
d. Temp tables are created in TempDB database.
e. We can use the same temp table name for the different user sessions.
f. Mostly used in stored procedure to handle the intermediate data.

41. What does @ and @@ suffixed by property names specify?

A. @- This is used for the variable declaration
e.g. @name varchar2(50)
@@- This is used for the Global variable declaration
e.g. @@Error=0

42. Self-join queries.

A. Self-Join is a type of join which is used to join the same table by creating the multiple instances of the same table. So we can join 2 instances of the same table in case of self-join. This type of join is used when there is the requirement to get the referenced data which is available in the same table.
e.g. A table contains EmpId, Ename and ManagerId
As the manager is also an employee. Now if we want that who is the manager of which employee. In this situation, we need to create the instance of the same table and get the required data as:

```
SELECT EMPID, ENAME, ENAME AS [MANAGER NAME]
FROM EMP E1, EMP E2
WHERE E1.EMPID= E2.MANAGERID
```

43. Types of Index.

A. Index is one of the database objects which is used to improve the performance of the database queries. It reduces the table scan while retrieving the data from the database and the search gets fast-
There are 2 types of indexes used in the SQL server:
a. Clustered index
b. Non clustered index
There are 3 more types of index but those comes under the above two-
a. Unique index
b. Composite Index
c. XML Index-added in SQL Server 2005
The index basically works on searching like binary tree where the root value is the finding value and it will be compared with the

partitioned value of the tree.

44. Difference between Primary key, Unique key and Candidate key?

A. Primary Key- It is a key to make the unique identification of the row in a table. It doesn't allow null values in the primary key column. We can create the lookup columns based on the primary key. One table allows maximum of 1 primary key and in 1 table, we can create the primary key column by using 16 columns. Due to one of the normalization rule, we have to create primary key for the table to make the rows unique.
Unique Key:- (Primary Key + Not null) is called as unique key. Unique key is also used to make the rows as unique in a table. The only difference between primary key and unique key is that primary key does not allow null value while the unique key allow. The limitation of the null in unique key is that it allows only one Null value. So only in one row, we can make the key as null for the unique key.
Candidate key- The key other than primary key, to make the rows as unique is called as candidate key. In candidate key, we take the columns which are not in the primary key and make the key for uniqueness of the row.

45. What is the default value for Datetime. What are Min and Max values for Date in 2008.

A. The default value of Date is CURRENT_TIMESTAMP
Below are the new date and time values in Sql Server 2008:
In SQL Server 2008:
1. DateTime2
Min Value: 0001-01-01 00:00:00.0000000
Max Value: 9999-12-31 23:59:59.9999999
2. Date
Min Value: 0001-01-01
Max Value: 9999-12-31

WCF

46. What is WCF also known as?

A. WCF (Windows Communication Foundation) is also know as Indigo by its code name.

47. Difference between WCF and Web Services?

A. Below are the main differences between the WCF and Web Service:
Web Service:
a. Can be hosted in IIS only
b. Only two types of operations affects- One-Way, Request-Response
c. To serialize the data use System.Xml.Serialization

d. To encode the data use- XML 1.0, MTOM, DIME, Custom
e. Web Service can be accessed through HTTP channel.
WCF service:
a. Can be hosted in IIS, Self Hosting, WAS, Windows Services etc
b. Three types of operations affects- One-Way, Request-Response and Duplex
c. To serialize the data use System.Runtimel.Serialization
d. To encode the data use- XML 1.0, MTOM,Binary, Custom
e. WCF Service can be accessed through HTTP, TCP, Named pipes, MSMQ,P2P etc.

48. What are Endpoints?

A. The collection of Address, Binding and Contract is called as End Point. In Sort,
EndPoint = A+B+C
Address (Where)- It means where the service is hosted. URL of the service shows the address.
Binding (How)- How to connect to the service, is defined by the Binding. It basically has the definition of the communication channel to communicate to the WCF service
Contract (what)- It means what the service contains for the client. What all the methods are implemented in the WCF service is implemented in the Contract.

49. What are Behavior and Bindings?

A. Binding mainly describes about the communication of the client and service. For this, there are protocols corresponding to the binding behavior which will take care of the communication channel. There are different protocols which we use for the different types of bindings. E.g. HTTP, TCP, MSMQ, Named Pipes etc.
Behavior is used for the common configurations that could be for endpoints. When we use the common behavior, they affect to all the end points. Adding the service behavior affect the service related stuff while the endpoint related behavior affects the end points. Also operations level behavior affects the operations.

50. What are different types of Contracts supported?

A. There are mainly 5 type of contracts used in WCF service:
a. Service Contract
b. Operation Contract
c. Data Contract
d. Message Contract
e. Fault Contract

51. What is the difference between Transport and Message Security mode?

A. WCF supports 2 types of security- Transport Level Security and Message Level Security

Transport Level Security- In this type of security, we make the transport channel as secure so that the data flows in that channel will be automatically secured. For HTTP channel, we use the client certificate for the security of the web address. SSL is used for the HTTP channel security. As we donâ€™t need to secure each of the messages which are floating between the client and the service, the speed is faster as direct message is going to the client from the service.
Message level security- This type of security in WCF is used where we don't have the fixed transport medium and we need to secure each message which is floating between the server and the client. In this type of security we use certain algorithms for making the message as secure message. We use some extra bits and send with the message. We also use some encryption techniques like SHA1 or MD5 which make the proper security for our message. As each message needs to be secured, this type of security makes some delay in the process of sending and receiving the messages.

52. How to configure WCF security to support Windows authentication?

A. To support the WCF security in Windows Authentication, we need to add the ClientCredetialType attribute to â€œWindowsâ€ under the security tab element:
transport clientCredentialType="Windows"

53. How to use Fault Contract?

A. Fault Contract is mainly used for viewing and displaying the errors which occurred in the service. So it basically documents the error and the error message can be shown to the user in the understandable way. We canâ€™t use here the tryâ€¦.catch block for the error handling because the tryâ€¦catch is the technology specific (.Net Technology). If we use the try...catch block for handling the errors, the error will not be reached to the client who is consuming the service. Because this error will not be included in the message. So we use the Fault contract for the error handling.
e.g. To use the Fault contract, we can simply write like the below:

```
public  int Add(int number1,int number2)
{
 // write some implementation
 throw new FaultException (â€œError while adding data..â€);
}
```

Here the fault Exception method is the inbuilt method which will throw the exception and display the message . We can use the custom class so that the message can be customized and the customized message can be sent to the client.
So we can create a clss like:

```
public Class CustomException
{
 public int ID {get;set;}
 public string Message {get;set;}
 public string Type{get;set;}
}
```

Now this custom type we can use with the Operation Contract as:

```
[ServiceContract]
public interface IMyInterface
{
 [OperationContract]
 [FaultContract(typeOf(CustomException))]
 int Add(int num1,int num2);
}
```

Now while implementation of the Add method, we can assign the class properties.

8) List out the differences between Array and ArrayList in C#?

- Array stores the values or elements of same data type but arraylist stores values of different datatypes.
- Arrays will use the fixed length but arraylist does not uses fixed length like array.

Â· **9) Why to use â€œusingâ€ in C#?**

Â· â€œUsingâ€ statement calls â€“ â€œdisposeâ€ method internally, whenever any exception occurred in any method call and in â€œUsingâ€ statement objects are read only and cannot be reassignable or modifiable.

12) What is the difference between â€œconstantâ€ and â€œreadonlyâ€ variables in C#?

- â€œConstâ€ keyword is used for making an entity constant. We cannot modify the value later in the code. Value assigning is mandatory to constant variables.
- â€œreadonlyâ€ variable value can be changed during runtime and value to readonly variables can be assigned in the constructor or at the time of declaration.

13) Explain â€œstaticâ€ keyword in C#?

â€œStaticâ€ keyword can be used for declaring a static member. If the class is made static then all the members of the class are also made static. If the variable is made static then it will have a single instance and the value change is updated in this instance.

14) What is the difference between â€œdisposeâ€ and â€œfinalizeâ€ variables in C#?

- Dispose - This method uses interface â€“ â€œIDisposableâ€ interface and it will free up both managed and unmanaged codes like â€“

database connection, files etc.

- Finalize - This method is called internally unlike Dispose method which is called explicitly. It is called by garbage collector and canâ€™t be called from the code.

18) What is the difference between â€œfinalizeâ€ and â€œfinallyâ€ methods in C#?

- Finalize â€“ This method is used for garbage collection. So before destroying an object this method is called as part of clean up activity.
- Finally â€“ This method is used for executing the code irrespective of exception occurred or not.

Â· **20) Can we have only â€œtryâ€ block without â€œcatchâ€ block in C#?**
Â· Yes we can have only try block without catch block.
Â· **22) Do we get error while executing â€œfinallyâ€ block in C#?**
Â· Yes. We may get error in finally block.

24) What are the differences between static, public and void in C#?

- Static classes/methods/variables are accessible throughout the application without creating instance. Compiler will store the method address as an entry point.
- Public methods or variables are accessible throughout the application.
- Void is used for the methods to indicate it will not return any value.

Â· **25) What is the difference between â€œoutâ€ and â€œrefâ€ parameters in C#?**
Â· â€œoutâ€ parameter can be passed to a method and it need not be initialized where as â€œrefâ€ parameter has to be initialized before it is used.
Â· **26) Explain Jagged Arrays in C#?**
Â· If the elements of an array is an array then itâ€™s called as jagged array. The elements can be of different sizes and dimensions.

29) What are reference types in C#?

Below are the list of reference types in C# -

- class

- string
- interface
- object

Â· **33) What you mean by inner exception in C#?**
Â· Inner exception is a property of exception class which will give you a brief insight of the exception i.e, parent exception and child exception details.

35) What is the difference between â€œStringBuilderâ€ and â€œStringâ€ in C#?

- StringBuilder is mutable, which means once object for stringbuilder is created, it later be modified either using Append, Remove or Replace.
- String is immutable and it means we cannot modify the string object and will always create new object in memory of string type.

Â· **40) Explain Generics in C#?**
Â· Generics in c# is used to make the code reusable and which intern decreases the code redundancy and increases the performance and type safety.
Namespace â€“ â€œSystem.Collections.Genericâ€ is available in C# and this should be used over â€œSystem.Collectionsâ€ types.
Â· **41) Explain object pool in C#?**
Â· Object pool is used to track the objects which are being used in the code. So object pool reduces the object creation overhead.

42) What you mean by delegate in C#?

Delegates are type safe pointers unlike function pointers as in C++. Delegate is used to represent the reference of the methods of some return type and parameters.

43) What are the types of delegates in C#?

Below are the uses of delegates in C# -

- Single Delegate
- Multicast Delegate

- Generic Delegate

44) What are the three types of Generic delegates in C#?

Below are the three types of generic delegates in C# -

- Func
- Action
- Predicate

45) What are the differences between events and delegates in C#?

Main difference between event and delegate is event will provide one more of encapsulation over delegates. So when you are using events destination will listen to it but delegates are naked, which works in subscriber/destination model.

47) What are the uses of delegates in C#?
Below are the list of uses of delegates in C# -

- Callback Mechanism
- Asynchronous Processing
- Abstract and Encapsulate method
- Multicasting

Â· **49) Why to use â€œNullable Coalescing Operatorâ€ (??) in C#?**
Â· Nullable Coalescing Operator can be used with reference types and nullable value types. So if the first operand of the expression is null then the value of second operand is assigned to the variable. For example,
Â·
```
double? myFirstno = null;
double mySecno;
mySecno = myFirstno ?? 10.11;
```

50) What is the difference between â€œasâ€ and â€œisâ€ operators in C#?

- â€œasâ€ operator is used for casting object to type or class.
- â€œisâ€ operator is used for checking the object with type and this will return a Boolean value.

51) Define Multicast Delegate in C#?

A delegate with multiple handlers are called as multicast delegate. The example to demonstrate the same is given below

```
public delegate void CalculateMyNumbers(int x, int y);
int x = 6;
int y = 7;
CalculateMyNumbers addMyNumbers = new CalculateMyNumbers(FuncForAddingNumbers);
CalculateMyNumbers multiplyMyNumbers = new CalculateMyNumbers(FuncForMultiplyingNumbers);
CalculateMyNumbers multiCast = (CalculateMyNumbers)Delegate.Combine (addMyNumbers, multiplyMyNumbers);
multiCast.Invoke(a,b);
```

52) What is the difference between CType and Directcast in C#?

- CType is used for conversion between type and the expression.
- Directcast is used for converting the object type which requires run time type to be the same as specified type.

53) Is C# code is unmanaged or managed code?

C# code is managed code because the compiler â€“ CLR will compile the code to Intermediate Language.

54) Why to use lock statement in C#?

Lock will make sure one thread will not intercept the other thread which is running the part of code. So lock statement will make the thread wait, block till the object is being released.

55) Explain Hashtable in C#?

It is used to store the key/value pairs based on hash code of the key. Key will be used to access the element in the collection. For example,

Hashtable myHashtbl = new Hashtable();
myHashtbl.Add("1", "TestValue1");
myHashtbl.Add("2", "TestValue2");

56) How to check whether hash table contains specific key in C#?

Method â€“ â€œContainsKeyâ€ can be used to check the key in hash table. Below is the sample code for the same â€“

```
Eg: myHashtbl.ContainsKey("1");
```

57) What is enum in C#?

enum keyword is used for declaring an enumeration, which consists of named constants and it is called as enumerator lists. Enums are value types in C# and these canâ€™t be inherited. Below is the sample code of using Enums

```
Eg: enum Fruits { Apple, Orange, Banana, WaterMelon};
```

58) Which are the loop types available in C#?

Below are the loop types in C# -

For
While
Do-While

59) What is the difference between â€œcontinueâ€ and â€œbreakâ€ statements in C#?

- â€œcontinueâ€ statement is used to pass the control to next iteration. This statement can be used with â€“ â€œwhileâ€, â€œforâ€, â€œforeachâ€ loops.
- â€œbreakâ€ statement is used to exit the loop.

70) What are the collection types can be used in C#?

Below are the collection types in C# -

- ArrayList
- Stack
- Queue

- SortedList
- HashTable
- Bit Array

71) Explain Attributes in C#?

- Attributes are used to convey the info for runtime about the behavior of elements like â€“ â€œmethodsâ€, â€œclassesâ€, â€œenumsâ€ etc.
- Attributes can be used to add metadata like â€“ comments, classes, compiler instruction etc.

72) List out the pre defined attributes in C#?

Below are the predefined attributes in C# -

- Conditional
- Obsolete
- Attribute Usage

73) What is Thread in C#?

Thread is an execution path of a program. Thread is used to define the different or unique flow of control. If our application involves some time consuming processes then itâ€™s better to use Multithreading. which involves multiple threads.

74) List out the states of a thread in C#?

Below are the states of thread â€“

- Unstarted State
- Ready State

- Not Runnable State
- Dead State

Â· **79) Explain Static Members in C# ?**

Â· If an attribute's value had to be same across all the instances of the same class, the static keyword is used. For example, if the Minimum salary should be set for all employees in the employee class, use the following code.

Â· `private static double MinSalary = 30000;`

Â· To access a private or public attribute or method in a class, at first an object of the class should be created. Then by using the object instance of that class, attributes or methods can be accessed. To access a static variable, we don't want to create an instance of the class containing the static variable. We can directly refer that static variable as shown below.

Â· `double var = Employee.MinSalary ;`

#temp tables are available ONLY to the session that created it and are dropped when the session is closed. ##temp tables (global) are available to ALL sessions, but are still dropped when the session that created it is closed and all other references to them are closed.

1. What are Windows services?

Windows services, previously known as NT services, are applications that are installed on the system as system services. In other words, Windows services are applications that run in the background with the Windows operating system. The primary use of Windows services is to reduce the consumption of memory required for performing backend operations. Let's take an example to understand this easily. Suppose you want to perform a variety of functions, such as monitor the performance of your computer or application, check the status of an application, and manage various devices, such as printers.

In such a case, you can use Windows services to reduce memory consumption. In addition, Windows services can run on your system even if you have not logged on to your computer. In addition, these services do not have any user interface.