**Software development environments**

**What are the different environments in your development process or development life cycle at your company?**  
This is a general interview question and not very specific to ASP.NET. Usually, the interviewer asks this question to **measure your understanding of the different environments and their role in software development**. Some interviewers, may also ask this question, to check if you really have the work experience you are claiming or **just faking** it.   
   
   
  
**1.** Development  
**2.** QA  
**3.** Staging  
**4.** UAT (User Acceptance Testing)  
**5.** Production  
  
**1. Development Environment** - All the developers check in their current work into development environment.  
  
**2. QA (Quality Assurance) Environment** - This is the environment, where testers (QA) test the application. QA cannot test on development environment, because developers are continuously checking in new code. So, if there is a bug, we don't know, if it's caused by the old or new code. In short,  if development is going on in the same environment it would be difficult to keep up with the current state. There will be lot of confusion, if the developer is trying to fix in the same area as the tester is testing. Without development and QA environment being seperate their is no way to do proper testing.  
  
**3. Staging Environment**- Many organisations, try to keep their staging environment as identical as possible to the actual production environment. The primary reason for this environment is to identify any deployment related issues. Also, if you are developing a B2B (Business to Business) application, you may be interfacing with other service provider systems. Many organisations, usually setup their staging environment to interface with the service providers as well, for complete end to end testing.  
  
**4. Production Environment** - The actual live environment, that we use for day to day business.   
  
**Note:** In general, the code flows from **Development** => **QA** => **Staging** => **Production**

**When I create a new ASP.NET 4 web application, the web.config file is almost empty. What happened to all the configuration elements that were there prior to ASP.NET 4?**

All the major configuration settings are moved into machine.config file, and all the applications will inherit the setting from this file. If an application needs to override the default settings, we can do so using the application specific configuration (web.config) file. ASP.NET 4 applications have clean web.config files.  
  
  
**If you create a new asp.net 4 empty web application, the only entry that you will find is shown below**.

|  |
| --- |
| http://3.bp.blogspot.com/-kzsxToiqfgM/TuZa0JXwc3I/AAAAAAAAAI0/piFQvPYFTVU/s1600/CleanWebConfig.png |

**Note:** If you create a **new ASP.NET Web Application**, instead of a **new ASP.NET Empty Web Application**, you will find a lot more confiuration entries in the web.config file. This is because, the template for ASP.NET web application overrides some of the default settings inherited from machine.config.

**ASP.NET Page is very slow. What will you do to make it fast**

This is a very common asp.net interview question asked in many interviews. There are several reasons for the page being slow. We need to identify the cause.   
  
**1.** Find out which is slow, is it the **application**or the **database** : If the page is executing SQL queries or stored procedures, run those on the database and check how long do they take to run. If the queries are taking most of the time, then you know you have to tune the queries for better performance. To tune the queries, there are several ways and I have listed some of them below.  
   a) Check if there are indexes to help the query  
   b) Select only the required columns, avoid **Select \*.**  
   c) Check if there is a possiblity to reduce the number of joins  
   d) If possible use **NO LOCK**on your select statements  
   e) Check if there are cursors and if you can replace them with joins  
  
**2.** If the queries are running fast, then we know it is the **application code**that is causing the slowness. Isolate the page event that is causing the issue by turning tracing on. To turn tracing on, set **Trace="true"** in the page directive. Once you have tracing turned on you should see trace information at the bottom of the page as shown in the image below. In this case Page Load event is taking the maximum time. So we know, the code in**Page\_Load**event is causing the issue. Once you look at the code, you should be able to nail down the issue.

|  |
| --- |
| http://3.bp.blogspot.com/--Tixyx637yY/TmJG0Zi0v_I/AAAAAAAAAIA/X0kXmY87dmk/s1600/PageSlow.png |

**Difference between EnableViewState and ViewStateMode properties**

1. Using **EnableViewState** property we only have 2 options

     We can turn off view state altogether (Set **EnableViewState to false for page, then whatever the value of ViewStateMode is it will be ignored**)

                              or

     Enable viewstate for the entire page and then turn it off on a control-by-control basis.

(Set **EnableViewState to true and set EnableViewState for control as true**)  
  
2. If you want to turn of **ViewState** for the entire page and only enable it for specific controls on the page, then we have to use **ViewStateMode** property in conjunction with **EnableViewState**.  
  
3. **EnableViewState** property only accepts true or false values and the default value is true, where as **ViewStateMode** property can have a value of - Enabled, Disabled and inherit. Inherit is the default value for **ViewStateMode** property.  
  
4. **ViewStateMode** property is introduced in ASP.NET 4, where as **EnableViewState** exists from a long time.  
  
5. If **EnableViewState** is to True, only then the **ViewStateMode** settings are applied, whereas, if **EnableViewState** is set to False then the control will not save its view state, regardless of the **ViewStateMode** setting. In short if **EnableViewState** is set to False, **ViewStateMode** setting is not respected.  
  
6. To disable view state for a page and to enable it for a specific control on the page, set the **EnableViewState** property of the page and the control to true, set the **ViewStateMode** property of the page to Disabled, and set the **ViewStateMode** property of the control to Enabled.

**What is SQL Injection Attack**

Let us understand **SQL injection attack**, with an example. I have an **Employee Search Page** as shown in the image below. This webform has a very simple functionality. You enter the ID of the employee, you want to search and click the **Search Employee button**. If a match is found in the database, we show the employee record in the GridView.

|  |
| --- |
| **Employee Search Page** http://4.bp.blogspot.com/-oPNfz3LirRk/TiiFNFL8VXI/AAAAAAAAAHo/0rvYwua1k-w/s1600/EmployeeSearch.png |

**Here is a youtube video that I have recorded on SQL Injection. Hope, you will find it useful.**  
   
  
The HTML for the **Employee Serach Page**is shown below. As you can see from the HTML, the **Employee Serach Page** contains TextBox, Button and a GridView control.

|  |
| --- |
| **Employee Search Page HTML** http://2.bp.blogspot.com/-MJBISzaxxs0/TiiG2ZsYp5I/AAAAAAAAAHs/4Jsrgzo0nfo/s1600/EmployeeSearchPageHTML.png |

The **codebehind** page for the EmployeeSearchPage is shown below.

|  |
| --- |
| **Employee Search Page Code Behind** http://2.bp.blogspot.com/-2Qf8EyCN1lE/TiiHztglZbI/AAAAAAAAAHw/ZQvrFMxo6vg/s1600/EmployeeSearchPageCodeBehind.png |

The **Button1\_Click** event handler has the required ADO.NET code to get data from the database. This code is highly **susceptible to sql injection attack**and I will never ever have code like this in production environment. The **second line in Button1\_Click event handler**, dynamically builds the sql query by concatenating the **Employee ID** that we typed into the **TextBox**.   
  
So, for example, if we had typed 2 into the **Employee ID textbox**, we will have a SQL query as shown below.  
**Select \* from Employees where Id=2**  
  
If a malicious user, types something like **2; Delete from Employees** into the TextBox, then we will have a SQL query as shown below.  
**Select \* from Employees where Id=2; Delete from Employees**  
  
When this query is executed, we loose all the data in the **Employees** table. This is **SQL Injection Attack**, as the user of the application is able to inject SQL and get it executed against the database. It is very easy to avoid SQL Injection attacks by using either **parameterized queries** or using **stored procedures**.  
  
You may be thinking, how will the user of the application know the name of the table. Well, one way is to **simply guess**or **inject a sql syntax error**. The injected SQL syntax error causes the page to crash and can possibly reveal the name of the table as shown below. However, proper exception handling and custom error pages can be used to prevent the end user from seeing the yello screen of death. The screen shot below shows the table name **Employees**.

|  |
| --- |
| **Page crash revealing Employees table name** http://1.bp.blogspot.com/-TYBgb39io9E/TiiKLo-ODFI/AAAAAAAAAH0/90jJ4KtQtlA/s1600/PageCrash.png |

**To solve SQL injection attack, create a Stored Procedure as shown below.**  
Create Procedure spGetEmployees  
@Id int  
as  
Begin  
 Select \* from Employees where Id=@Id  
End  
  
**Modify the codebehind page for the EmployeeSearchPage, to use the stored procedure as shown below**

|  |
| --- |
| using System; using System.Data; using System.Data.SqlClient;   namespace TestWeb {     public partial class EmployeeSearch : System.Web.UI.Page     {         protected void Page\_Load(object sender, EventArgs e)         {                      }           protected void Button1\_Click(object sender, EventArgs e)         {             // Create the SQL Connection object.              SqlConnection con = new SqlConnection             ("server=localhost; database=TestDB; integrated security=SSPI");             // Create the SQL command object. Pass the stored procedure name              // as a parameter to the constructor of the SQL command class             SqlCommand cmd = new SqlCommand("spGetEmployees", con);             // Create the SQL parameter object, specifying the name and the value              // we want to pass to the SP.             SqlParameter paramId = new SqlParameter("@Id", txtEmployeeId.Text);             // Associate the Id parameter object with the command object, using              // parameters collection property of the SQL Command object.              cmd.Parameters.Add(paramId);             // Specify the command type as stored procedure. This tells the command              // object, that the command is a SQL stored procedure and not an adhoc sql query             cmd.CommandType = CommandType.StoredProcedure;             // Open the connection             con.Open();             // Execute the command and assign the returned results as the data source for              // the employyes girdview             gvEmployees.DataSource = cmd.ExecuteReader();             // Call the DataBind() method, to bind the results to the employees grid view control             gvEmployees.DataBind();             // Finally close the sql server connection object             con.Close();         }     } } |

**Explain Dependency Injection with an example**

One of the very common interview questions, asked these days. This is the most common approach used today to solve dependencies between objects. In many of the enterprise class ASP.NET application, Dependency Injection is a common standard to follow. Let us understand Dependency Injection with an example.

|  |
| --- |
| http://2.bp.blogspot.com/-TN2uNRQQdk8/Te_crUOZeGI/AAAAAAAAAGw/dOVAAsIE4Dw/s1600/DependencyInjection.png |

In the example above, Employee class depends on EmployeeDAL class to get the data from the database. In GetAllEmployees() method of theEmployee class, we create an instance of the EmployeeDAL (Employee Data Access Layer) class and then invoke SelectAllEmployees() method. This is tight coupling, EmployeeDAL is tightly copuled with the Employee class. Everytime the EmployeeDAL class changes, the Employee class also needs to change. EmployeeDAL cannot be mocked and hence unit testing becomes cumbersome and time consuming.   
  
The same example can be re-written using dependency injection as shown below. First thing to notice is that, we are using **interface types instead of concrete types**. Using interfaces help us to plugin any implemenation of the interface, with less or no code modification at all. We are not creating the instance of the EmployeeDAL in the Employee class, instead we are passing it as a parameter to the constructor of the Employee class. As, we are injecting an instance of a class into a class that depends on it, we can call this process as **Dependency Injection**.

|  |
| --- |
| http://1.bp.blogspot.com/-9LCE73kiDPI/Te_dF_wLMwI/AAAAAAAAAG0/W9Cj-MK-liw/s1600/ConstructorInjection.png |

Dependency Injection is of 2 types.  
**1.** Constructor Injection  
**2.** Setter Injection.  
  
We have already seen how to use Constructor Injection in the example above. An, example for Setter Injection is shown below. We are injecting an object instance through the Setter property, instead of a constructor. Hence, we call Setter Injection. It is very important to use the propertyEmployeeDataObject to access the instance of IEmployeeDAL, rather than the private variable employeeDAL. The property checks to see ifemployeeDAL is null, and throws the exception accordingly.

|  |
| --- |
| http://4.bp.blogspot.com/-FI2QmuyR12A/Te_dpr3i5tI/AAAAAAAAAG4/CgAxXvv9uT0/s1600/SetterInjection.png |

For other follow up interview questions on Dependency Injection, please read the articles below.   
[1. When do you choose Setter Injection over Constructor Injection and vice versa ?](http://venkataspinterview.blogspot.com/2011/06/interview-questions-related-to.html)   
[2. What are the advantages of using Dependency Injection ?](http://venkataspinterview.blogspot.com/2011/06/interview-questions-related-to.html)   
[3. What Dependency Injection Container have you used in your project ?](http://venkataspinterview.blogspot.com/2011/06/interview-questions-related-to.html)

**Interview Questions related to Dependency Injection**

[It is very important that, you understand the concept of dependency injection before you read these questions. Please read Dependency Injection with an example article if you have not done so already.](http://venkataspinterview.blogspot.com/2011/06/explain-dependency-injection-with.html)   
  
**When do you use Setter Injection over Constructor Injection and vice versa ?**  
Constructor Injection is normally the standard for dependency injection. Constructor Injection, ensures that all the dependency objects are initialized before we attempt to invoke any methods or peroperties, thus avoiding null reference exceptions.   
  
Setter Injection is used rarely in exceptional cases. For example, if I have a class that has several methods which does not depend on any other objects, and now I need to introduce a new method with in this class, which depends on another object. We can use, constructor injection here, but we need to change the exisitng constructor calls at all the other places. This can become cumbersome, especially if the project is a huge one. Hence, Setter Injection could be a choice when we want to change the legacy code.   
  
Setter Injection, enables you to inject the object, just before you need it. So, if the object is an expensive object to create, and you want to create it as late as possible, then setter injection is your choice.

**What are the advantages of using Dependency Injection?**  
**1.** Dependency Injection allows us to develop very loosely coupled systems.  
  
**2.** Easy to swap in a different implementation of a component, as long as the component implements the interface type.  
  
**3.** Dependency Injection, allows objects to be mocked with in the Unit Tests. This is the greatest advantage of Dependency Injection.  
  
**What Dependency Injection Container have you used in your project?**  
There are lot of Dependency Injection Containers available in the market. I have included, the very commonly used list of Dependency Injection Containers for your quick refernce.  
**1.** Unity  
**2.** Castle Windsor   
**3.** StructureMap   
**4.** Spring.NET   
  
**I have used Unity in most of my projects.  
  
If you can improve any of these answer further, please feel free to do so by submitting the form below. Thank you very much for your contribution.**

**What is the process for strong naming an assembly**

|  |
| --- |
| **What is the process for strong naming an assembly ?** or **What is the purpose of strong naming tool ( sn.exe ) in .NET ?**  In .NET, the **assembly name**usually consists of **4 parts** as listed below. **1.** Simple Textual Name **2.** Version Number (The version number is also divided into 4 parts) **3.** Culture **4.** Public Key Token   If an assembly contains, all the 4 parts, then the assembly is a **strongly named assembly**, other wise the assembly is called as a **weak named assembly**. In general, when you compile any .NET application, the generated assembly by default will have the Simple Textual Name, Version Number and Culture but not the public key token. If you have to sign the assembly with a public key token, you first have to generate the key pair using **key generation tool**called **strong naming tool (sn.exe)**. The generated key pair will consist of a **private and a public key** and are written into a key file. **Key files** have the extension of **.snk**.   We now have to associate the **key file**with the **project**, so that when we compile the project, the generated assembly is signed using the key pair. To do this, In **AssemblyInfo.cs** file of the project, specify **AssemblyKeyFile** attribute as shown below.               [assembly: AssemblyKeyFile("MyKey.snk")]  The last and final step is to build the project which will automatically sign the assembly using the key file. This process generates the strongly named assembly.  **In short, there are 3 simple steps to generate a strongly named assembly.** **1.** Generate the key pair using strong naming tool, SN.exe.  **2.** Associate the generated Key file to the project using AssemblyKeyFile, which is present in AssemblyInfo.cs file.  **3.** Build the project.  **Once, you have strongly named the assembly, you can copy it to GAC. There are 2 ways to copy an assembly into GAC.** **1.** **Using simple drag and drop** : Drag the generated assembly into the GAC folder. Usually the path for GAC is **c:\windows\assembly**. On some machines this could be **c:\winnt\assembly**.  **2. Use GAC utility** : Use GAC Utility tool(**gacutil.exe**) as shown below in visual studio command prompt.                 **gacutil.exe -i C:\MyAssembly.dll (- i stands for install)**  Once, you have successfully copied the assembly into GAC, notice the four parts of the assembly name. The **culture** column could be empty, indicating that the assembly is language neutral.   Please click here to read few other follow up interview questions related to strong named assemblies.  **If you can improve this answer further, please feel free to do so by submitting the form below. Thank you very much for your contribution.** |

**Interview Questions related to strong named assemblies**

**Can you copy a weak named assembly into GAC?**  
**No**, an assembly has to be strongly named to be copied into the GAC

**What is the difference between a strong and weak named assemblies?**  
The following are the differences between a strong and weak named assemblies  
**1.** Weak named assemblies can be duplicated and tampered with, where as strong named assemblies cannoth be tampered and duplicated.   
  
**2.** Strong named assemblies can be copied into GAC, where as weak named assemblies cannot be copied.  
  
**3.** A single copy of strong named assembly present in the GAC can be shared with multiple applications, where as weak named assembly must be copied into the bin directory of each project.  
  
**What is the main advantage of strong naming an assembly?**  
A shared assembly that is strongly named solves the DLL hell problem, that microsoft is notoriously known for. Please follow the articles below, to understand DLL hell and the solution.  
[What is DLL hell](http://venkataspinterview.blogspot.com/2011/05/what-is-dll-hell-in-net.html)  
[How is DLL hell problem solved](http://venkataspinterview.blogspot.com/2011/06/how-is-dll-hell-problem-solved-in-net.html)  
  
**Can a strong named assembly refer, a weak named assembly and why?**  
No, a strong named assembly cannot reference, another weak named assembly. If this is possible, the DLL hell problem will reoccur.   
  
**Explantion:** When you reference a strong named assembly, you expect to get certain benefits, such as versioning and naming protection. If the strong named assembly then references an assembly with a simple name, which does not have these benefits, you lose the benefits you would derive from using a strong named assembly and revert to DLL conflicts. Therefore, strong named assemblies can only reference other strong named assemblies.  
  
**What is the downside of strong naming an executable (.EXE) ?**  
A strongly named assembly cannot reference a weak named assembly. Therefore, strong naming an EXE prevents the EXE from referencing weak named dlls, that are deployed with the application. For this reason, the Visual Studio project system does not strong name application EXEs. Instead, it strong-names the Application manifest, which internally points to the weak named application EXE. In addition, you may want to avoid strong-naming components that are private to your application. In this case, strong-naming can make it more difficult to manage dependencies and add unnecessary overhead for private components.  
  
**If you can improve any of these answer further, please feel free to do so by submitting the form below. Thank you very much for your contribution.**

**What is the difference between layers and tiers**

**Layers** refer to logical seperation of code. **Logical layers** help you organise your code better. For example an application can have the following layers.  
  
  
**1)**Presentation Layer or UI Layer  
**2)**Business Layer or Business Logic Layer  
**3)**Data Access Layer or Data Layer  
  
  
The aboove three **layers** reside in their own projects, may be 3 projects or even more. When we compile the projects we get the respective layer DLL. So we have 3 DLL's now.  
  
  
Depending upon how we deploy our application, we may have **1 to 3 tiers**. As we now have 3 DLL's, if we deploy all the DLL's on the same machine, then we have only **1 physical tier but 3 logical layers.**  
  
  
If we choose to deploy each DLL on a seperate machine, then we have **3 tiers** and **3 layers**.  
  
  
So, **Layers** are a logical separation and **Tiers** are a physical separation. We can also say that, **tiers** are the physical deployment of **layers**.  
  
  
   
**Tiers:**  
**1)** Presenation Tier or UI Tier (Hosts the Presentation Layer or UI Layer). This can be considered as web server in case of an ASP.NET web application.  
**2)** Application Tier or Business Tier (Hosts Business Layer or Business Logic Layer).   
**3)** Data Access Tier or Data Tier (Hosts Data Access Layer or Data Layer).  
**4)** Database Tier - SQL Server or Oracle (or any other database) which has tables, stored procedures and other database objects.  
  
  
**In general the following are the responsibilities of each layer or tier:**  
  
  
**1)**Presentation Layer or Tier is usually responsible for interacting with the user.  
**2)**Business Layer or Tier is responsible for implementing the business logic of the application.  
**3)**Data Access Layer or Tier is responsible for encapsulating the code that accesses the persistent data stores such as a relational database.

**What are the advantages and disadvantages of a layered architecture**

**The following are the advantages of a layered architecture:**  
  
**Layered architecture increases flexibility, maintainability, and scalability.** In a Layered architecture we separate the user interface from the business logic, and the business logic from the data access logic. Separation of concerns among these logical layers and components is easily achieved with the help of layered architecture.

**Multiple applications can reuse the components.** For example if we want a windows user interface rather than a web browser interface, this can be done in an easy and fast way by just replacing the UI component. All the other components like business logic, data access and the database remains the same. Layered architecture allows to swap and reuse components at will.  
  
**Layered architecture** enables teams to work on different parts of the application parallely with minimal dependencies on other teams.  
  
**Layered architecture**enables develop loosely coupled systems.  
  
**Different components** of the application can be independently deployed, maintained, and updated, on different time schedules.  
  
**Layered architecture**also makes it possible to configure different levels of security to different components deployed on different boxes. sO Layered architecture, enables you to secure portions of the application behind the firewall and make other components accessible from the Internet.  
  
**Layered architecture**also helps you to test the components independently of each other.  
  
**The following are the disadvantages of a layered architecture:**  
  
**There might be a negative impact**on the performance as we have the extra overhead of passing through layers instead of calling a component directly.   
  
**Development of user-intensive** applications can sometime take longer if the layering prevents the use of user interface components that directly interact with the database.  
  
**The use of layers helps** to control and encapsulate the complexity of large applications, but adds complexity to simple applications.  
  
**Changes to lower level interfaces**tend to percolate to higher levels, especially if the relaxed layered approach is used.

**What is the difference between const and static read-only member?**

**A const field must be initialized at the place where it is declared as shown in the example below.**  
class Program  
{  
      public const int Number = 100;  
}  
**It is a compile time error to declare a const without a value. The code below will generate a compiler error stating "A const field requires a value to be provided"**  
class Program  
{  
      public const int Number;  
}  
**It is a compile time error to change the value of a constant. The following code will generate a compiler error stating "The left-hand side of an assignment must be a variable, property or indexer"**  
class Program  
{  
     public const int Number = 100;  
     static void Main()  
     {  
           Number = 200;  
     }  
}  
  
  
   
**It is not mandatory to initialize a static readonly field where it is declared. You can declare a static readonly field without an initial value and can later initialize the static field in a static constructor as shown below.**  
class Program  
{  
     public static readonly int Number;  
     static Program()  
     {  
           Number = 100;  
     }  
}  
  
  
**Once a static readonly field is initialized, the value cannot be changed. The code below will generate a compiler error stating "A static readonly field cannot be assigned to (except in a static constructor or a variable initializer)"**  
class Program  
{  
       public static readonly int Number;  
       static Program()  
       {  
             Number = 100;  
       }  
       static void Main()  
       {  
              Number = 200;  
        }  
}  
  
  
**In short, the difference is that static readonly field can be modified by the containing class, but const field can never be modified and must be initialized where it is declared. A static readonly field can be changed by the containing class using static constructor as shown below.**  
class Program  
{  
       // Initialize the static readonly field   
       // to an initial value of 100  
          public static readonly int Number=100;  
          static Program()  
          {  
               //Value changed to 200 in the static constructor  
                 Number = 200;  
          }  
}

**Best pactices in developing asp.net applications - Part 1**

1. Remove unused private fields and functions.  
  
2. Do not cast unnecessarily. Avoid duplicate casts where possible, since there is a cost associated with them.  
  
3. Properties that return arrays are prone to code inefficiencies. Consider using a collection or making this a method.  
  
4. To test for empty strings, check if String.Length is equal to zero. Constructs such as "".Equals(someString) and String.Empty.Equals(someString) are less efficient than testing the string length. Replace these with checks for someString.Length == 0.  
  
5. Methods in the same type that differ only by return type can be difficult for developers and tools to properly recognize. When extending a type, be sure not to define new methods that differ from base type methods only by type.  
   
6. Use stringbuilder instead of string types for string manipulation.   
  
7. Use String.Format instead of concatenating and appending strings.  
  
8. Use Type.TryParse rather than Convert.ToDestinationType(). For example use int.TryParse() rather than Convert.ToInt32() which might throw an exception.  
  
9. Override Equals() method wherever applicable in your classes.  
  
10. Consider passing base types as parameters - Using base types as parameters to methods improves re-use of these methods if you only use methods & properties from the parameter's base class. E.g. use Stream instead of FileStream as a parameter when only calling Stream.Read(), this makes the method work on all kind of streams instead of just File streams.

**Best pactices in developing asp.net applications - Part 2**

1. Do not catch general exception types - You should not catch Exception or SystemException. Catching generic exception types can hide run-time problems from the library user, and can complicate debugging. You should catch only those exceptions that you can handle gracefully.  
  
2. Use properties instead of visible instance fields.  
  
3. Follow the same naming conventions accross the solution.   
  
4. Remove unwanted commented code, Indent code properly.  
  
5. Use curly braces with in an if statement, even if there is a single statement in the if block. This will provide better readability.  
  
6. Make sure to refactor your code to move the duplicated code to common reusable functions.  
   
7. Move one time control settings into the .aspx page rather than having them in the code behind in if(!IsPostback) block.  
  
8. Use inheritance whereever possible, which enables code reuse and also reduces the amount of code we have to write and test.  
  
9. Move the reusable javascript functions to an external .js file instead of having them on the page.  
  
10. For controls that are declarativley specified on the page, tie the event handlers to the controls events on the aspx page rather than initializing them in the codebehind. If the controls are built dynamically then we donot have a choice.  
  
11. Make sure to check for nulls when using any type retrieved from a session, querystring or a database to avoid NullReferenceExceptions.  
  
12. Use foreach loop instead of using for loop which may lead to out of boundary run time exceptions.

**ASP.NET Interview Questions on Data Access Security**

**What are the best practices to follow to secure connection strings in an ASP.NET web application?**  
**1.**Always store connection strings in the site's Web.config file. Web.config is very secure. Users will not be able to access web.config from the browser.  
**2.** Do not store connection strings as plain text. To help keep the connection to your database server secure, it is recommended that you encrypt connection string information in the configuration file.  
**3.** Never store connection strings in an aspx page.  
**4.**Never set connection strings as declarative properties of the SqlDataSource control or other data source controls.

**Why is "Connecting to SQL Server using Integrated Security" considered a best practice?**Connecting to SQL Server using integrated security instead of using an explicit user name and password, helps avoid the possibility of the connection string being compromised and your user ID and password being exposed.  
  
**What is the advantage of storing an XML file in the applications App\_Data folder?**  
The contents of the App\_Data folder will not be returned in response to direct HTTP requests.  
   
**What is Script injection?**  
A script injection attack attempts to send executable script to your application with the intent of having other users run it. A typical script injection attack sends script to a page that stores the script in a database, so that another user who views the data inadvertently runs the code.**What is SQL injection?**A SQL injection attack attempts to compromise your database by creating SQL commands that are executed instead of, or in addition to, the commands that you have built into your application.**What are the best practices to keep in mind when accepting user input on a web application?  
1.** Always use validation controls whenever possible to limit user input to acceptable values.  
**2.** Always check the IsValid property of the aspx page. Run the server side code only if the IsValid property value is true. A value of false means that one or more validation controls have failed a validation check.  
**3.** Always perform server side validation irrespective of client side validation being performed or not. This will protect your web application even if the client has by passed the client side validation by disabling javascript in the web browser.  
**4.**Also make sure to re validate user input in the business logic layer of your application.**What are the steps to follow to avoid Script Injection attacks?  
1.** Encode user input with the HtmlEncode method. This method turns HTML into its text representation.  
**2.** If you are using the GridView control with bound fields, set the BoundField object's HtmlEncode property to true. This causes the GridView control to encode user input when the row is in edit mode.**What are the steps to follow to avoid SQL Injection attacks?**Always use parameterized queries or stored procedures instead of creating SQL commands by concatenating strings together.  
  
**Can you encrypt view state data of an aspx page?**  
Yes, you encrypt view state data of an aspx page by setting the page's ViewStateEncryptionMode property to true.

**ASP.NET Interview Questions on HTTP modules and HTTP Handlers**

**What is an HTTP Handler?**  
An ASP.NET **HTTP handler**is the process (frequently referred to as the "endpoint") that runs in response to a request made to an ASP.NET Web application. The most common handler is an ASP.NET page handler that processes .aspx files. When users request an .aspx file, the request is processed by the page through the page handler. You can create your own HTTP handlers that render custom output to the browser.  
  
**What is HTTP module?**An **HTTP module** is an assembly that is called on every request that is made to your application. HTTP modules are called as part of the ASP.NET request pipeline and have access to life-cycle events throughout the request. HTTP modules let you examine incoming and outgoing requests and take action based on the request.  
  
**What is the interface that you have to implement if you have to create a Custom HTTP Handler?**  
Implement **IHttpHandler** interface to create a synchronous handler.  
Implement **IHttpAsyncHandler**to create an asynchronous handler.  
  
**What is the difference between asynchronous and synchronous HTTP Handlers?**A **synchronous handler**does not return until it finishes processing the HTTP request for which it is called.  
  
An **asynchronous handler** runs a process independently of sending a response to the user. Asynchronous handlers are useful when you must start an application process that might be lengthy and the user does not have to wait until it finishes before receiving a response from the server.  
  
**Which class is responsible for receiving and forwarding a request to the appropriate HTTP handler?**  
IHttpHandlerFactory Class  
   
**Can you create your own custom HTTP handler factory class?**  
Yes, we can create a custom **HTTP handler**factory class by creating a class that implements the IHttpHandlerFactory interface.  
  
**What is the use of HTTP modules?**  
HTTP modules are used to implement various application features, such as forms authentication, caching, session state, and client script services.  
  
**What is the difference between HTTP modules and HTTP handlers?**An **HTTP handler** returns a response to a request that is identified by a file name extension or family of file name extensions. In contrast, an HTTP module is invoked for all requests and responses. It subscribes to event notifications in the request pipeline and lets you run code in registered event handlers. The tasks that a module is used for are general to an application and to all requests for resources in the application.  
  
**What is the common way to register an HTTP module?**The common way to register an HTTP module is to have an entry in the application's Web.config file.  
  
**Much of the functionality of a module can be implemented in a global.asax file. When do you create an HTTP module over using Global.asax File?**  
You create an **HTTP module** over using Global.asax file if the following conditions are true  
  
1. You want to re-use the module in other applications.  
2. You want to avoid putting complex code in the Global.asax file.  
3. The module applies to all requests in the pipeline.

**ASP.NET Interview Questions on Themes and Skins**

**What is a "theme" in ASP.NET?**A "theme" is a collection of property settings that allow you to define the look of pages and controls, and then apply the look consistently across pages in a Web application, across an entire Web application, or across all Web applications on a server.  
  
**What is the extension for a skin file?**  
.skin  
  
**What are the 2 types of control skins in ASP.NET?  
1.**Default skins  
**2.** Named skins  
  
**What is the difference between Named skins and Default skins?**A **default skin** automatically applies to all controls of the same type when a theme is applied to a page. A control skin is a default skin if it does not have a SkinID attribute. For example, if you create a default skin for a Calendar control, the control skin applies to all Calendar controls on pages that use the theme. (Default skins are matched exactly by control type, so that a Button control skin applies to all Button controls, but not to LinkButton controls or to controls that derive from the Button object.)  
  
A **named skin** is a control skin with a SkinID property set. Named skins do not automatically apply to controls by type. Instead, you explicitly apply a named skin to a control by setting the control's SkinID property. Creating named skins allows you to set different skins for different instances of the same control in an application.  
  
**What are the 3 levels at which a theme can be applied for a web application?  
1**. At the page level - Use the Theme or StyleSheetTheme attribute of the @ Page directive.  
  
**2.** At the application level - Can be applied to all pages in an application by setting the <pages> element in the application configuration file.  
  
**3.** At the web server level - Define the <pages> element in machine.config file. This will apply the theme to all the web applications on that web server.  
  
**What is the name of the folder that contains the application themes?**App\_Themes  
   
**What is a global theme?**  
A **global theme**is a theme that you can apply to all the Web sites on a server. Global themes allow you to define an overall look for your domain when you maintain multiple Web sites on the same server.  
  
**What is the difference between themes and CSS?**  
**1.** Themes can define many properties of a control or page, not just style properties. For example, using themes, you can specify the graphics for a TreeView control, the template layout of a GridView control, and so on.  
  
**2.** Themes can include graphics.  
  
**3.** Themes do not cascade the way style sheets do. By default, any property values defined in a theme referenced by a page's Theme property override the property values declaratively set on a control, unless you explicitly apply the theme using the StyleSheetTheme property.  
  
**4.**Only one theme can be applied to each page. You cannot apply multiple themes to a page, unlike style sheets where multiple style sheets can be applied.  
  
**What are the security concerns to keep in mind when using themes?**Themes can cause security issues when they are used on your Web site. Malicious themes can be used to:  
  
**1.** Alter a control's behavior so that it does not behave as expected.  
  
**2.** Inject client-side script, therefore posing a cross-site scripting risk.  
  
**3.** Expose sensitive information.  
  
**4.**The mitigations for these common threats are:  
  
**5.** Protect the global and application theme directories with proper access control settings. Only trusted users should be allowed to write files to the theme directories.  
  
**6.**Do not use themes from an untrusted source. Always examine any themes from outside your organization for malicious code before using them on you Web site.  
  
**7.** Do not expose the theme name in query data. Malicious users could use this information to use themes that are unknown to the developer and thereby expose sensitive information.

**ASP.NET Interview Questions on DataSet**

**What is a DataSet?**  
DataSet is an in-memory cache of data.  
  
**In which namespace is the DataSet class present?**  
System.Data  
  
**Can you add more than one table to a dataset?**  
Yes  
  
**Can you enforce constarints and relations on tables inside a DataSet?**  
Yes, the DataSet consists of a collection of DataTable objects that you can relate to each other with DataRelation objects. You can also enforce data integrity in the DataSet by using the UniqueConstraint and ForeignKeyConstraint objects.  
  
**What happens when you invoke AcceptChanges() method on a DataSet?**  
Invoking AcceptChanges() method on the DataSet causes AcceptChanges() method to be called on each table within the DataSet.  
  
Both the DataRow and DataTable classes also have AcceptChanges() methods. Calling AcceptChanges() at the DataTable level causes the AcceptChanges method for each DataRow to be called.  
  
When you call AcceptChanges on the DataSet, any DataRow objects still in edit-mode end their edits successfully. The RowState property of each DataRow also changes. Added and Modified rows become Unchanged, and Deleted rows are removed.  
  
If the DataSet contains ForeignKeyConstraint objects, invoking the AcceptChanges method also causes the AcceptRejectRule to be enforced.  
  
**Is there a way to clear all the rows from all the tables in a DataSet at once?**  
Yes, use the DataSet.Clear() method to clear all the rows from all the tables in a DataSet at once.  
  
**What is the difference between DataSet.Copy() and DataSet.Clone()?**  
DataSet.Clone() copies the structure of the DataSet, including all DataTable schemas, relations, and constraints. Does not copy any data.  
  
DataSet.Copy() copies both the structure and data.  
  
**How do you get a copy of the DataSet containing all changes made to it since it was last loaded?**  
Use DataSet.GetChanges() method  
  
**What is the use of DataSet.HasChanges() Method?**  
DataSet.HasChanges method returns a boolean true if there are any changes made to the DataSet, including new, deleted, or modified rows. This method can be used to update a DataSource only if there are any changes.  
  
**How do you roll back all the changes made to a DataSet since it was created?**  
Invoke the DataSet.RejectChanges() method to undo or roll back all the changes made to a DataSet since it was created.  
  
**What happnes when you invoke RejectChanges method, on a DataSet that contains 3 tables in it?**  
RejectChanges() method will be automatically invoked on all the 3 tables in the dataset and any changes that were done will be rolled back for all the 3 tables.  
  
When the DataTable.RejectChanges method is called, any rows that are still in edit-mode cancel their edits. New rows are removed. Modified and deleted rows return back to their original state. The DataRowState for all the modified and deleted rows will be flipped back to unchanged.  
  
**What is the DataSet.CaseSensitive property used for?**  
When you set the CaseSensitive property of a DataSet to true, string comparisons for all the DataTables within dataset will be case sensitive. By default the CaseSensitive property is false.

**ASP.NET Interview Questions on Globalization**

**What is Globalization?**  
Globalization is the process of creating an application that meets the needs of users from multiple cultures. This process involves translating the user interface elements of an application into multiple languages, using the correct currency, date and time format, calendar, writing direction, sorting rules, and other issues. Accommodating these cultural differences in an application is called localization.  
  
The Microsoft .NET Framework simplifies localization tasks substantially by making its formatting, date/time, sorting, and other classes culturally aware. Using classes from the System.Globalization namespace, you can set the application’s current culture, and much of the work is done automatically!  
  
**What are the 3 different ways to globalize web applications?  
  
Detect and redirect approach :** In this approach we create a separate Web application for each supported culture, and then detect the user’s culture and redirect the request to the appropriate application. This approach is best for applications with lots of text content that requires translation and few executable components.  
  
**Run-time adjustment approach :**In this approach we create a single Web application that detects the user’s culture and adjusts output at run time using format specifiers and other tools. This approach is best for simple applications that present limited amounts of content.  
  
**Satellite assemblies approach :** In this approach we create a single Web application that stores culture-dependent strings in resource files that are compiled into satellite assemblies. At run time, detect the user’s culture and load strings from the appropriate assembly. This approach is best for applications that generate content at run time or that have large executable components.  
  
**In ASP.NET, how do you detect the user's language preference on his/her computer?**  
Use the Request object’s UserLanguages property to return a list of the user’s language preferences. The first element of the array returned by UserLanguages is the user’s current language on his/her computer.  
  
**What are the steps to follow to get user's culture at run time?**  
To get the user’s culture at run time, follow these steps:  
**1.** Get the Request object’s UserLanguages property.  
**2.** Use the returned value with the CultureInfo class to create an object representing the user’s current culture.  
  
For example, the following code gets the user’s culture and displays the English name and the abbreviated name of the culture in a label the first time the page is displayed:  
private void Page\_Load(object sender, System.EventArgs e)  
{  
// Run the first time the page is displayed  
if (!IsPostBack)  
{  
// Get the user's preferred language.  
string sLang = Request.UserLanguages[0];  
// Create a CultureInfo object from it.  
CultureInfo CurrentCulture = new CultureInfo(sLang);  
lblCulture.Text = CurrentCulture.EnglishName + ": " +  
CurrentCulture.Name;  
}  
}  
  
**What are the advantages of using detect and redirect approach to globalizing web applications?**  
**1.** Content is maintained separately, so this approach allows the different applications to present very different information, if needed.  
**2.** Users can be automatically directed to sites that are likely to be geographically close, and so can better meet their needs.  
**3.** Content files (Web forms and HTML pages, for example) can be authored in the appropriate natural language without the complexity of including resource strings.  
  
**What are the disadvantages of using detect and redirect approach to globalizing web applications?**  
**1.** Using this approach requires that the executable portion of the Web application be compiled and deployed separately to each culture-specific Web site.  
**2.** This approach requires more effort to maintain consistency and to debug problems across Web sites.  
  
**What is the use of culture attribute of the globalization element in web.config?**  
The Web.config file’s globalization element is used to create a culture-specific Web application. The culture attribute of the globalization element specifies how the Web application deals with various culture-dependent issues, such as dates, currency, and number formatting.  
  
Web.config globalization settings in subordinate folders override the globalization settings in the application’s root Web.config file. You can store content for various cultures in subfolders within your application, add Web.config files with the globalization settings for each culture, then direct users to the appropriate folder based on the user’s CurrentCulture.  
  
**The text on the webform is usually written from left to right. How do you change the writing direction to "right to left"?**  
The wrting direction of a webform can be changed using the HTML dir attribute as shown below.  
**<body dir="rtl">**  
  
You can use the dir attribute individually in panels, text boxes, or other controls as well. Setting the dir attribute on the body element applies right-to-left formatting to the entire page.  
  
**What do you mean by neutral cultures?**  
Neutral cultures represent general languages, such as English or Spanish, rather than a specific language and region. When you set the culture attribute for a Web application in Web.config, ASP.NET assigns that culture to all the threads running for that Web application. Threads are the basic unit to which the server allocates processor time. ASP.NET maintains multiple threads for a Web application within the aspnet\_wp.exe worker process.  
  
**What are advantages of setting the culture dynamically at the thread level over creating separate Web applications for each culture?**  
**1.** All cultures share the same application code, so the application doesn’t have to be compiled and deployed for each culture.  
**2.** The application resides at a single Web address, you don’t need to redirect users to other Web applications.  
**3.** The user can choose from a full array of available cultures.  
  
**For what type of web applications setting the culture dynamically is best suited?**  
Setting the culture dynamically is best suited for simple Web applications that don’t contain large amounts of text that must be translated into different languages.

**C# Interview Questions - Arrays**

**What is the difference between arrays in C# and arrays in other programming languages?**  
Arrays in C# work similarly to how arrays work in most other popular languages There are, however, a few differences as listed below  
  
**1.**When declaring an array in C#, the square brackets ([]) must come after the type, not the identifier. Placing the brackets after the identifier is not legal syntax in C#.

**int[] IntegerArray; // not int IntegerArray[];  
  
2.** Another difference is that the size of the array is not part of its type as it is in the C language. This allows you to declare an array and assign any array of int objects to it, regardless of the array's length.  
  
int[] IntegerArray; // declare IntegerArray as an int array of any size  
IntegerArray = new int[10]; // IntegerArray is a 10 element array  
IntegerArray = new int[50]; // now IntegerArray is a 50 element array  
  
**What are the 3 different types of arrays that we have in C#?  
1.** Single Dimensional Arrays  
**2.** Multi Dimensional Arrays also called as rectangular arrays  
**3.** Array Of Arrays also called as jagged arrays  
  
**Are arrays in C# value types or reference types?**Reference types.  
  
**What is the base class for all arrays in C#?**System.Array  
  
**How do you sort an array in C#?**The Sort static method of the Array class can be used to sort array items.  
  
**Give an example to print the numbers in the array in descending order?**using System;  
namespace ConsoleApplication  
{  
class Program  
{  
static void Main()  
{  
int[] Numbers = { 2, 5, 3, 1, 4 };  
//Print the numbers in the array without sorting  
Console.WriteLine("Printing the numbers in the array without sorting");  
foreach (int i in Numbers)  
{  
Console.WriteLine(i);  
}  
//Sort and then print the numbers in the array  
Console.WriteLine("Printing the numbers in the array after sorting");  
Array.Sort(Numbers);  
foreach (int i in Numbers)  
{  
Console.WriteLine(i);  
}  
//Print the numbers in the array in desceding order  
Console.WriteLine("Printing the numbers in the array in desceding order");  
Array.Reverse(Numbers);  
foreach (int i in Numbers)  
{  
Console.WriteLine(i);  
}  
}  
}  
}  
  
**What property of an array object can be used to get the total number of elements in an array?**  
Length property of array object gives you the total number of elements in an array. An example is shown below.  
using System;  
namespace ConsoleApplication  
{  
class Program  
{  
static void Main()  
{  
int[] Numbers = { 2, 5, 3, 1, 4 };  
Console.WriteLine("Total number of elements = " +Numbers.Length);  
}  
}  
}  
  
**Give an example to show how to copy one array into another array?**We can use CopyTo() method to copy one array into another array. An example is shown below.  
using System;  
namespace ConsoleApplication  
{  
class Program  
{  
static void Main()  
{  
int[] Numbers = { 2, 5, 3, 1, 4 };  
int[] CopyOfNumbers=new int[5];  
Numbers.CopyTo(CopyOfNumbers,0);  
foreach (int i in CopyOfNumbers)  
{  
Console.WriteLine(i);  
}  
}  
}  
}

[[http://img1.blogblog.com/img/icon18_email.gif](http://www.blogger.com/email-post.g?blogID=5199087724247829980&postID=8403175727937283724)](http://www.blogger.com/email-post.g?blogID=5199087724247829980&postID=8403175727937283724)

**XML Interview Questions - Validating XML documents**

**What determines the validity of an XML document?**  
Document Type Definition(DTD) or an XML Schema determines the validity of an XML document.  
  
**What is a valid XML document?**XML documents are compared to rules that are specified in a DTD or schema. A well-formed XML document that meets all of the requirements of one or more specifications is called a valid XML Document.  
  
**What are the 2 types of XML parsers?**  
Nonvalidating Parsers - Parsers that don’t support validation  
Validating Parsers - Parsers that support validation  
  
**Can you combine both Schema and DTD references in a single XML document?**Yes  
  
**Are DTD's well-formed XML documents?**  
No, DTDs are not well-formed XML documents. This is because they follow DTD syntax rules rather than XML document syntax.  
  
**Are XML schema's well-formed XML documents?**Yes.  
   
**What is the difference between an XML schema and a DTD?**  
The XML Schema is the officially sanctioned Schema definition. Unlike DTDs, the format of XML Schemas follows the rules of well-formed XML documents. The Schema also allows for much more granular control over the data that is being described. Because of the XML format and the detailed format controls, Schemas tend to be very complex and often much longer than the XML documents that they are describing. Schemas are often much more easy for developers to read and follow,due to the less cryptic nature of the references in Schemas versus DTDs.  
  
**How do you define references to schemas in an XML document?**References to schemas are defined by creating an instance of the XMLSchemainstance namespace. An example is shown below.  
**<rootelement xmlns:xsi=”http://www.w3.org/2001/XMLSchemainstance” xsi:noNamespaceSchemaLocation=”schemafile.xsd”>**  
  
The namespace declaration reference to http://www.w3.org/2001/XMLSchemainstance resolves to an actual document at that location, which is a brief description of the way that the W3C Schema should be referenced. The noNamespaceSchemaLocation value tells us that there is no predefined namespace for the Schema. This means that all of the elements in the XML document should be validated against the schema specified. The location of the Schema is schemafile.xsd. Because there is no path defined, the file containing the schema should be located in the same directory as the XML file to be validated by the Schema.  
  
You can also define the schema location, and map it to a specific namespace by using the schemaLocation attribute declaration instead of noNamespace SchemaLocation. If you do so, you have to declare a namespace that matches the schemaLocation attribute value. The declaration must be made before you reference the schema in a schemaLocation attribute assignment.

**XML related Interview Questions**

**What Is XML?**  
**XML**stands for Extensible Markup Language, and it is used to describe documents and data in a standardized, text-based format that can be easily transported via standard Internet protocols. XML, like HTML, is based on, Standard Generalized Markup Language (SGML).  
  
**What are Well-formed XML documents?  
XML**, is very strict about a small core of format requirements that make the difference between a text document containing a bunch of tags and an actual XML document. XML documents that meet W3C XML document formatting recommendations are described as being well-formed XML documents. Well-formed XML documents can contain elements, attributes, and text.  
  
**What is an empty XML element?**  
Elements with no attributes or text are called as empty XML element. Empty XML elements can be represented in an XML document as shown below:  
**<element/>**  
  
**What is meant by XML document declaration?**Most XML documents start with an <?xml?> element at the top of the page. This is called an XML document declaration. An XML document declaration is an optional element that is useful to determine the version of XML and the encoding type of the source data. It is not a required element for an XML document to be well formed. Most common XML document declaration is shown below:  
<?xml version=”1.0” encoding=”UTF-8”?>  
  
**What does UTF stands for?**  
UTF stands for Universal Character Set Transformation Format.  
  
**Should every XML document have a root element?**Yes.  
  
**Can an XML document contain multiple root level elements?**  
No, an XML document can contain only one root level element.  
  
**What is the use of XML attributes?**XML attributes are used for adding more information and descriptions to the values of elements,and the text associated with elements.  
   
**Is XML case sensitive?**  
Yes  
  
**How do you comment lines in XML?**You can comment lines in XML as shown below.  
<! -- This is commented line in an XML document -->  
  
**What are XML namespaces?**  
Namespaces are a method for separating and identifying duplicate XML element names in an XML document. Namespaces can also be used as identifiers to describe data types and other information. Namespace declarations can be compared to defining a short variable name for a long variable (such as pi=3.14159....) in programming languages. In XML, the variable assignment is defined by an attribute declaration. The variable name is the attribute name, and the variable value is the attribute value. In order to identify namespace declarations versus other types of attribute declarations, a reserved xmlns: prefix is used when declaring a namespace name and value. The attribute name after the xmlns: prefix identifies the name for the defined namespace. The value of the attribute provides the unique identifier for the namespace. Once the namespace is declared, the namespace name can be used as a prefix in element names.  
  
**Why is it a good idea to use a URL as the XML namespace value?**  
Although the namespace declaration value does not need to be a URL or resolve to an actual URL destination, it is a good idea to use a URL anyway, and to choose a URL that could resolve to an actual destination, just in case developers want to add documentation for the namespace to the URL in the future.  
  
**When to use namespaces?**Namespaces are optional components of basic XML documents. However, namespace declarations are recommended if your XML documents have any current or future potential of being shared with other XML documents that may share the same element names. Also, newer XML-based technologies such as XML Schemas,SOAP, and WSDL make heavy use of XML namespaces to identify data encoding types and important elements of their structure.

**ASP.NET Interview Questions on caching application data**

**Which object can be used to store frequently used items in the server’s memory for quick retrieval?**  
Cache object can be used to store frequently used items in the server’s memory for quick retrieval.  
  
**Is the cache object available for all web forms with in a web application?**  
Yes, the Cache object is global, that is, data stored in the Cache object is available anywhere within a Web application. In this way, the Cache object is very similar to the intrinsic Application object.  
  
**What are the 3 different ways to store data in the Cache object?**  
**Use assignment.**  
Assigning a value to an unused key in the Cache object automatically creates that key and assigns the value to that key. Assigning a value to a key that already exists replaces the cached value with the assigned value.  
**Use the Insert method.**The Insert method uses parameters rather than assignment to create or change cached data. Insert optionally accepts parameters to establish dependencies and set expiration policy.  
**Use the Add method.**The Add method is similar to Insert; however, it requires all parameters and returns an object reference to the cached data.  
  
For example, the following Cache statements all add the same item to the cache:  
  
using System.Web.Caching;  
private void Page\_Load(object sender, System.EventArgs e)  
{   
if(!IsPostBack)   
{   
Cache["NewItem"] = "Some string data";   
Cache.Add("NewItem", "Some string data", null, Cache.NoAbsoluteExpiration, System.TimeSpan.FromMinutes(1), CacheItemPriority.Default, null);   
Cache.Insert("NewItem", "Some string data");   
}  
}  
  
  
  
  
**What are absoluteExpiration and slidingExpiration parmeters of the Insert and Add methods?**  
**absoluteExpiration**A DateTime object that identifies when the data should be removed from the cache. If you’re using sliding expiration, specify Cache.NoAbsoluteExpiration for this parameter.  
**slidingExpiration**A TimeSpan object that identifies how long the data should remain in the cache after the data was last accessed. If you’re using absolute expiration, specify Cache.NoSlidingExpiration for this parameter.  
  
**Which delegate can be used to notify the application when items are removed from the cache?**  
onRemoveCallback is used to notify the application when items are removed from the cache.  
  
**How do you retrieve the value of a cache item stored in the servers memory?**  
You can retrieve the value of a cache item stored in the servers memory through the item’s key, just as you do with the Application and Session objects. Because cached items might be removed from memory, you should always check for their existence before attempting to retrieve their value, as shown in the following code:  
  
private void Button1\_Click(object sender, EventArgs e)  
{   
if (Cache["ChachedItem"] == null)   
{   
Lable1.Text = "Cached Item not found.";   
}   
else   
{   
Lable1.Text = Cache["ChachedItem"].ToString();   
}  
}  
  
**Which method can be used to remove data from the cache?**  
Cache object’s Remove method can be used to remove data from the cache as shown in the following code example / sample.  
  
private void RemoveButton\_Click(object sender, System.EventArgs e)  
{   
Cache.Remove("CachedItem");  
}  
  
**How do you control how long data is cached?**  
The Cache object’s Add and Insert method parameters allow you to control how long an item is stored in the server’s memory. In practice, these parameter settings provide only indirect control of how long data remains in memory. If your server runs low on available memory, ASP.NET recovers as much memory as possible from expired cache items. If that’s not enough, ASP.NET will unload unexpired items from the cache based on their priority and when they were last accessed.  
  
**What is CacheItemPriority enumeration used for?**  
CacheItemPriority enumeration is used to set the relative importance of cached items. CacheItemPriority.NotRemoveable has the highest priority and CacheItemPriority.Low has the lowest priority.  
  
**Which is the only "event” provided by Cache object?**  
CacheItemRemoved "event” is the only "event” provided by Cache object.   
  
**How do you update the Cache object when data changes?**  
Items stored in the cache are often copies of data that is stored and maintained elsewhere, such as records in a database. Use the Add and Insert methods’ dependency parameter to establish a relationship between a cached data item and an external source, such as a file, a folder, or a group of files.  
  
The dependency parameter accepts a CacheDependency object, which in turn identifies the file, folder, or set of files to watch for changes. ASP.NET checks the time stamp of the items in the CacheDependency object, if one of those time stamps is later than the DateTime entered for the cached item, ASP.NET unloads that item from the cache.

**Application build related ASP.NET interview questions**

**What are the 2 build options for an ASP.NET web application?**

1. Debug  
2. Release  
  
**What are the 3 levels at which we can have a configuration file for an ASP.NET web application?**  
1. At the web server level : The Machine.config file located in the Windows\Microsoft.NET\Framework\version\config directory. This sets the base configuration for all .NET assemblies running on the server.  
2. At the application or web site level : The Web.config file located in the IIS root directory.This sets the base configuration for all Web applications and overrides settings in Machine.config.  
3. In the sub directory of an application root folder : These settings override the settings in the root folder's web.config file.  
  
**What happens when you make changes to an application’s Web.config file?**  
When you make changes to an application’s Web.config file, IIS automatically restarts the application and applies the changes. This has the side effect of resetting current Application or Session state variables, which can adversely affect users.  
  
**What happens when you access the Web.config file from a browser?**  
For security reasons, you can’t access the Web.config file from a browser. If a user requests the Web.config file from your Web site, he or she will receive an "This type of page is not served" error message.  
  
**What happens when you access the Global.asax file from a browser?**  
For security reasons, you can’t access the Global.asax file from a browser. If a user requests the Global.asax file from your Web site, he or she will receive an "This type of page is not served" error message.  
  
**What are the steps to follow to host a web application on a web server?**  
1. Use IIS to set up a virtual folder for the application.  
2. Copy the Web application to the virtual directory.  
3. Add any shared .NET components to the server’s global assembly cache (GAC).  
4. Set the security permissions on the server to allow the application to access required resources.  
  
**What are the 2 components that should be installed on a web server where, the ASP.NET web application runs?**  
ASP.NET Web applications run under IIS, so both IIS and the Microsoft .NET Framework must be installed on the server before that server can host Web applications.  
  
What is the purpose of default start page in IIS?  
A default page enables IIS to display a page if the user does not specify one in his or her request. For example, a user might make a request using only your domain name, such as <http://www.venkataspinterview.blogspot.com/>. If a default page is enabled, IIS will respond with <http://www.venkataspinterview.blogspot.com/default.aspx>.  
  
Don’t confuse IIS default pages with the Web Forms application start page in Visual Studio .NET. Visual Studio .NET requires that you set a start page for each project so that the development environment knows which page to display first during debugging. This setting has no effect on the IIS default page settings.  
  
**How do you copy the COM component to the server and register?**  
COM components generally provide a setup program to install or remove them from the system. If the component doesn’t provide a setup program, you can copy it to the server and register it using the MFC RegSvr32.exe utility, as shown below:  
*RegSvr32 MyComname.dll*  
  
**What is GAC?**  
GAC stands for Global Assembly Cache. The global assembly cache (GAC) is a special subfolder within the Windows folder that stores the shared .NET components.  
  
**What is the difference between weak-named .NET components and strong-named .NET components?**  
This difference is how the names are stored within the assembly. Weak names are not guaranteed to be unique and thus cannot be shared without potentially causing conflicts. Strong names are digitally signed and provide a public key that ensures there are no conflicts. Furthermore, .NET components with strong names can’t call unmanaged code (such as COM components) and thus avoid potential conflicts with dependencies.  
  
Weak-named .NET components must be individually copied to the /bin directories of the Web applications where they are used. Strong-named .NET components can be copied into the server’s GAC  
  
**What is the account under which the ASP.NET worker process run?**  
By default, the ASP.NET worker process runs using the ASPNET account, which is created when you install the .NET Framework. This account has limited privileges, which can cause permission-denied errors if your application writes files or tries to read files outside the current Web application’s boundaries.  
  
What are the 3 ways in which you can modify the additional permissions required by your application.?  
1. Grant the ASPNET user access to the required files. To use this option, the server must be using the Windows NT file system (NTFS).  
2. Change the group the ASPNET user belongs to.  
3. Use impersonation to run the process as another user.  
  
**Why is it not a good idea to add ASPNET user to the Administrators group?**  
Adding the ASPNET user to the Administrators group gives your Web application full privileges on the server; however, it also poses a potential security risk because outside users might be able to manipulate your application to hack your server.  
  
**How do you impersonate the ASP.NET worker process?**To use impersonation to run the ASP.NET worker process as a user other than ASPNET, set the identity element’s impersonation attribute in the application’s Web.config file.

**ASP.NET Interview Questions on Master Pages**

**What are Master Pages in ASP.NET? or What is a Master Page?**  
ASP.NET master pages allow you to create a consistent layout for the pages in your application. A single master page defines the look and feel and standard behavior that you want for all of the pages (or a group of pages) in your application. You can then create individual content pages that contain the content you want to display. When users request the content pages, they merge with the master page to produce output that combines the layout of the master page with the content from the content page.  
  
**What are the 2 important parts of a master page?**  
The following are the 2 important parts of a master page  
1. The Master Page itself  
2. One or more Content Pages  
  
**Can Master Pages be nested?**  
Yes, Master Pages be nested.  
  
**What is the file extension for a Master Page?**  
.master  
  
**How do you identify a Master Page?**  
The master page is identified by a special @ Master directive that replaces the @ Page directive that is used for ordinary .aspx pages.  
  
**Can a Master Page have more than one ContentPlaceHolder?**  
Yes, a Master Page can have more than one ContentPlaceHolder  
  
**What is a ContentPlaceHolder?**  
ContentPlaceHolder is a region where replaceable content will appear.  
  
**How do you bind a Content Page to a Master Page?**  
MasterPageFile attribute of a content page's @ Page directive is used to bind a Content Page to a Master Page.  
  
**Can the content page contain any other markup outside of the Content control?**  
No.  
  
**What are the advantages of using Master Pages?**  
1. They allow you to centralize the common functionality of your pages so that you can make updates in just one place.  
2. They make it easy to create one set of controls and code and apply the results to a set of pages. For example, you can use controls on the master page to create a menu that applies to all pages.  
3. They give you fine-grained control over the layout of the final page by allowing you to control how the placeholder controls are rendered.  
4. They provide an object model that allows you to customize the master page from individual content pages.  
  
**What are the 3 levels at which content pages can be attached to Master Page?**  
At the page level - You can use a page directive in each content page to bind it to a master page  
  
At the application level - By making a setting in the pages element of the application's configuration file (Web.config), you can specify that all ASP.NET pages (.aspx files) in the application automatically bind to a master page.  
  
At the folder level - This strategy is like binding at the application level, except that you make the setting in a Web.config file in one folder only. The master-page bindings then apply to the ASP.NET pages in that folder.  
  
**What is @MasterType directive used for?**  
@MasterType directive is used to create a strongly typed reference to the master page.  
  
**Are controls on the master page accessible to content page code?**  
Yes, controls on the master page are accessible to content page code.  
  
**At what stage of page processing master page and content page are merged?**  
During the initialization stage of page processing, master page and content page are merged.  
  
**Can you dynaimically assign a Master Page?**  
Yes, you can assign a master page dynamically during the PreInit stage using the Page class MasterPageFile property as shown in the code sample below.  
void Page\_PreInit(Object sender, EventArgs e)  
{  
this.MasterPageFile = "~/MasterPage.master";  
}  
  
**Can you access non public properties and non public methods of a master page inside a content page?**  
No, the properties and methods of a master page must be public in order to access them on the content page.  
  
**From the content page code how can you reference a control on the master page?**  
Use the FindControl() method as shown in the code sample below.  
void Page\_Load()  
{  
// Gets a reference to a TextBox control inside  
// a ContentPlaceHolder  
ContentPlaceHolder ContPlaceHldr = (ContentPlaceHolder)Master.FindControl ("ContentPlaceHolder1");  
if(ContPlaceHldr != null)  
{  
TextBox TxtBox = (TextBox)ContPlaceHldr.FindControl("TextBox1");  
if(TxtBox != null)  
{  
TxtBox.Text = "TextBox Present!";  
}  
}  
// Gets a reference to a Label control that not in  
// a ContentPlaceHolder  
Label Lbl = (Label)Master.FindControl("Label1");  
if(Lbl != null)  
{  
Lbl.Text = "Lable Present";  
}  
}  
  
**Can you access controls on the Master Page without using FindControl() method?**  
Yes, by casting the Master to your MasterPage as shown in the below code sample.  
protected void Page\_Load(object sender, EventArgs e)  
{  
MyMasterPage MMP = this.Master;  
MMP.MyTextBox.Text = "Text Box Found";  
}

**ASP.NET Interview Questions on Passport Authentication**

**What is Passport Authentication?**  
**Passport authentication**identifies users via Microsoft Passport’s single sign-on service. Microsoft Passport is meant to provide Internet users with a single identity that they can use to visit a wide variety of Web sites that require authentication. Information about the user is available to your application through a profile that is stored with Microsoft.  
  
**What are the advantages of Passport authentication?**  
The advantages of Passport authentication are that the user doesn’t have to remember separate user names and passwords for various Web sites and that the user can maintain his or her profile information in a single location. Passport authentication also provides access to other Microsoft services, such as Passport Express Purchase.  
  
**What is passport software development kit (passport SDK)?**  
To use Passport authentication in your Web application, you must install the Passport SDK. The Passport SDK is free for preproduction development and testing. To deploy a site for public use, you must obtain an annual license from Microsoft.  
  
**How does Passport authentication work?**  
When a user accesses an application that implements Passport authentication, ASP.NET checks the user’s machine for a current passport authentication cookie. If none is found, ASP.NET directs the user to a Passport sign-on page. Once the user signs in, the Passport service authenticates the user, stores an authentication cookie on the user’s computer, and directs the user back to the originally requested Web page.  
  
**What are the steps to follow to use Passport authentication?**  
1. Install the Passport SDK. Passport is not included with Visual Studio, although the .NET Framework does include classes for working with the Passport SDK once it is installed.  
2. Set the application’s authentication mode to Passport in Web.config. Set authorization to deny unauthenticated users.  
3. Use the PassportAuthentication\_OnAuthenticate event to access the user’s Passport profile to identify and authorize the user.  
4. Implement a sign-out procedure to remove Passport cookies from the user’s machine.  
  
**Where is PassportAuthentication\_OnAuthenticate event present?**  
PassportAuthentication\_OnAuthenticate event is present in Global.asax.

**SSL and HTTPS related ASP.NET Interview Questions**

**How do you provide Secure Communication over the world wide web?**  
Security is not just a matter of identifying users and preventing unauthorized users from accessing your Web applications, but it’s just as important to ensure that sensitive data sent across the Internet can’t be read by others.  
  
To provide secure communication across the Internet, IIS supports a standardized means of encrypting and decrypting Web requests and responses. This cryptography requires that you request an encryption key called a server certificate from an independent third party called a certificate authority.  
  
**What is Secure Sockets Layer (SSL)?**  
**The Secure Sockets Layer (SSL)**is the standard means of ensuring that data sent over the Internet can’t be read by others. When a user requests a secure Web page, the server generates an encryption key for the user’s session and then encrypts the page’s data before sending a response. On the client side, the browser uses that same encryption key to decrypt the requested Web page and to encrypt new requests sent from that page.  
  
**Explain the process of secure communication using SSL?**  
Using SSL in your application requires special authorization from a recognized certificate authority. This authorization comes in the form of a server certificate, which you install in IIS to identify your server. The certificate authority licenses server certificates (for a fee) and acts as a clearinghouse to verify your server’s identity over the Internet.  
  
When a user’s browser begins secure communications, it requests the server certificate and checks it against a list of trusted sites provided by the certificate authority. If the server certificate does not match one of the sites already authorized by the user, or if the server certificate does not match the Web address for which it was registered, or if there are any other problems with the server certificate, the browser displays a warning.  
  
In this way, the certificate authority not only provides encryption for secure data transmission, but it also provides assurance to users that your Web site is authentic.  
  
**What is the largest certificate authority?**  
The largest certificate authority is VeriSign.  
  
**What are the steps to follow to use SSL in your Web application?**  
1. Generate a certificate request from IIS.  
2. Request a certificate from a certificate authority.  
3. Install the certificate on the server using IIS.  
4. Install the certificate on browsers if you are using a test certificate.  
5. Use the Secure Hypertext Transfer Protocol (HTTPS) when accessing secure pages in your application.  
  
**What should you do before you can request a server certificate from a certificate authority?**  
Before you can request a server certificate from a certificate authority, you must generate a certificate request from IIS. The certificate request contains encrypted information about your server that the certificate authority uses to identify your server over the Internet.  
  
**What are the steps to follow to generate a certificate request from the IIS?**  
1. Select Default Web Site in the console tree of the IIS, and then choose Properties from the Action menu. IIS displays the Default Web Site Properties dialog box.  
2. Click the Directory Security tab in the Properties dialog box, and then click Server Certificate. IIS starts the Web Server Certificate Wizard.  
3. Step through the wizard by reading each screen and clicking Next. The wizard instructions are straightforward.  
4. When you click Finish at the end, the wizard creates an encrypted text file with the .cer file extension. That file is the certificate request that you send to the certificate authority.  
  
**Why do you have to select Default Web Site when generating a Certificate Request from IIS?**  
IIS requires that a certificate be created at the server root before secure communications can be created or configured for subordinate sites on the server. That’s why you have to select Default Web Site (or the root Web site if you have renamed it). After you have installed a server certificate at the root, you can repeat the process for subordinate sites if you want separate certificates for those sites.  
  
**What is the file extension of a server certificate?**  
.cer  
  
**What are the steps to follow to install the Certificate to enable SSL for your Web applications?**  
To install a server certificate in IIS:  
1. Select Default Web Site in the console tree of the IIS snap-in, and then choose Properties from the Action menu. IIS displays the Default Web Site Properties dialog box.  
2. Click the Directory Security tab in the Properties dialog box, and then click Server Certificate. IIS starts the Web Server Certificate Wizard.  
3. Click Next, and select Process The Pending Request And Install The Certificate.  
4. Click Next, and enter the name of the certificate file.  
5. Click Next, and then click Finish to complete the installation.  
  
**What is the protocol on which secure pages are generally requested?**  
HTTPS, the protocol HTTPS is what initializes the secure communication. When you’ve begun secure communication, it continues until you specify a nonsecure site.  
  
**What are the steps to follow to make a web page secure in a web application?**  
To require secure communication for a Web page using IIS, follow these steps  
1. Select the folder or file that requires secure communication, and then choose Properties from the Action menu. IIS displays the Properties dialog box.  
2. Click the Directory Security tab, and then click Edit in the Secure Communications group. IIS displays the Secure Communications dialog box.  
3. Select the Require Secure Channel (SSL) check box, and click OK.  
  
**Can a user access secure web page over HTTP protocol instead of HTTPS?**  
No, When you require secure communication for a Web page, that page can’t be viewed using HTTP. The user must type in or click a link using HTTPS, otherwise, access will be denied.

**ASP.NET Forms Authentication related interview questions**

[**Back to the list of all ASP.NET interview questions**](http://venkataspinterview.blogspot.com/2008/07/aspnet-interview-questions.html) **What is the advantage of using Forms authentication?**  
The advantage of using Forms authentication is that users do not have to be member of a domain-based network to have access to your application. Another advantage is that many Web applications, particularly commercial sites where customers order products, want to have access to user information. Forms authentication makes these types of applications easier to create.  
  
**List the steps to use Forms authentication in a web application?**  
1.Set the authentication mode in Web.config to Forms.  
2.Create a Web form to collect logon information.  
3.Create a file or database to store user names and passwords.  
4.Write code to add new users to the user file or database.  
5.Write code to authenticate users against the user file or database.  
  
**What happens when someone accesses a Web application that uses Forms authentication?**  
When someone accesses a Web application that uses Forms authentication, ASP.NET displays the logon Web form specified in Web.config. Once a user is authorized, ASP.NET issues an authorization certificate in the form of a cookie that persists for an amount of time specified by the authentication settings in Web.config.  
  
**What is the difference between Windows authentication and Forms authentication?**  
The difference between Windows authentication and Forms authentication is that in Forms authentication your application performs all the authentication and authorization tasks. You must create Web forms and write code to collect user names and passwords and to check those items against a list of authorized users.  
  
**What is the use of mode attribute in authentication element in a web.config file?**  
You use the mode attribute to specify the type of authentication your web application is using. Set the mode attribute to forms to enable Forms authentication.  
  
**What is the use of name attribute and loginUrl attribute of a forms element in a web.config file?**  
**Name attribute**of forms element is used to set the name of the cookie in which to store the user’s credential. The default is .authaspx. If more than one application on the server is using Forms authentication, you need to specify a unique cookie name for each application.  
**loginUrl** attribute of forms element is used to set the name of the Web form to display if the user has not already been authenticated. If omitted, the default is Default.aspx.  
  
**What is protection attribute in a forms element used for in web.config file?**  
The protection attribute of a forms element of web.config file is used for setting how ASP.NET protects the authentication cookie stored on the user’s machine. The default is **All**, which performs encryption and data validation. Other possible settings are**Encryption, Validation, and None**.  
  
**What is timeout attribute in a forms element used for in web.config file?**  
Timeout attribute is used to set the number of minutes the authentication cookie persists on the user’s machine. The default is 30, indicating 30 minutes. ASP.NET renews the cookie automatically if it receives a request from the user and more than half of the allotted time has expired.  
  
**In which namespace the FormsAuthentication class is present?**  
System.Web.Security namespace  
  
**Which method checks the user name and password against the user list found in the credentials element of Web.config?**  
The FormsAuthentication class’s Authenticate method checks the user name and password against the user list found in the credentials element of Web.config.  
  
**Which method can be used to remove forms authentication cookie?**  
Use the signout() method of FormsAuthentication class to sign out when the user has finished with the application or when you want to remove the authentication cookie from his or her machine. For example, the following code ends the user’s access to an application and requires him or her to sign back in to regain access  
***FormsAuthentication.SignOut();***  
  
**What is the advantage of Authenticating Users with a Database?**  
You can authenticate users based on a list in Web.config. The FormsAuthentication class’s Authenticate method is set up to read from web.config file automatically. That’s fine if user names and passwords are created and maintained by a system administrator, but if you allow users to create their own user names or change their passwords, you’ll need to store that information outside the Web.config file. This is because changing Web.config at run time causes the Web application to restart, which resets any Application state and Session state variables used by the application.  
  
**What are the advantages of storing user names and passwords in a database rather than a file?**  
You can store user names and passwords in any type of file; however, using a database has the following significant advantages:  
1. User names can be used as primary keys to store other information about the user.  
2. Databases can provide high performance for accessing user names and passwords.  
3. Adding, modifying, and accessing records are standardized through SQL.  
  
**Can you encrypt user names and passwords stored in a file or a database?**  
Yes, you encrypt user names and passwords stored in a file or a database. You can encrypt them using the FormsAuthentication class’s **HashPasswordForStoringInConfigFile**method. This method uses the **SHA1** or **MD5**algorithms to encrypt data, as shown below:  
***Password = FormsAuthentication.HashPasswordForStoringInConfigFile(Password, "SHA1");***  
  
**Can you change authentication type in a subfolder's web.config file?**  
Authentication type (Windows, Forms, or Passport) can be set only at the application’s root folder. To change authentication type in a subfolder's web.config file, you must create a new Web application project and application starting point for that subfolder.  
  
**How can you control access to subfolders in a web application?**  
The authorization settings in the Web.config file apply hierarchically within the folder structure of a Web application. For instance, you might want to allow all users access to the root folder of a Web application but restrict access to Web forms (and tasks) available from a subfolder. To do this, set the authentication type in the root folder’s Web.config file, and then use the authorization element in the subfolder’s Web.config file to restrict access.

**ASP.NET Interview Questions on windows authentication**

**What is the advantage of using Windows authentication in a Web application?**Windows authentication uses the security features integrated into the Windows NT and Windows XP operating systems to authenticate and authorize Web application users. The advantage of Windows authentication is that your Web application can use the exact same security scheme that applies to your corporate network - user names, passwords, and permissions are the same for network resources and Web applications. One of the key advantages of Windows authentication is that users who are logged on to the network don’t have to log on again to access the Web application.  
  
**What is the default authentication method when you create a new Web application project?**  
Windows authentication is the default authentication method when you create a new Web application project.  
  
**How do you allow or deny access to specific users using an authorization list from Web.config file, when using windows authentication?**  
When the application uses Windows authentication, ASP.NET checks the project’s Web.config authorization list to see which network users are allowed to access the application. The asterisk (\*) and question mark (?) characters have special meaning in the authorization list. The \* character indicates all users. The ? character indicates unauthenticated users.  
  
To restrict access to specific users, list their names separated by commas in an element. When ASP.NET checks the authorization list in Web.config, it accepts the first match that it finds. Be sure to end the authorization list with a element to deny access to any nonapproved users.  
   
**What is Role-Based authorization in windows authentication?**Role-based authorization lets you identify groups of users to allow or deny based on their role in your organization. In Windows NT and Windows XP, roles map to names used to identify user groups. Windows defines several built-in groups, including Administrators, Users, and Guests. You can view, modify, or add groups using the Computer Management console  
  
To allow or deny access to certain groups of users, add the element to the authorization list in your Web application’s Web.config file.  
  
**How do you get a User Identity?**Once a user is authenticated and authorized, your application can get information about the user by using the User object’s Identity property. The Identity property returns an object that includes the user name and role information, as shown in the following code:  
  
private void Page\_Load(object sender, System.EventArgs e)  
{  
Label1.Text = User.Identity.IsAuthenticated.ToString();  
Label2.Text = User.Identity.Name;  
Label3.Text = User.Identity.AuthenticationType;  
}  
  
**How do you determine, what is the role of the current user?**  
The User object provides an IsInRole method to determine the role of the current user, as shown in the following example:  
if(User.IsInRole("Administrators"))  
{  
// Do something.  
}  
  
**Can you specify authorization settings both in Web.config and in IIS?**  
Yes, you can specify authorization settings both in Web.config and in IIS. The IIS setting is evaluated first and then the setting in Web.config is evaluated. In general, this means that the most restrictive setting will be used.  
  
**What is the user account under which an ASP.NET web application runs by default?**Web application runs under the identity of the ASPNET user account by default.  
  
**How can you set the web application to run under a specific user’s account?**  
You can set the application to run under a specific user’s account by setting the application’s identity element to enable impersonation  
  
**How can you see the impersonated identity under which code is executing?**To see the impersonated identity under which code is executing, use the WindowsIdentity class’s GetCurrent method, as shown in the sample code below  
Response.Write(System.Security.Principal.WindowsIdentity.GetCurrent().Name);   
  
The identity element can be used with any type of authentication; however, it is most useful with Windows authentication because Windows authentication users have accounts with specific permissions.

**Frequently asked ADO.NET Interview Questions**

**How do you create an instance of SqlDataReader class?**To create an instance of SqlDataReader class, you must call the ExecuteReader method of the SqlCommand object, instead of directly using a constructor.  
//Error! Cannot use SqlDataReader() constructor  
//to create an instance of SqlDataReader class  
SqlDataReader ReaderObject = new SqlDataReader();  
  
//Call the ExecuteReader method of the SqlCommand object  
SqlCommand CommandObject = new SqlCommand();  
SqlDataReader ReaderObject = CommandObject.ExecuteReader();  
  
Creating an instance of SqlDataReader class using SqlDataReader() constructor generates a compile time error - The type 'System.Data.SqlClient.SqlDataReader' has no constructors defined.  
  
**How do you programatically check if a specified SqlDataReader instance has been closed?**Use the IsClosed property of SqlDataReader to check if a specified SqlDataReader instance has been closed. If IsClosed property returns true, the SqlDataReader instance has been closed else not closed.  
  
**How do you get the total number of columns in the current row of a SqlDataReader instance?**  
FieldCount property can be used to get the total number of columns in the current row of a SqlDataReader instance.  
  
**Give an example for executing a stored procedure with parameters?**//Create the Connection Object  
SqlConnection ConnectionObject = new SqlConnection(ConnectionString);  
//Create the Command Object  
SqlCommand CommandObject = new SqlCommand("StoredProcedureName", ConnectionObject);  
//Specify to CommandObject that you intend to execute a Stored Procedure  
CommandObject.CommandType = CommandType.StoredProcedure;  
//Create an SQL Parameter object  
SqlParameter ParameterObject = new SqlParameter();  
//Specify the name of the SQL Parameter  
ParameterObject.ParameterName = "Parameter1";  
//Assign the Parameter value  
ParameterObject.Value = "Some Value";  
//Specify the Database DataType of the Parameter  
ParameterObject.DbType = DbType.String;  
//Specify the type of parameter - input-only, output-only, bidirectional  
ParameterObject.Direction = ParameterDirection.Input;  
//Associate the Parameter to the Command Object  
CommandObject.Parameters.Add(ParameterObject);  
//Open the connection  
ConnectionObject.Open();  
//Execute the command  
int Records\_Affected = CommandObject.ExecuteNonQuery();  
//Close the Connection  
ConnectionObject.Close();  
  
  
  
**What is the use of SqlParameter.Direction Property?**SqlParameter.Direction Property is used to specify the Sql Parameter type - input-only, output-only, bidirectional, or a stored procedure return value parameter. The default is Input.  
  
**How do you retrieve two tables of data at the same time by using data reader?**  
Include 2 select statements either in a stored procedure or in a select command and call the ExecuteReader() method on the command object. This will automatically fill the DataReader with 2 Tables of data.  
  
The datareader will always return the data from first table only. If you want to get the second table then you need to use ReaderObject.NextResult() method. The NextResult() method will return true if there is another table. The following code shows you how do it.  
//Create the SQL Query with 2 Select statements  
string SQLQuery = "Select \* from Customers;Select \* from Employees;";  
//Create the Connection Object  
SqlConnection ConnectionObject = new SqlConnection(ConnectionString);  
//Create the Command Object  
SqlCommand CommandObject = new SqlCommand(SQLQuery, ConnectionObject);  
//Open the connection  
ConnectionObject.Open();  
//Execute the command. Now reader object will have 2 tables of data.  
SqlDataReader ReaderObject = CommandObject.ExecuteReader();  
//Loop thru the tables in the DataReader object  
while (ReaderObject.NextResult())  
{  
while (ReaderObject.Read())  
{  
//Do Something  
}  
}  
//Close the Reader  
ReaderObject.Close();  
//Close the Connection  
ConnectionObject.Close();  
  
**What are the advantages of using SQL stored procedures instead of adhoc SQL queries in an ASP.NET web application?**  
**Better Performance :** As stored procedures are precompiled objects they execute faster than SQL queries. Every time we run a SQL query, the query has to be first compiled and then executed where as a stored procedure is already compiled. Hence executing stored procedures is much faster than executing SQL queries.  
**Better Security :**For a given stored procedure you can specify who has the rights to execute. You cannot do the same for an SQL query. Writing the SQL statements inside our code is usually not a good idea. In this way you expose your database schema (design) in the code which may be changed. Hence most of the time programmers use stored procedures instead of plain SQL statements.  
**Reduced Network Traffic :**Stored Procedures reside on the database server. If you have to execute a Stored Procedure from your ASP.NET web application, you just specify the name of the Stored Procedure. So over the network you just send the name of the Stored Procedure. With an SQL query you have to send all the SQL statements over the network to the database server which could lead to increased network traffic.  
  
**Can you update the database using DataReader object?**No, You cannot update the database using DataReader object. DataReader is read-only, foward only. It reads one record at a time. After DataReader finishes reading the current record, it moves to the next record. There is no way you can go back to the previous record.  
  
**What is the difference between a DataReader and a DataSet?**  
**DataReader  
1.** DatReader works on a Connection oriented architecture.  
**2.** DataReader is read only, forward only. It reads one record at atime. After DataReader finishes reading the current record, it moves to the next record. There is no way you can go back to the previous record. So using a DataReader you read in forward direction only.  
**3.** Updations are not possible with DataReader.  
**4.** As DataReader is read only, forward only it is much faster than a DataSet.  
**DataSet  
1.** DataSet works on disconnected architecture.  
**2.** Using a DataSet you can move in both directions. DataSet is bi directional.  
**3.** Database can be updated from a DataSet.  
**4.** DataSet is slower than DataReader.  
  
**Give an example scenario of using a DataSet and a DataReader?**If you want to just read and display the data(No updates, deletes, or inserts) then use a DataReader.  
If you want to do a batch inserts, updates and deletes then use a DataSet.

**Basic ADO.NET Interview Questions**

**What is Microsoft ADO.NET?**Visual Studio .NET provides access to databases through the set of tools and namespaces collectively referred to as Microsoft ADO.NET  
  
**What are the 3 major types of connection objects in ADO.NET?**  
**OleDbConnection object :** Use an OleDbConnection object to connect to a Microsoft Access or third-party database, such as MySQL. OLE database connections use the OleDbDataAdapter object to perform commands and return data.  
**SqlConnection object :** Use a SqlConnection object to connect to a Microsoft SQL Server database. SQL database connections use the SqlDataAdapter object to perform commands and return data.  
**OracleConnection object :** Use an OracleConnection object to connect to Oracle databases. Oracle database connections use the OracleDataAdapter object to perform commands and return data. This connection object was introduced in Microsoft .NET Framework version 1.1.  
  
**List the 4 common ADO.NET Namespaces?  
System.Data :** Contains Classes, types, and services for creating and accessing data sets and their subordinate objects  
**System.Data.SqlClient :** Contains Classes and types for accessing Microsoft SQL Server databases  
**System.Data.OracleClient :** Contains Classes and types for accessing Oracle databases (Microsoft .NET Framework version 1.1 and later)  
**System.Data.OleDb :** Contains Classes and types for accessing other databases  
  
**List all the steps in order, to access a database through ADO.NET?  
1.** Create a connection to the database using a connection object.  
**2.** Invoke a command to create a DataSet object using an adapter object.  
**3.**Use the DataSet object in code to display data or to change items in the database.  
**4.** Invoke a command to update the database from the DataSet object using an adapter object.  
**5.** Close the database connection if you explicitly opened it in step 2 using the Open method. Invoking commands without first invoking the Open method implicitly opens and closes the connection with each request.  
  
**Why will you usually create an ASPNET user account in the Database for an ASP.NET web application?**Web applications run using the ASPNET user account. The SQL database administrator will have to set up this account and grant it permissions before your Web application will have access to a SQL database. For file-based databases, such as Microsoft Access, you must grant permissions on the database file to the ASPNET user account using Windows file security settings.  
  
**What is the difference between DataReader and DataAdapter?**  
**1.**Data Reader is read only forward only and much faster than DataAdapter.  
**2.** If you use DataReader you have to open and close connection explicitly where as if you use DataAdapter the connection is automatically opened and closed.  
**3.** DataReader is connection oriented where as Data Adapter is disconnected  
  
**Can you inherit from SqlConnection Class?**No, you cannot inheirt from SqlConnection Class. SqlConnection Class is a sealed class. It is a compile time error.  
  
**Will the connection be closed, if the SqlConnection object goes out of scope?**  
No, If the SqlConnection goes out of scope, it won't be closed. Therefore, you must explicitly close the connection by calling Close or Dispose.  
  
**What happens if connection pooling is enabled?**If connection pooling is enabled and when you call Close or Dispose methods, then the connection is returned to the connection pool. This connection can then be resused.If connection pooling is disabled and when you call Close or Dispose methods, the underlying connection to the server is actually closed.  
  
**How do you ensure that the database connections are always closed?**  
To ensure that the database connections are always closed, open the connection inside of a using block, as shown in the following code fragment. Doing so ensures that the connection is automatically closed when the code exits the block.  
using (SqlConnection ConnectionObject = new SqlConnection())  
{  
ConnectionObject.Open();  
//The database connection will be closed when the control exits the using code block  
}  
  
**How do you read an XML file into a DataSet?**Using the DataSet object’s ReadXML method.  
  
**When do you use ExecuteReader, ExecuteNonQuery, ExecuteScalar methods?**If the command or stored procedure that is being executed returns a set of rows, then we use ExecuteReader method.  
If the command or stored procedure that is being executed returns a single value then we use ExecuteScalar method.  
If the command or stored procedure performs INSERT, DELETE or UPDATE operations, then we use ExecuteNonQuery method. ExecuteNonQuery method returns an integer specifying the number of rows inserted, deleted or updated.  
  
**Can your class inherit from SqlCommand Class?**  
No, you cannot inheirt from SqlCommand Class. SqlCommand Class is a sealed class. It is a compile time error.  
  
**Give an example that shows how to execute a stored procedure in ADO.NET?**using (SqlConnection ConnectionObject = new SqlConnection())  
{  
//Specify the name of the stored procedure to execute and the Connection Object to use  
SqlCommand CommandObject = new SqlCommand("StoredProcedureName", ConnectionObject);  
//Specify the SQL Command type is a stored procedure  
CommandObject.CommandType = CommandType.StoredProcedure;  
//Open the connection  
ConnectionObject.Open();  
//Execute the Stored Procedure  
int RecordsAffected = CommandObject.ExecuteNonQuery();  
}  
  
**Can you reuse a SqlCommand object?**  
Yes, you can reset the CommandText property and reuse the SqlCommand object.  
  
**What are the methods that can ensure asynchronous execution of the Transact-SQL statement or stored procedure?**BeginExecuteNonQuery  
BeginExecuteReader  
  
**What is SqlCommand.CommandTimeout Property used for?**  
CommandTimeout Property is used to Get or set the wait time before terminating the attempt to execute a command and generating an error.  
//Specify the CommandTimeout property value  
SqlCommand CommandObject = new SqlCommand("StoredProcedureName", ConnectionObject);  
//Wait for 10 seconds to execute the Stored procedure  
CommandObject.CommandTimeout = 10;   
The time is in seconds. The default is 30 seconds.

**Interview Questions on ASP.NET controls**

**What are the 2 types of controls that you can use on a webform in ASP.NET?**Web Server Controls  
HTML Controls  
  
**What’s the difference between Server controls and HTML controls?**  
**1.** Server controls can trigger control-specific events on the server.HTML controls can trigger only page- level events on server (postback).  
**2.** Data entered in a server control is maintained across requests. Server controls retain state.Data is not maintained in an HTML control. Data must be saved and restored using page-level scripts.  
**3.** The Microsoft .NET Framework provides a set of properties for each server control. Properties allow you to change the server control’s appearance and behavior within server-side code.HTML controls have HTML attributes only.  
**4.** Server controls automatically detect browser and adapt display as appropriate.HTML controls do not adapt automatically. You must detect browser in code or write for least common denominator.  
  
**What are the 2 Layouts supported by a Web form in ASP.NET?  
Grid layout -** Controls are placed exactly where you draw them, and they have absolute positions on the page. Use grid layout for Microsoft Windows–style applications, in which controls are not mixed with large amounts of text. Pages using grid layout will not always display correctly in non-Microsoft browsers.  
**Flow layout -**This layout positions controls relative to other elements on the page. If you add elements at run time, the controls that appear after the new element move down. Use flow layout for document-style applications, in which text and controls are intermingled.  
  
**When do you choose between GridLayout and Flow layout for Web forms?**You use GridLayout for Web forms that have a fixed appearance. You use FlowLayout for Web forms that incorporate text and controls.When you create controls with GridLayout, Visual Studio adds style attributes to each control that set the position of the control.When you create controls with FlowLayout, Visual Studio omits the style attribute.  
   
**Give 3 reasons why we use HTML controls over Server Controls?**  
**Migration from earlier versions of Active Server Pages (ASP) :** You can load an ASP application into Visual Studio and revise it gradually, rather than rewrite it completely. Earlier versions of ASP supported only HTML elements, and these elements become HTML controls when you load the project in Visual Studio .NET.  
**Not all controls require server-side events or state management :** This is particularly true when you’re doing data binding. Bound items are usually refreshed from the data source with each request, so it’s more efficient not to maintain state information for bound controls. This means that you can use HTML controls or turn off state management for bound server controls.  
**You have complete control over what is rendered with HTML controls :** ASP.NET adjusts the appearance of server controls based on the browser making the request. HTML controls are not adjusted, so you have direct control over their appearance.  
  
**How can you prevent users from editing Text in TextBox control on a web form?**By making the TextBox a readonly TextBox. To make a TextBox readonly set the ReadOnly property to True.  
  
**How do you convert an ASP.NET TextBox to accept passwords?**  
To convert and ASP.NET TextBox to accept passwords set the TextMode property to "Password"  
  
**What happens when you set the AutoPostBack property of a TextBox to true?**When AutoPostBack property is set to True, the TextBox control fires a TextChanged postback event when the user leaves the TextBox control after changing the contents. By default, this property is set to False and the Text­Changed event is cached until some other postback event occurs.  
  
**What are the 3 values that a TextMode property of TextBox can have?**  
**SingleLine :** Single Line TextBox  
**MultiLine :** Multi Line TextBox(scrollable)  
**Password :** When set to Password, the text box displays dots in place of the characters typed.  
  
**How do you limit the number of characters entered by a user in the ASP.NET TextBox?**By setting the MaxLength property of the TextBox. If you set the MaxLength property to 10, a user can enter only 10 characters into the TextBox.

**ASP.NET Interview Questions on Cookies**

**What are Cookies in ASP.NET?**  
Cookies are small pieces of information stored on the client computer.Use cookies to store small amounts of information on the client’s machine. Web sites often use cookies to store user preferences or other information that is client-specific. Because cookies can be refused, it is important to check whether the browser allows them before you try to create them.They are limited to storing only character data and they are limited to 4K in size.  
  
**What are different types of Cookies?**Session Cookies  
Persistent Cookies  
  
**What are Session Cookies?**  
Session cookies are stored in-memory during the client browser session. When the browser is closed the session cookies are lost.  
  
**How can you create Session Cookies?**You can create session cookies by calling the Add method of the Cookies collection on the Response object. The Cookies collection contains individual cookie objects of type HttpCookie.  
  
//Code to create a UserName cookie containing the name David.  
HttpCookie CookieObject = new HttpCookie("UserName", "David");  
Response.Cookies.Add(CookieObject);   
  
//Code to read the Cookie created above  
Request.Cookies["UserName"].Value;  
  
  
  
**What is the difference between Session Cookies and Persistent Cookies?**Persistent Cookies are same as Session Cookies except that, persistent cookies have an expiration date. The expiration date indicates to the browser that it should write the cookie to the client's hard drive. Keep in mind that because a user can delete cookies from their machine that there is no guarantee that a cookie you "drop" on a user machine will be there the next time they visit your site.  
  
**What are Persistent Cookies used for?**  
Persistent cookies are generally used to store information that identifies a returning user to a Web site. Typical information found in Persistent Cookies includes user names or user IDs.  
  
**How do you create a Persistent Cookie?**You create a persistent cookie the same way as session cookies except that you set the Expires property to a Date in the future which will store the Cookie to the client computer harddrive.  
  
//Code to create a UserName Persistent Cookie that lives for 10 days  
HttpCookie CookieObject = new HttpCookie("UserName", "David");  
CookieObject.Expires = DateTime.Now.AddDays(10);  
Response.Cookies.Add(CookieObject);   
  
//Code to read the Cookie created above  
Request.Cookies["UserName"].Value;   
  
**What is Cookie Dictionary?**  
A cookie dictionary is a single cookie object that stores multiple pieces of information. You use the Values property to access and assign new values to the cookie dictionary.  
  
**Give an example using Cookie Dictionary?**//Code to create a Cookie Dictionary  
HttpCookie CookieObject = new HttpCookie("UserPreference");   
  
//Use the Values property to assign new values to the cookie dictionary  
CookieObject.Values.Add("UserName", "David");  
CookieObject.Values.Add("Country", "USA");  
CookieObject.Values.Add("PreviousVisit", DateTime.Now.ToString());  
CookieObject.Expires = DateTime.MaxValue;   
  
//Add the Cookie to the client machine using the Response object  
Response.Cookies.Add(CookieObject);  
  
//Code to read the Cookie created above  
HttpCookie ObjectCookie = Request.Cookies["UserPreference"];  
string UserName = ObjectCookie.Values["UserName"];  
string Country = ObjectCookie.Values["Country"];  
string PreviousVisit = ObjectCookie.Values["PreviousVisit"];  
  
**What are the advantages of Using Cookies?**  
**1.** Cookies do not require any server resources since they are stored on the client.  
**2.** Cookies are easy to implement.  
**3.** You can configure cookies to expire when the browser session ends (session cookies) or they can exist for a specified length of time on the client computer (persistent cookies).  
  
**What are the disadvantages of Using Cookies?  
1.** Users can delete a cookies.  
**2.** Users browser can refuse cookies,so your code has to anticipate that possibility.  
**3.** Cookies exist as plain text on the client machine and they may pose a possible security risk as anyone can open and tamper with cookies.  
  
**How do you create a Cookie that never expires?**To create a Cookie that never expires set the Expires property of the Cookie object to DateTime.MaxValue.  
  
**Are Cookies secure?**  
No, Cookies are not secure. You must pay attention to the type of data you store in cookies.  
**1.** Cookies are not designed to store critical information so storing passwords in a cookie is a bad idea.  
**2.** Keep the lifetime of a cookie as short as practically possible.  
**3.** Encrypt cookie data to help protect the values stored in the cookie.

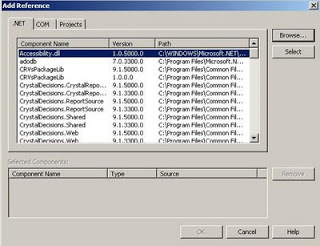
**ASP.NET Events related Interview Questions**

**What are the different levels at which events can occur in an ASP.NET web application?**Web application events occur at the application, page, and server control levels  
  
**Give some examples for application level events?**  
**1. Application\_Start**Occurs when the first user visits a page within your Web application.  
  
**2. Application\_End**  
Occurs when there are no more users of the application.  
  
**3. Application\_BeginRequest**  
Occurs when at the beginning of each request to the server. A request happens every time a browser navigates to any of the pages in the application.  
  
**4. Application\_EndRequest**  
Occurs when at the end of each request to the server.  
  
**5. Session\_Start**  
Occurs when a new user visits a page within your application.  
  
**6. Session\_End**  
Occurs when a user stops requesting pages from the Web application and their session times out. Sessions time out after a period specified in the Web.config file.  
  
**7. Application\_Error**  
Occurs when when there is an unhandled exception in an application.  
  
**Where are the application level event handlers present in an ASP.NET web application?**Application level event handlers are present in Global.asax of an ASP.NET web application  
  
**What is the difference between Application and Session Events?**  
At an application level we can have Application and Session events. Use Application events to initialize objects and data that you want to make available to all the current sessions of your Web application. Use Session events to initialize data that you want to keep throughout individual sessions, but that you don’t want to share between sessions.  
  
**Give an example of how to use Application and Session events?**To see how Application and Session events occur, add the following code to the Global.asax file in a Web forms project.  
void Application\_Start(object sender, EventArgs e)  
{  
// Create Application state variables.  
Application["AppCount"] = 0;  
Application["SessCount"] = 0;  
// Record application start.  
Application["AppCount"] = (int)Application["AppCount"] + 1;  
}  
void Session\_Start(object sender, EventArgs e)  
{  
// Count sessions.  
Application["SessCount"] = (int)Application["SessCount"] + 1;  
}  
void Session\_End(object sender, EventArgs e)  
{  
// Decrement sessions.  
Application["SessCount"] = (int)Application["SessCount"] - 1;  
}  
  
Add the following code to WebForm1.aspx file in a Web forms project and set WebForm1 as start up page.  
protected void Page\_Load(object sender, EventArgs e)  
{  
// Display Application count.  
Response.Write("Number of applications: " +  
Application["AppCount"] + "  
");  
// Display session count.  
Response.Write("Number of sessions: " +  
Application["SessCount"] + "  
");  
}  
To demonstrate the events, run the preceding code, and then start a new instance of the browser and navigate to the address. Each new instance of the browser increments the session count, but the application count stays at 1.  
  
**How long a webform instance is available on the web server?**  
Web forms live for barely a moment. When we request a webform from the browser, the applications executable creates an instance of the requested Web form, generates the HTML to respond to the request, and posts that response to the browser. It then destroys the instance of the Web form.  
  
When the client browser has the generated HTML, the user can type text in boxes, select options, and perform other tasks until triggering a postback event, such as a button click. Postback events cause the browser to send the page’s data (view state) back to the server for event processing. When the server receives the view state, it creates a new instance of the Web form, fills in the data from the view state, and processes any events that occurred. As soon as the server has finished, it posts the resulting HTML back to the browser and destroys the instance of the Web form.  
  
**Give some examples for page level events and when they occur?  
1. Page\_Init**  
During Page\_Init the server controls are loaded and initialized from the Web form’s view state. This is the first step in a Web form’s life cycle.  
  
**2. Page\_Load**  
During Page\_Load the server controls are loaded in the Page object. View state information is available at this point, so this is where you put code to change control settings or display text on the page.  
  
**3. Page\_PreRender**  
During Page\_PreRender the application is about to render the Page object.  
  
**4. Page\_Unload**  
During Page\_Unload the page is unloaded from memory.  
  
**5. Page\_Disposed**  
During Page\_Disposed the Page object is released from memory. This is the last event in the life of a Page object.  
  
**6. Page\_Error**  
Page\_Error event is raised when an unhandled exception occurs.  
  
**7. Page\_AbortTransaction**  
Page\_AbortTransaction is raised when a transaction is aborted.  
  
**8. Page\_CommitTransaction**  
Page\_CommitTransaction is raised when a transaction is commited.  
  
**9. Page\_DataBinding**  
Page\_DataBinding occurs when a server control on the page binds to a data source.  
  
**What is the order in which Page level events occur?**Page\_Init  
Page\_Load  
Page\_PreRender  
Page\_Unload  
Page\_Disposed  
  
**What page property is used to differentiate between a postback and initial get request of a page?**IsPostback property of the Page class.  
  
**What are the 3 types of Server Control Events?**  
Server controls, such as a Button, TextBox, and DropDownList, each have their own sets of events that occur in response to user actions. However, not all server control events are created equal. There are three types of server control events as listed below.  
  
**Postback events :**These events cause the Web page to be sent back to the server for immediate processing. Postback events affect perceived performance because they trigger a round-trip to the server. A Button control triggers a postback event. A dropdowinlist can also trigger a post back if the AutoPostBack property of the DropDownList is set to true, else a dropdownlist triggers a cached event. The same is the case with a TextBox.  
  
**Cached events :**  
These events are saved in the page’s view state to be processed when a postback event occurs. A DropDownList and a TextBox can trigger a cached event if the AutoPostBack property is set to false.  
  
**Validation events :**  
These events are handled on the page without posting back or caching. The validation server controls use these types of events.  
  
**How can you convert a Cached event of control to a post back event?**You can convert a Cached event of control to a post back event by setting the AutoPostBack property to true.  
  
**What is the order in which events are executed?**  
The validations controls are evaluated before the page is posted back to the server. When the page is posted back, the Page\_Init and Page\_Load events are handled, then cached events are handled, and finally the event that caused the postback is processed. Among cached events, the event order is determined by the order of the controls on the Web form.

**Interview Questions on ASP.NET Exception Handling**

**What are Exceptions?**  
Exceptions are unusual occurrences that happen within the logic of an application.  
  
  
**What are the 3 approaches to handle exceptions in a Web application?**  
**1.** Use exception-handling structures to deal with exceptions within the scope of a procedure. This technique is called structured exception handling (SEH) in the Visual Studio .NET documentation.  
                try  
                catch  
                finally   
**2.** Use error events to deal with exceptions within the scope of an object.  
                Page\_Error  
                Global\_Error   
                Application\_Error  
**3.** Use custom error pages to display informational messages for unhandled exceptions within the scope of a Web application.  
  
  
**Where will the control flow if an exception occurs inside a try block?**  
If a statement in a try block causes an exception, control flow passes immediately to the next catch statement. When control flow passes to a catch block, the statements contained in the catch block are processed to correct the error or otherwise handle the exception.  
  
  
**Will the finally block gets executed, if an exception occurs?**  
**Yes**, a finally block **will always be executed** irrespective of whether an exception has occured or not.  
  
  
**What is the main use of a finally block in exception handling?**  
Finally block is mainly used to free resources used within the try block.  
  
  
**How do you raise an exception?**  
Use the throw keyword to raise an exception. Use this keyword within your exception-handling structure to immediately pass control flow to the catch statement.  
   
**Will the following code block compile?**  
try  
{  
     throw new System.IO.FileNotFoundException();  
}  
catch (Exception E)  
{  
     Response.Write(E.Message);  
}  
catch (System.IO.FileNotFoundException FNFE)  
{  
     Response.Write(FNFE.Message);  
}  
  
  
**No**, the following compile time error is reported.  
**A previous catch clause already catches all exceptions of this or of a super type ('System.Exception').**  
  
  
Catch blocks are evaluated in the order in which they appear in code. The exception declaration of each catch block determines which type of exception the catch block handles. Always order catch blocks from most specific to most general. So, in the preceding sample, FileNotFoundException should be placed before the general Exception catch block.  
  
  
**What is ApplicationException class used for?**  
If you are creating a large application or creating components that are used by other applications, you might want to define your own exception classes based on the ApplicationException class.   
  
For example, the following code defines a class for the UserLoggedOnException:  
  
public class UserLoggedOnException : System.ApplicationException  
{  
     // Exception constructor (overloaded).  
     public UserLoggedOnException()  
     : this("The user is already logged on to the server", null)  
     {  
     }  
     public UserLoggedOnException(string message)  
     : this(message, null)  
     {  
     }  
     public UserLoggedOnException(string message, Exception inner)  
     : base(message, inner)  
     {  
     }  
}   
The preceding UserLoggedOnException class inherits its properties and methods from the ApplicationException base class. The new exception class provides only its own constructor to set the default message to display. This is a standard practice.  
  
  
**What are Error Events?**  
Another way to handle exceptions is through the Web objects’ built-in error events. When an unhandled exception occurs in a Web application, ASP.NET fires the error events shown below.  
  
  
**Page\_Error** : Occurs when an unhandled exception occurs on the page. This event procedure resides in the Web form.  
**Global\_Error** : Occurs when an unhandled exception occurs in the application. This event procedure resides in the Global.asax file.  
**Application\_Error** : Occurs when an unhandled exception occurs in the application. This event procedure resides in the Global.asax file.  
  
  
Error events let you handle exceptions for an entire object in a single, centralized location—the error event procedure. This is different from using exception-handling structures, in which exceptions are handled within the procedure where they occurred. You can use error events in the following ways:  
  
  
**As a substitute for exception-handling structures :**  
Because error events occur outside the scope of the procedure in which the error occurred, you have less information about the steps leading up to the exception and therefore less ability to correct the exception condition for the user. However, using exception-handling events is fine for tasks where you might not be able to correct the exception in code.  
**As an adjunct to exception-handling structures :**  
Error events can provide a centralized “backstop” against exceptions that were not foreseen or handled elsewhere. Using the two exception-handling techniques together lets you catch all exceptions before the user sees them, display a reasonable message, and even record the exception in a log as part of an ongoing effort to improve your application.  
  
  
**Give an example to show how error events can be used to handle exceptions?**  
To handle an exception using error events, follow these steps:  
**1.** In the Page\_Error event procedure, get the exception that occurred using the GetLastError method.  
**2.** Do something with the exception, such as display a message to the user, take steps to correct the problem, or write to an error log.  
**3.** Clear the exception using the ClearError method.  
**4.** Redisplay the page. Web form processing stops immediately when an exception occurs, so server controls and other items on the page might not be displayed after the exception is cleared.  
**5.** Add the following code to Page\_Error event procedure on the web page.  
private void Page\_Error(object sender, System.EventArgs e)  
{  
     // Get the error.  
     Exception ex = Server.GetLastError();  
     // Store the message in a session object.  
     Session["Error"] = ex.Message;  
     // Clear the error message.  
     Server.ClearError();  
     // Redisplay this page.  
     Server.Transfer("ErrorEvents.aspx");  
}  
The preceding code stores the exception message as a Session state variable before clearing the exception so that the message can be displayed when the page is reloaded by the Transfer method. The following code displays the saved exception message when the page is redisplayed:  
  
  
Add the following code to Page\_Load event procedure on the web page.  
private void Page\_Load(object sender, System.EventArgs e)  
{  
     // Display error. if any.  
     if (Session["Error"] != null)  
     {  
     litError.Text = "The following error occurred:  
     " +  
     Session["Error"].ToString();  
     // Clear the Session state variable.  
     Session["Error"] = null;  
     }  
}  
  
  
**Can you have a try block without a catch or a finally block?**  
No, you cannot have a try block without a catch or a finally block. A try block cannot exist in isolation. A try block should be followed by either a catch block or a finally block or both.  
  
  
**Is the following code legal?**  
try  
{  
     Response.Write("Try block executed");  
}  
finally  
{  
     Response.Write("Finally block executed");  
}   
  
  
Yes, it's legal. A try statement does not have to have a catch statement if it has a finally statement.  
  
  
**What is wrong with using the following type of exception handler?**  
catch(Exception E)  
{  
     //Some Code  
}  
This handler catches exceptions of type Exception, therefore, it catches any exception. This can be a poor implementation because you are losing valuable information about the type of exception being thrown and making your code less efficient. As a result, your program may be forced to determine the type of exception before it can decide on the best recovery strategy.  
  
  
  
  
**Will the second catch block handle the exception thrown by the first catch block?**  
try  
{  
     throw new System.IO.FileNotFoundException();  
}  
catch (System.IO.FileNotFoundException FNFE)  
{  
     Response.Write(FNFE.Message);  
     throw new Exception();  
}  
catch(Exception E)  
{  
     Response.Write(E.Message);  
}  
  
  
No. For a catch block to handle the exception, the statement that raised the exception must be inside a try block.  
  
  
**What will happen to the exception raised by the code in the following Button1\_Click event procedure?**  
protected void Button1\_Click(object sender, EventArgs e)  
{  
     throw new Exception();  
     try  
     {  
          Response.Write("Hello");  
     }  
     catch (Exception E)  
     {  
          Response.Write(E.Message);  
     }  
}  
  
The exception will not be handled by the catch block because the statement that raised the exception must be inside a try block.

**Managed Code and Unmanaged Code related ASP.NET Interview Questions**

**What is Managed Code and Unmanaged Code?**  
Microsoft ASP.NET Web applications run under the control of the common language runtime (CLR). The CLR controls how the application’s assembly executes, allocates, and recovers memory; therefore, ASP.NET applications are said to use managed code. In contrast, most other Windows executables use unmanaged code because the executable itself determines how memory is used.  
  
Examples of unmanaged code include the Microsoft Win32 API, legacy DLLs and EXEs created for Windows applications prior to the Microsoft .NET Framework, and COM objects.  
  
**What is Platform Invoke or pinvoke?**The process of executing native code from within a .NET assembly is called platform invoke, or pinvoke for short. You use platform invoke to call the Win32 API directly, to access existing (legacy) DLLs your company uses, or to access procedures compiled to native code for performance reasons.  
  
**What are the steps to follow to use Platform Invoke?**  
To use platform invoke, follow the following steps:  
**1.** Import the System.Runtime.InteropServices namespace.  
**2.** Declare the unmanaged procedure using the DllImport attribute or the Declare statement.  
**3.** Map the data types of the procedures parameters to the equivalent .NET types.  
**4.** Call the unmanaged procedure and test its return value for success.  
**5.** If the procedure did not succeed, retrieve and handle the exception code using the Marshal object’s GetLastWin32Error method.  
  
**What are the limitations of using Unmanaged Code from within a .NET assembly?  
Performance :**Although native-code DLLs can perform some operations more quickly than equivalent code managed by the CLR, these benefits might be offset by the time it takes to marshal the data to pass between the unmanaged procedure and the .NET assembly.  
**Type safety :** Unlike .NET assemblies, unmanaged procedures might not be type-safe. This can affect the reliability of your .NET application. In general, reliability is a paramount concern with ASP.NET Web applications.  
**Code security :** Unmanaged procedures do not use the .NET Framework’s model for code security.  
Versioning:Unmanaged code does not support .NET versioning; therefore, assemblies that call unmanaged procedures might lose the benefit of being able to coexist with other versions of the same assembly.  
   
**What are COM objects?**COM objects are another type of unmanaged code that you can use from .NET assemblies. Because COM is widely used, Visual Studio includes built-in tools for importing and using COM objects within .NET assemblies. Visual Studio also includes the option of automatically registering .NET class library assemblies for use from COM.  
  
**List the steps in order, to use a COM object from a .NET assembly in Visual Studio?**  
**1.** Install and register the COM object on your system.  
**2.** Open the .NET project in Visual Studio, and add a reference to the COM object, as shown in diagram below. If the COM object does not appear on the COM tab of the Add Reference dialog box, you can add a reference directly to the executable by clicking Browse.   
[](http://bp1.blogger.com/_1pKHzUD_GjE/SIsxcykNauI/AAAAAAAAAAk/aVpMNBFQkn8/s1600-h/Com.JPG)  
**3.** Create an instance of the COM object in code, and use it as you would any other object.  
  
**What happens when you add a reference to a COM object from with in a dot net application?**When you add a reference to a COM object, Visual Studio automatically generates an interop assembly for the object and places it in the project’s /bin folder. The interop assembly is created from the COM object’s type information and contains the metadata that the CLR uses to call the unmanaged code in the COM object. You can then use COM objects from within .NET code the same way that you use .NET classes.  
  
You can view this interop assembly using the Microsoft Intermediate Language Disassembler (Ildasm.exe) included in the .NET Framework.  
  
**Can we create a .NET object for use from COM?**  
Yes, Visual Studio can automatically generate type library information and register a .NET class library assembly for use from COM. These automatic tools do not work for ASP.NET Web applications, so you must isolate the code you want to use from COM in its own Class Library project.  
  
**How do you hide Public .NET Classes and other public members from COM?**In some cases, you might want to hide selected .NET classes from COM but keep them public for use from other .NET assemblies. The ComVisible attribute allows you to select which public .NET classes and members are included in the generated type library. This attribute applies hierarchically for the assembly, class, and member levels.  
**How do you handle exceptions between .NET and COM?**  
.NET handles errors through exception classes. COM handles errors through 32-bit data types called HRESULTs. All of the .NET exception classes include HResult properties that map to COM HRESULT codes.  
  
If an exception occurs in a .NET object, the exception is automatically mapped to the appropriate HRESULT and returned to COM. Similarly, if an exception occurs in a COM object, the COM HRESULT is mapped to the appropriate exception class, which is returned to .NET, where it can be handled just like any other exception.  
  
If you are creating your own .NET exception classes for use with COM, be sure to set the class’s HResult property so that the exception can be handled within COM.  
  
**What are the technical limitations of COM Interop?**The .NET Framework was developed to address the limitations of COM. Because of this evolution, there are limits to the .NET features that you can use from COM. The following list describes these limits:  
**Static members :**COM requires objects to be created before use, so it does not support .NET Static members.  
**New members :** COM flattens the inheritance tree of .NET objects, so members in a derived class that hides members inherited from a base class are not callable.  
**Constructors with parameters :**COM can’t pass parameters to an object’s constructor.  
  
**What are the practical limitations of using COM objects?**  
The following are the practical limitations of using COM objects from .NET:  
**Shared solutions might not allow COM objects :** ASP.NET host service providers that use nondedicated servers can limit or prohibit the installation of COM objects on their servers.  
**COM objects are prone to memory leaks :**COM uses reference counting to determine when to destroy objects and free memory. It is possible for this reference count to become incorrect, leaving objects in memory indefinitely.  
**Type libraries might be inaccurate :** Because COM separates the object’s description from its implementation, it’s possible for this description to not accurately reflect the object. In this case, the generated interop assembly will also include those inaccuracies.  
**COM is unmanaged code :** All the limitations of unmanaged code apply to COM objects as well.

**Interview Questions on ASP.NET Page navigation techniques**

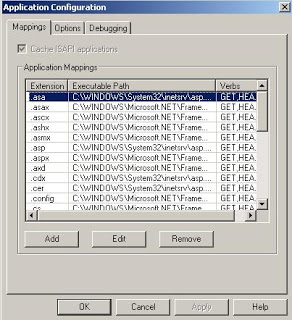
**What are different page navigation techniques in ASP.NET?**  
**Hyperlink control :**Navigate to another page.  
**Response.Redirect :** Navigate to another page from code. This is equivalent to clicking a hyperlink.  
**Server.Transfer :** End the current Web form and begin executing a new Web form. This method works only when navigating to a Web Forms page (.aspx).  
**Server.Execute :** Begin executing a new Web form while still displaying the current Web form. The contents of both forms are combined. This method works only when navigating to a Web Forms page (.aspx).  
**Window.Open script method :**Display a page in a new browser window on the client.  
  
**What is the difference between Response.Redirect and Server.Transfer?  
1.** When we use Server.Transfer the redirection happens on the server where as when we use Response.Redirect the redirection happens from the browser.  
**2.** Server.Transfer is faster as there is no round trip involved while navigating from one webform to another webform.Response.Redirect is slower than Server.Transfer as there is round trip from the server to the client browser.  
**3.** Server.Transfer works only with .aspx files where as Response.Redirect works with .aspx and .Htm pages.  
**4.** Server.Transfer will work with files on the same web server. You can't use Server.Transfer to send the user to an external site where as Response.Redirect can do that.  
**5.** Server.Transfer does not update the URL in the browser. For example when you navigate from WebForm1.aspx to WebForm2.aspx using Server.Transfer the URL in the browser still shows WebForm1.aspx while you are actually looking at WebForm2.aspx. Response.Redirect updates the URL in the browser.  
  
  
**What is the use of Server.Execute method?**Server.Execute method is used to process a second Web form without leaving the first Web form. This technique lets you direct the results from a Web form to a region on the current page.  
  
**Is it possible to send a webform's QueryString, ViewState, and event procedure information to another webform?**  
Yes, we can use Server.Transfer or Server.Execute to send a webform's QueryString, ViewState, and event procedure information to another webform.  
  
For this to work you have to set the preserveForm argument to True.To be able to read one Web form’s ViewState from another, you must first set the EnableViewStateMac attribute in the Web form’s Page directive to False. By default, ASP.NET hashes ViewState information, and setting this attribute to False disables that hashing so that the information can be read on the

**Interview Questions on Query Strings in ASP.NET**

**Give an example of using querystrings to send data from one page to another?**  
Query strings are a very simple and popular technique to pass data from one Web page to the next. You send data as part of the URL. In the below example FName and LName are sent as part of the URL. In the page load of QueryStrings2.aspx we use Request.QueryString to read the values. As we are sending more than one query string we use the & symbol to seperate query strings.

**//Code to send query strings FName and LName as part of the URL**QueryStrings2.aspx?FName=David&LName=Boon   
  
protected void Page\_Load(object sender, EventArgs e)  
{  
//Code to read Query String values  
string FirstName = Request.QueryString["FName"];  
string LastName = Request.QueryString["LName"];  
Response.Write("Data from QueryStrings1.aspx : " + FirstName + ", " + LastName);  
}   
  
**Give an example to send Query Strings from code?**  
You can send query strings from server side code using the Response.Redirect() method as shown below.  
Response.Redirect("QueryStrings2.aspx?FName=David&LName=Boon");  
  
**What are the advantages of using Query Strings?  
1.** Query strings are easy to implement.  
**2.** Browser support for passing values in a query string is nearly universal.  
**3.** Query strings are contained in the HTTP request for a specific URL and do not require server resources.  
  
**What are the disadvantages of using querystrings to send data from one page to another?  
1.** Query strings are insecure because the information in the query string is directly visible to the user on the address line in the browser.  
**2.** Many browsers impose a 255 URL character limit which can limit their flexibility.

**ASP.NET Interview Questions on web application Security**

**What is the difference between Authentication and Authorization?**  
Authentication is the process of identifying users. Authorization is the process of granting access to those users based on identity. Together, authentication and authorization provide the means to keeping your Web application secure from intruders.  
  
**What is Anonymous access?**Anonymous access is the way most public Web sites work. Sites containing public information allow anyone to see that information, so they don’t authenticate users. ASP.NET Web applications provide anonymous access to resources on the server by impersonation. Impersonation is the process of assigning a user account to an unknown user.  
  
**What is the account that is associated with Anonymous access?**  
By default, the anonymous access account is named IUSER\_machinename. You use that account to control anonymous users’ access to resources on the server.  
  
**What is the default user account under which an ASP.NET web application run on a web server?**Under the default settings, ASP.NET uses the ASPNET account to run the Web application. This means that if the application attempts to perform any tasks that are not included in the ASPNET account’s privileges, a security exception will occur and access will be denied.  
  
**How do you restrict the access of anonymous users?**  
You restrict the access of anonymous users by setting Windows file permissions. To be secure, your server must use the Microsoft Windows NT file system (NTFS). The earlier FAT or FAT32 file systems do not provide file-level security.  
**What are the 3 major ways to authenticate and authorize users within an ASP.NET Web application?  
Windows authentication :**Identifies and authorizes users based on the server’s user list. Access to resources on the server is then granted or denied based on the user account’s privileges. This works the same way as regular Windows network security.  
**Forms authentication :**Directs users to a logon Web form that collects user name and password information, and then authenticates the user against a user list or database that the application maintains.  
**Passport authentication :**Directs new users to a site hosted by Microsoft so that they can register a single user name and password that will authorize their access to multiple Web sites. Existing users are prompted for their Microsoft Passport user name and password, which the application then authenticates from the Passport user list.  
  
**What is the namespace where all security related classes are present?**  
System.Web.Security  
  
**What type of authentication can be used for Public Internet Web application?**Anonymous access. This is the common access method for most Web sites. No logon is required, and you secure restricted resources using NTFS file permissions.  
  
**What type of authentication can be used for Intranet Web application?**  
Windows authentication. Windows authentication authenticates network users through the domain controller. Network users have access to Web application resources as determined by their user privileges on the server.  
  
**What type of authentication can be used for Private corporate Web application?**Windows authentication. Corporate users can access the Web application using their corporate network user names and passwords. User accounts are administered using the Windows network security tools.  
  
**What type of authentication can be used for Commercial Web application?**  
Forms authentication. Applications that need to collect shipping and billing information should implement Forms authentication to gather and store customer information.  
  
**What type of authentication can be used for Multiple commercial Web applications?**Passport authentication. Passport authentication allows users to sign in once through a central authority. The user’s identity is then available to any application using the Passport SDK. Customer information is maintained in a Passport profile, rather than in a local database.  
  
**Can you use ASP.NET Authentication with HTM and HTML Files?**  
The three ASP.NET authentication modes apply to files that are part of the Web application. That includes Web forms (.aspx), modules (.asax), and other resources that are processed through the Web application’s executable. It does not automatically include HTML pages (.htm or .html). Those pages are handled by Internet Information Services (IIS), rather than ASP.NET. If you want to authenticate users who access HTML pages from within your Web application using Windows, Forms, or Passport authentication modes, you must map those files to the ASP.NET executable.  
  
**How do map .htm and .html files to the ASP.NET executable using the IIS snap-in?**  
To map .htm and .html files to the ASP.NET executable using the IIS snap-in, follow these steps:  
1. In the IIS snap-in, select the folder containing your Web application, and then choose Properties from the Action menu. IIS displays the Properties dialog box.  
2. Click the Home Directory or Virtual Directory tab, and then click Configuration. IIS displays the Application Configuration dialog box, as shown in the diagram below.  
[](http://bp0.blogger.com/_1pKHzUD_GjE/SIaL0HHsLVI/AAAAAAAAAAM/_QFfKjWzFT8/s1600-h/Security1.JPG)  
3. Click Add. IIS displays the Add/Edit Application Extension Mapping dialog box, as shown in the diagram below.  
[](http://bp1.blogger.com/_1pKHzUD_GjE/SIaMSVPgEwI/AAAAAAAAAAU/ozFsHeH0tUk/s1600-h/Security2.JPG)  
4. Click Browse, and select the aspnet\_isapi.dll file. That file is stored in the Windows Microsoft .NET Framework directory; the path will be something like C:\Windows\Microsoft.NET\Framework\versionnumber\aspnet\_isapi.dll.  
5. Type .htm in the File Extension box, and click OK.  
6. Repeat steps 3 through 5 for the .html file extension. Click OK to close the IIS dialog boxes when you’ve finished.

**ASP.NET Session State and Application State Interview Questions**

**What is a Session?**A Session is a unique instance of the browser. A single user can have multiple instances of the browser running on his or her machine. If each instance visits your Web application, each instance has a unique session.A session starts when a user accesses a page on a Web site for the first time, at which time they are assigned a unique session ID. The server stores the user's session ID in the Session.SessionID property.  
  
**What is the default session timeout period?**  
20 minutes.  
  
**Where do you generally specify the Session Timeout?**You specify the Session Timeout setting in the web.config file.  
  
**Can you specify Session Timeout in a code behind file?**  
Yes, can specify the Session.Timeout property as shown below in a code behind file.  
Session.Timeout = 10;  
  
**How do you end a user session?**You can call the Session.Abandon() method to end a user session. If a user then tries to access a page the server will assign them a new session ID and it will clear all the previous session variables. You'll typically use Session.Abandon() on log-out pages.  
  
**What type of data can you store in Application State and Session State variables?**  
Application State and Session State variables are used to store data that you want to keep for the lifetime of an application or for the lifetime of a session. You can store any type of data in the Application or Session state, including objects.  
  
  
**Are Application State or Session State variables type safe?**No, Application and Session state variables are created on the fly, without variable name or type checking.  
  
**Do maintaining Session state affects performance?**  
Yes  
  
**Can you turn of Session state?**Yes, Session state can be turned off at the application and page levels.  
  
**Are Application state variables available throughout the current process?**  
Yes, Application state variables are available throughout the current process, but not across processes. If an application is scaled to run on multiple servers or on multiple processors within a server, each process has its own Application state.  
  
**How do you disable Session state for a Web form?**To turn Session state off for a Web form set EnableSessionState property of the Page to False.  
  
**How do you turn Session state off for an entire web application?**  
In the Web.config file, set the sessionstate tag to False.  
  
**What are Application State variables?**  
Application State variables are global variables that are available from anywhere in the application. All Sessions can access Application State variables.  
  
**How to add and remove data to Application State Variables?**  
//Code to add data to Application State  
Application.Add("AppName", "Sample");  
  
//Code to remove data from Application State  
Application.Remove("AppName");   
  
**How do you remove all Application State Variables data?**  
//Code to remove all Application State Variables data  
Application.RemoveAll();

**Techniques to send data from one web form to another web form**

**What are the different techniques to send data from one web form to another web form?**  
**1. Query strings :**Use these strings to pass information between requests and responses as part of the Web address. Query strings are visible to the user, so they should not contain secure information such as passwords.  
  
**2. Cookies :**  
Use cookies to store small amounts of information on a client. Clients might refuse cookies, so your code has to anticipate that possibility.  
  
**3. Session state :**  
Use Session state variables to store items that you want keep local to the current session (single user).  
  
**4. Application state :**  
Use Application state variables to store items that you want be available to all users of the application.

**ASP.NET Interview Questions on Tracing**

**What is an exception log?**  
An exception log is a list of handled exceptions that occur while your application is running. Reviewing the exception log periodically helps you verify that exceptions are being handled correctly, are not occurring too frequently, and are not preventing users from accomplishing tasks with your application.  
  
**What is Tracing and what are the adavantages of using tracing to log exceptions?**

Tracing is a technique for recording events, such as exceptions, in an application. There have always been ways to record errors in an application - usually by opening a file and writing error messages to it. But tracing offers the following significant advantages:  
**Standardization:**Building tracing into the .NET Framework ensures that programming techniques are the same across all the applications you develop with the .NET Framework.  
**Built-in Web support:**ASP.NET extends the .NET Framework tools by including information related to the performance and behavior of Web requests.  
**Configuration:**You can turn tracing on and off using settings in your application’s configuration file. You don’t have to recompile your application to enable or disable tracing.  
**Performance:**While disabled, tracing statements do not affect application performance.  
  
**How do you turn tracing on and off for an ASP.NET web application?**  
Tracing can be turned on or off for an entire Web application or for an individual page in the application:  
**1.** To turn tracing on for an entire application, in the application’s Web.config file, set the trace element’s Enabled attribute to True.  
Or  
**2.**To turn tracing on for a single page, set the DOCUMENT object’s Trace property to True in the Visual Studio .NET Properties window. This sets the @ Page directive’s Trace attribute to True in the Web form’s HTML.  
  
**Where is the trace output displayed by default?**By default, trace output is displayed at the end of each Web page.  
  
While this is fine for debugging purposes, you’ll generally want to write trace output to a log file when you start testing your completed application. To write trace messages to a log file for an entire application, in the application’s Web.config file, set the trace element’s PageOutput attribute to False. ASP.NET then writes trace output to the Trace.axd file in your application’s root folder.  
  
**How do you specify, how many page requets should be written to the trace log?**

The <trace> element's RequestLimit attribute can be used to specify how many page requests to write to the trace log. For example, the following line from a Web.config file turns on tracing for the application and writes the first 10 requests to the Trace.axd file:  
  
**How do you write trace messages to a log file for only selected pages in an application?**To write trace messages to a log file for only selected pages in an application, follow these steps:  
In the application’s Web.config file, set the trace element’s Enabled attribute to True and PageOutput attribute to False.  
For each Web page you want to exclude from tracing, set the @ Page directive’s Trace attribute to False.  
  
**What is the difference between Trace.Write() and Trace.Warn() methods of a trace object?**  
The Trace object provides the Write and Warn methods to allow you to write messages to a request’s trace information. The two methods are identical with one difference: messages written with Trace.Write are displayed in black, whereas messages written with Trace.Warn are displayed in red.  
  
**How do you programatically check if tracing is enabled?**The Trace object’s IsEnabled property can be used to programatically check if tracing is enabled.  
  
**How do you prevent from trace output being written at the bottom of the web page?**  
You can prevent from trace output being written at the bottom of the web page by setting the trace element’s PageOutput attribute to False in the Web.config file.  
  
**What is the name of the file to which trace log is written?**Trace.axd  
  
**Can you view Trace.axd from a remote machine?**No, by default, you can view Trace.axd only from the local server running the application. If you want to view the trace log from a remote machine, set the trace element’s LocalOnly attribute to False in the Web.config file

**Transactions related ASP.NET Interview Questions**

**What is a transaction?**  
A transaction is a group of commands that change the data stored in a database. The transaction, which is treated as a single unit, assures that the commands are handled in an all-or-nothing fashion. if one of the commands fails, all of the commands fail, and any data that was written to the database by the commands is backed out. In this way, transactions maintain the integrity of data in a database. ADO.NET lets you group database operations into transactions.  
  
**What is the main purpose of database transactions?**The main purpose of database transactions is to maintain the integrity of data in a database.  
  
**How do you determine which SQL commands are part of a transaction?**

You can determine what database commands belong in a transaction by using the ACID test. Commands must be atomic, consistent, isolated, and durable.   
Commands belong in a transaction if they are:  
**Atomic:**In other words, they make up a single unit of work. For example, if a customer moves, you want your data entry operator to change all of the customer’s address fields as a single unit, rather than changing street, then city, then state, and so on.  
**Consistent:**All the relationships between data in a database are maintained correctly. For example, if customer information uses a tax rate from a state tax table, the state entered for the customer must exist in the state tax table.**Isolated:**Changes made by other clients can’t affect the current changes. For example, if two data entry operators try to make a change to the same customer at the same time, one of two things occurs: either one operator’s changes are accepted and the other is notified that the changes weren’t made, or both operators are notified that their changes were not made. In either case, the customer data is not left in an indeterminate state.  
**Durable:**Once a change is made, it is permanent. If a system error or power failure occurs before a set of commands is complete, those commands are undone and the data is restored to its original state once the system begins running again.  
  
**Why is transaction processing very important for web applications?**Transaction processing is very important for Web applications that use data access, because Web applications are distributed among many different clients. In a Web application, databases are a shared resource, and having many different clients distributed over a wide area can present the below key problems.  
**Contention for resources:**Several clients might try to change the same record at the same time. This problem gets worse the more clients you have.  
**Unexpected failures:**The Internet is not the most reliable network around, even if your Web application and Web server are 100 percent reliable. Clients can be unexpectedly disconnected by their service providers, by their modems, or by power failures.  
**Web application life cycle:**Web applications don’t follow the same life cycle as Windows applications—Web forms live for only an instant, and a client can leave your application at any point by simply typing a new address in his or her browser.  
  
**List the steps in order to process a transaction?**1.Begin a transaction.  
2.Process database commands.  
3.Check for errors.  
4.If errors occurred, restore the database to its state at the beginning of the transaction. If no errors occurred, commit the transaction to the database.  
  
**Explain how a DataSet provides transaction processing?**  
DataSet provide transaction processing through the RejectChanges and Update methods. DataSet also provide an AcceptChanges method that resets the state of records in a data set to Unchanged. Data sets provide implicit transaction processing, because changes to a data set are not made in the database until you invoke the Update method on the data adapter object. This lets you perform a set of commands on the data and then choose a point at which to make the changes permanent in the database.  
  
If an error occurs during the Update method, none of the changes from the data set is made in the database. At that point, you can either attempt to correct the error and try the Update method again or undo the changes pending in the data set using the data set’s RejectChanges method.  
  
**Give an example to show how DataSets provide transaction processing?**  
Let us assume we have a DataGrid that displays employee information. Every row also has a delete button, which when you click will delete that row. On this page we also have a Restore and Commit buttons. When you click the Restore button you should be able to restore the data to its previous state. When you click the Commit button you should be able to update the database with the deletions made in the DataSet.  
  
The code for Commit and Restore buttons is shown below.  
private void butRestore\_Click(object sender, System.EventArgs e)  
{  
// Restore the data set to its original state.  
dsContacts.RejectChanges();  
// Refresh the data grid.  
grdContacts.DataBind();  
}  
  
private void butCommit\_Click(object sender, System.EventArgs e)  
{  
int intRows;  
// Update the database from the data set.  
intRows = adptContacts.Update(dsContacts);  
// Save changes to state variable.  
Session["dsContacts"] = dsContacts;  
// Refresh the data grid.  
grdContacts.DataBind();  
}  
The RejectChanges method in the preceding butRestore\_Click event procedure returns the data set to its state before the row was deleted. The data set’s AcceptChanges method is the inverse of RejectChanges—it resets the DataRowState property for all the changed rows in a data set to Unchanged and removes any deleted rows.  
  
The AcceptChanges method prevents the Update method from making those changes in the database, however, because Update uses the rows’ DataRowState property to determine which rows to modify in the database. For this reason, the AcceptChanges method is useful only when you do not intend to update a database from the data set.  
  
**What are the 3 types of transaction objects available in ADO.NET?**

As we have 3 types of database connections in ADO.NET, there are also 3 types of transaction objects:  
SqlTransaction  
OracleTransaction  
OleDbTransaction  
  
**What are the steps involved in using a transaction object in ADO.NET?**  
1.Open a database connection.  
2.Create the transaction object using the database connection object’s BeginTransaction method.  
3.Create command objects to track with this transaction, assigning the Transaction property of each command object to the name of the transaction object created in step 2.  
4.Execute the commands. Because the purpose of transaction processing is to detect and correct errors before data is written to the database, this is usually done as part of an error-handling structure.  
5.Commit the changes to the database or restore the database state, depending on the success of the commands.  
Close the database connection.  
  
**What property of a transaction object determines how concurrent changes to a database are handled?**IsolationLevel property of the transaction object is used to determine how concurrent changes to a database are handled.  
  
**What are different isolation levels of a transaction object in ADO.NET?  
ReadUncommitted:**Does not lock the records being read. This means that an uncommitted change can be read and then rolled back by another client, resulting in a local copy of a record that is not consistent with what is stored in the database. This is called a dirty read because the data is inconsistent.  
**Chaos:**Behaves the same way as ReadUncommitted, but checks the isolation level of other pending transactions during a write operation so that transactions with more restrictive isolation levels are not overwritten.  
**ReadCommitted:**Locks the records being read and immediately frees the lock as soon as the records have been read. This prevents any changes from being read before they are committed, but it does not prevent records from being added, deleted, or changed by other clients during the transaction. This is the default isolation level.  
**RepeatableRead:**Locks the records being read and keeps the lock until the transaction completes. This ensures that the data being read does not change during the transaction.  
**Serializable:**Locks the entire data set being read and keeps the lock until the transaction completes. This ensures that the data and its order within the database do not change during the transaction.  
  
**What is the default isolation level in a transaction?**  
ReadCommitted  
  
**What is a Save Point in a transaction in ADO.NET?**  
SqlConnection object provide one transaction capability that is unavailable for OLE database connections: the ability to create save points within a transaction. Save points let you restore the database state to a specific position within the current transaction. To set a save point within a SQL transaction, use the Save method as shown below.  
TransactionObject.Save("FirstStep");  
  
**How do you restore a SQL transaction to a specific save point?**  
To restore a SQL transaction to a save point, specify the name of the save point in the Rollback method as shown below.  
TransactionObject.Rollback("FirstStep");

**Interview Questions on ASP.NET Validation controls**

**What are ASP.NET Validation controls?**

ASP.NET provides validation controls to help you check Web form data entries before the data is accepted and saved in the Database. Validation controls can be used to address the following questions.  
1. Did the user enter anything?  
2. Is the entry the appropriate kind of data (For example, Date of Birth should be a valid Date, Name should be a string etc.)?  
3. Is the data within a required range?(For example age cannot be greater than 100 years)  
The validation controls check the validity of data entered in associated server controls on the client before the page is posted back to the server.Most validity problems can be caught and corrected by the user without a round-trip to the server.  
  
**Where do the ASP.NET validation controls validate data, on the Client or on the Web Server?**ASP.NET validation controls validate data first on the client and then on the web server. If a client disables javascript on the browser then, client side validations are bypassed and validations are performed on the web server.  
  
Client-side validation is provided by a JScript library named WebUIValidation.js, which is downloaded separately to the client. Validation controls also automatically provide server-side validation. Server-side validation is always performed, whether or not client-side validation has occurred. This double-checking ensures that custom validations are performed correctly and that client-side validation has not been circumvented.  
  
**What are the 6 different validation controls provided by ASP.NET?**  
**RequiredFieldValidator:**Checks whether a control contains data  
**CompareValidator:**Checks whether an entered item matches an entry in another control  
**RangeValidator:**Checks whether an entered item is between two values  
**RegularExpressionValidator:**Checks whether an entered item matches a specified format  
**CustomValidator:**Checks the validity of an entered item using a client-side script or a server-side code, or both  
**ValidationSummary:**Displays validation errors in a central location or display a general validation error description  
  
**What property of the validation control is used to specify which control to validate?**ControlToValidate property.  
  
**Explain in simple steps how to use validation controls?**  
1.Draw a validation control on a Web form and set its ControlToValidate property to the control you want to validate.   
2.If you’re using the CompareValidator control, you also need to specify the ControlToCompare property.  
3.Set the validation control’s ErrorMessage property to the error message you want displayed if the control’s data is not valid.  
4.Set the validation control’s Text property if you want the validation control to display a message other than the message in the ErrorMessage property when an error occurs. Setting the Text property lets you briefly indicate where the error occurred on the form and display the longer ErrorMessage property in a ValidationSummary control.  
5.Draw a ValidationSummary control on the Web form to display the error messages from the validation controls in one place.  
6.Provide a control that triggers a postback event. Although validation occurs on the client side, validation doesn’t start until a postback is requested.  
  
**Are the validation controls fired on the client side if javascript is disabled on the client browser?**No, validation controls are not fired on the client side if javascript is disabled on the client browser.  
  
**What is the use of CausesValidation property of an ASP.NET button control?**  
CausesValidation property of an ASP.NET button control is used to determine if the validation controls should be fired when the button is clicked. If CausesValidation property is set to true, then validation is performed and if the CausesValidation property is set to false then validation is not done.  
  
**Give an example of real time scenario where you might use CausesValidation property of an ASP.NET button control?**

Let us assume we have a Page that collects user information like name, age, date of birth, gender with a submit and reset buttons. When I click the submit button the information filled on the form should be validated and saved to the database. If I click the reset button then all the controls on the webform should default to their initial values without validation happening.So you have to set the CausesValidation property of the reset button to false for the validation to be bypassed. Other wise you will not be able to post back the page to the server.  
  
**What is ASP.NET Custom Validator used for?**ASP.NET Custom Validator is used to perform complex types of validation not provided by the standard validation control, use a CustomValidator control and write code to perform the validation on the server side and optionally on the client side.  
  
**How do you programatically check, if the client side validation is not bypassed by disabling the javascript on the client browser?**  
We use Page.IsValid property to determine if all the validations have succeeded. For this property to return true, all validation server controls in the current validation group must validate successfully.  
  
**How do you programatically invoke all validation controls on a page?**Call Page.Validate() method. When this method is invoked, it iterates through the validation controls contained in the ValidatorCollection object associated with the Page.Validators property and invokes the validation logic for each validation control in the current validation group.  
  
**What is a validation group?**  
Validation groups allow you to group validation controls on a page as a set. Each validation group can perform validation independently from other validation groups on the page.  
  
**How do you create a validation group?**You create a validation group by setting the ValidationGroup property to the same name for all the controls you want to group. You can assign any name to a validation group, but you must use the same name for all members of the group.  
  
**Explain how a validation group works when the Page is posted by clicking a button?**  
During postback, the Page class's IsValid property is set based only on the validation controls in the current validation group. The current validation group is determined by the control that caused validation to occur. For example, if a button control with a validation group of LoginGroup is clicked, then the IsValid property will return true if all validation controls whose ValidationGroup property is set to LoginGroup are valid.  
  
**Can a DropDownList fire validation controls?**Yes, DropDownList control can also fire validation if the control's CausesValidation property is set to true and the AutoPostBack property is set to true.  
  
**How do you programatically force all validation controls in a particular validation group to be fired?**  
Call the Page.Validate(string GroupName) method and pass the name of the validation group. This will fire only the validation controls in that validation group.  
  
**What is SetFocusOnError property of a validation control used for?**Use the SetFocusOnError property to specify whether focus is automatically set to the control specified by the ControlToValidate property when this validation control fails. This allows the user to quickly update the appropriate control.  
  
If multiple validation controls fail and this property is set to true, the control specified in the ControlToValidate property for the first validation control receives focus.  
  
**What is InitialValue property of a RequiredFieldValidator?**Use this property to specify the initial value of the input control.Validation fails only if the value of the associated input control matches this InitialValue upon losing focus.

**ASP.NET ViewState related Interview Questions**

**What is ViewState?**  
Web forms have very short lifetimes.In ASP.NET, the data that is entered in controls is encoded and stored in a hidden field. This encoded data is then sent with each request and restored to controls in Page\_Init. The data in these controls is then available in the Page\_Load event.The data that ASP.NET preserves between requests is called the Web form’s view state.  
  
**How do you enable or disable a ViewState for a control on the page?**

Every ASP.NET control has a property called EnableViewState. If EnableViewState is set to true ViewState is enabled for the control. If EnableViewState is set to false ViewState is disabled for the control.  
  
**How do you enable or disable a ViewState at the page level?**At the page level you can enable or disable ViewState using EnableViewState property of the page.  
  
**What is the name of the hidden form field in which ViewState of the page is saved?**  
\_\_ViewState  
  
**What are the performance implications of ViewState?**  
ViewState is usually good to retain the state of the controls on the webform across postbacks. If you have a huge DataGrid with tons of data being loaded on every page load. It is a good idea to disable the ViewState of the DataGrid for the page to load faster. If the ViewState of a large DataGrid is not disabled, ViewState can easily get very large, on the order of tens of kilobytes. Not only does the \_\_ViewState form field cause slower downloads, but, whenever the user posts back the Web page, the contents of this hidden form field must be posted back in the HTTP request, thereby lengthening the request time, as well.  
  
**When does ViewState restoration happens?**  
During the Page\_Init event  
  
**What are the disadvantages of using ViewState?**1. On all page visits, during the save view state stage the Page class gathers the collective view state for all of the controls in its control hierarchy and serializes the state to a base-64 encoded string. (This is the string that is emitted in the hidden \_\_ViewState form filed.) Similarly, on postbacks, the load view state stage needs to deserialize the persisted view state data, and update the pertinent controls in the control hierarchy.  
  
2. The \_\_ViewState hidden form field adds extra size to the Web page that the client must download. For some view state-heavy pages, this can be tens of kilobytes of data, which can require several extra seconds (or minutes!) for modem users to download. Also, when posting back, the \_\_ViewState form field must be sent back to the Web server in the HTTP POST headers, thereby increasing the postback request time.  
  
**Is ViewState encoded?**  
Yes, ViewState is base-64 encoded.  
  
**Can you encrypt ViewState of Page?**Yes, we can use the LosFormatter class to encrypt ViewState of Page  
  
**Can the HTML controls retains State accross postbacks?**

No, by default HTML controls donot retain state accross postbacks.  
  
**Can you make HTML controls retain State accross postbacks?**Yes, HTML controls can retain State accross postbacks, if you convert HTML controls to Server Controls. There are 2 ways to convert HTML control to Server Controls.  
  
1. Right click on the HTML Control and then click "Run As Server Control"  
Or  
2. Set runat="server" attribute for the Control.  
  
**Is ViewState supported in classic ASP?**  
No,ViewState is introduced in asp.net, it was not in classic asp.  
  
When a form is submitted in classic ASP, all form values are cleared. Suppose you have submitted a form with a lot of information and the server comes back with an error. You will have to go back to the form and correct the information. You click the back button, and what happens.......ALL form values are CLEARED, and you will have to start all over again! The site did not maintain your ViewState.  
  
When a form is submitted in ASP .NET, the form reappears in the browser window together with all form values. How come? This is because ASP .NET maintains your ViewState. The ViewState indicates the status of the page when submitted to the server.  
  
**Is ViewState of one page available to another page?**No, ViewState of a Page is available only in that page. You cannot access ViewState of one page from another page.  
  
**Can you programatically store and retrieve data from ViewState?**  
Yes. In ASP.NET you can programatically store and retrieve data from ViewState.See the example below  
  
//Save the value in ViewState object  
ViewState("SomeVar") = txtFirstName.text;   
  
//Retrieve the value from ViewState object  
String strFirstName = ViewState("SomeVar").ToString();  
  
**Can someone view the Page HTML source and read ViewState?**  
No. ViewState is base-64 encoded. Hence you cannot read ViewState. If you right click on the Page and View Source you will find \_\_ViewState is base-64 encoded.  
  
**What happens during the Page\_Init event?**The server controls are loaded and initialized from the Web form’s view state. This is the first step in a Web form’s life cycle.

**Web application maintainance related ASP.NET Interview Questions**

**Why is it important to monitor a deployed a Web application?**  
After you have deployed a Web application, you need to monitor how it performs on the server. Many issues can crop up at this point because of  
1. The number of users accessing the application.  
2. The unpredictable nature of user interaction.  
3. The possibility of malicious attack.  
  
**What are the 3 major steps invloved in maintaining a deployed web application?**  
Maintaining a deployed application is an ongoing task that involves three major steps:  
1. Monitoring the application for error, performance, and security issues.  
2. Repairing the application as issues are discovered.  
3. Tuning the application to respond to user traffic.  
  
**What are the MMC snap-ins provided by windows for monitoring security, performance, and error events?**  
The Event Viewer snap-in : Lists application, system, and security events as they occur on the system. Use this tool to see what is currently happening on the server and to get specific information about a particular event.  
The Performance snap-in : Lets you create new events to display in the Event Viewer and allows you to track event counters over time for display graphically or in report form.  
  
**How do you run the Event Viewer to view application, system, and security events as they happen?**  
To run the Event Viewer, choose Event Viewer from the Administrative Tools submenu on the Windows Start menu. (You can show the Administrative Tools menu in Windows XP Professional by opening the Start button’s Properties dialog box and clicking the Customize button on the Start Menu tab. The Administrative Tools option is located on the Advanced tab of the Customize Start Menu dialog box.) After clicking the Event Viewer shortcut, Windows displays the Event Viewer snap-in in the MMC.  
  
**What are the 3 levels at which the Event Viewer snap-in displays general application, system, and security events?**  
1. Informational events.  
2. Warning events.  
3. Error events.  
  
**Can you repair a deployed web application in place without restarting the server or IIS?**  
Yes, to repair a deployed Web application, copy the new assembly (.dll) and/or content files (.aspx, .ascx, and so on) to the application folder on the server. ASP.NET automatically restarts the application when you replace the assembly. You do not need to install or register the assembly on the server.  
  
**What is ASP.NET Process recycling?**  
ASP.NET Web applications have a limited ability to repair themselves through process recycling. Process recycling is the technique of shutting down and restarting an ASP.NET worker process (aspnet\_wp.exe) that has become inactive or is consuming excessive resources. You can control how ASP.NET processes are recycled through attributes in the processModel element in the Machine.config file.  
  
**Describes the processModel attributes found in machine.config that relate to process recycling?**  
1. **timeout attribute**:The amount of time (hh:mm:ss) before the process is shut down and restarted. Use this setting to automatically recycle a process after a certain number of requests as a preventive measure.  
2. **shutDownTimeOut attribute** : How much time each process has to shut itself down. After this amount of time, the process is terminated by the system if it still running.  
3. **requestLimit attribute** : The number of queued requests to serve before the process is shut down and restarted. Use this setting the same way you use the timeout attribute.  
4. **restartQueueLimit attribute** : The number of queued requests to retain while the process is shut down and restarted.  
5. **memoryLimit attribute** : The percentage of physical memory the ASP.NET process is allowed to consume before that process is shut down and a new process is started. This setting helps prevent memory leaks from degrading the server.  
6.**responseRestart­ - DeadlockInterval** : The amount of time to wait before restarting a process that was shut down because it was deadlocked. This setting is usually several minutes to prevent applications with serious errors from degrading the server.  
7.**responseDeadlockInterval**: The amount of time to wait before restarting a process that is deadlocked. The process is restarted if there are queued requests and the process has not responded within this time limit.  
  
**What is the use of processModel element apart from providing process recycling?**  
The processModel element in the server’s Machine.config file provides attributes that control certain performance aspects, such as  
1. The maximum number of requests to be queued.  
2. How long to wait before checking whether a client is connected.  
3. How many threads to allow per processor.  
  
**Describe the processModel attributes that relate to performance tuning?**  
**requestQueueLimit**  
The number of queued requests allowed before ASP.NET returns response code 503 (Server too busy) to new requests  
**clientConnectedCheck**  
The amount of time (hh:mm:ss) to wait before checking whether a client is still connected  
**maxWorkerThreads**  
The maximum number of threads per processor  
**maxIOThreads**  
The maximum number of I/O threads per processor  
  
In general, lowering these settings allows your server to handle fewer clients more quickly. Increasing these settings permits more clients and more queued requests, but slows response times.  
  
**How do you turn off Session state?**  
To turn off Session state, in the application’s Web.config file, set the sessionState element’s mode attribute to Off.  
  
**If you turn of Session State, can you use Session state variables in code anywhere in the application?**  
No  
  
**What is Optimization?**  
Optimization usually refers to writing code in a way that executes more quickly or consumes fewer resources. In general, optimizations simply reflect the good programming practices.  
  
**List some of the common steps to follow to optimize a web application for performance?**  
**Turn off debugging for deployed applications.**  
Code that has been compiled with release options runs faster than code compiled with debug options.  
**Avoid round-trips between the client and server.**  
ASP.NET uses postbacks to process server events on a page. Try to design Web forms so that the data on the Web form is complete before the user posts the data to the server. You can use the validation controls to ensure that data is complete on the client side before the page is submitted.  
**Turn off Session state if it isn’t needed.**  
In some cases, you can design your code to use other techniques, such as cookies, to store client data.  
**Turn off ViewState for server controls that do not need to retain their values.**  
Saving ViewState information adds to the amount of data that must be transmitted back to the server with each request.  
**Use stored procedures with databases.**  
Stored procedures execute more quickly than ad hoc queries.  
**Use SqlDataReader rather than data sets for read-forward data retrieval.**  
Using SqlDataReader is faster and consumes less memory than creating a data set.

**ASP.NET Interview Questions on web farm and web garden**

**What does the term Scalability mean?**  
Web applications that serve a large number of users or that present large amounts of data need to able to add capacity as users’ demands increase. The ability to add capacity to an application is called scalability. ASP.NET Web applications support this concept through their ability to run in multiple processes and to have those processes distributed across multiple CPUs and/or multiple servers.  
  
**What is the difference between a web farm and a web garden?**  
A Web application running on a single server that has multiple CPUs is called a **Web garden** in the ASP.NET documentation. A Web application running on multiple servers is called a **Web farm**.  
  
**If your web server has multiple processors, how can you specify that ASP.NET runs on all or some of the CPUs?**  
If your server has multiple processors, you can specify that ASP.NET runs on all or some of the CPUs by setting the **webGarden**attribute of the processModel element in the server’s Machine.config file  
  
**What are the implications on Application and Session state variables in a web farm or a web garden?**  
In both a **Web garden**and a **Web farm**, client requests are directed to the ASP.NET process that is currently least busy. That means that a single client can interact with different CPUs or servers over the course of his or her session. This has the following implications for Application and Session state variables:  
  
**Application state variables are unique to each separate instance of the Web application.**  
Clients can share information through Application state if the Web application is running on a Web garden or a Web farm.  
  
**Session state variables are stored in-process by default.**  
To enable Session state in a Web garden or Web farm, you need to specify a Session state provider.  
  
**How can you share Application State in a web farm or a web garden?**  
To share data across multiple sessions in a **Web garden**or **Web farm**, you must save and restore the information using a resource that is available to all the processes. This can be done through an XML file, a database, or some other resource using the standard file or database access methods.  
  
**What are the two built-in ways provided by ASP.NET to share Session state information across a Web garden or Web farm?**  
ASP.NET provides two built-in ways to share Session state information across a Web garden or Web farm. You can share Session state using:  
  
**A state server, as specified by a network location**  
This technique is simple to implement and doesn’t require you to install Microsoft SQL Server.  
**A SQL database, as specified by a SQL connection**  
This technique provides the best performance for storing and retrieving state information.  
  
**What are the steps to follow to share Session state information using a state server?**  
To share Session state information using a state server, follow these steps:  
**1.** In the Web application’s Web.config file, set the sessionState element’s mode and stateConnectionString attributes.  
**2.** Run the aspnet\_state.exe utility on the Session state server. The aspnet\_state.exe utility is installed in the \WINDOWS\Microsoft.NET \Framework\version folder when you install Visual Studio .NET Professional or Visual Studio .NET Enterprise Architect editions.  
  
**What are the steps to follow to share Session state information using a SQL database?**  
To share Session state information using a SQL database, follow these steps:  
**1.** In the Web application’s Web.config file, set the sessionState element’s mode and sqlConnectionString attributes.  
**2.** Run the InstallSqlState.sql utility on the Session state server. This utility installs the SQL database that shares Session state information across processes. The InstallSqlState.sql utility is installed in the \WINDOWS\Microsoft.NET \Framework\version folder when you install Visual Studio .NET Professional, Visual Studio .NET Enterprise Developer, or Visual Studio .NET Enterprise Architect editions.

**ASP.NET Interview Questions on web user controls**

**What are ASP.NET Custom controls?**  
Custom controls extend the tools available to Web developers. Using custom controls, you can encapsulate key aspects of the visual interface and program logic that you want to reuse throughout your application, or throughout your organization.  
  
**What are the 3 types of custom controls in ASP.NET?**  
Microsoft Visual Studio .NET provides three types of custom control for use on Web forms.  
**1. Web user controls**  
These combine existing server and HTML controls by using the Visual Studio .NET Designer to create functional units that encapsulate some aspect of the user interface. User controls reside in content files, which must be included in the project in which the controls are used.  
**2. Composite custom controls**  
These create new controls from existing server and HTML controls. Although similar to user controls, composite controls are created in code rather than visually, and therefore they can be compiled into an assembly (.dll), which can be shared between multiple applications and used from the Toolbox in Visual Studio .NET.  
**3. Rendered custom controls**  
These create entirely new controls by rendering HTML directly rather than using composition. These controls are compiled and can be used from the Toolbox, just like composite controls, but you must write extra code to handle tasks that are performed automatically in composite controls.  
  
**What are the limitations of user controls in ASP.NET?**  
As the user controls are not compiled into assemblies, they have the following limitations:  
**1.** A copy of the control must exist in each Web application project in which the control is used.  
**2.** User controls can’t be loaded in the Visual Studio .NET Toolbox; instead, you must create them by dragging the control from Solution Explorer to the Web form.  
**3.** User control code is initialized after the Web form loads, which means that user control property values are not updated until after the Web form’s Load event.  
  
**What are the steps to follow for creating and using a user control in a Web application?**  
**1.** Add a Web user control page (.ascx) to your project.  
**2.** Draw the visual interface of the control in the designer.  
**3.** Write code to create the control’s properties, methods, and events.  
**4.** Use the control on a Web form by dragging it from Solution Explorer to the Web form on which you want to include it.  
**5.** Use the control from a Web form’s code by declaring the control at the module level and then using the control’s methods, properties, and events as needed within the Web form.  
  
**How do you identify user controls?**  
User controls are identified by their .ascx file extensions.  
  
**What is the base class from which user controls derive?**  
User controls derive from System.Web.UI.UserControl base class. This base class provides the base set of properties and methods you use to create the control.  
  
**What are the steps to follow to create properties and methods for the user control that you can use from a Web form?**  
To create properties and methods for the user control that you can use from a Web form, follow these steps:  
**1.** Create the public property or method that you want to make available on the containing Web form.  
**2.** Write code to respond to events that occur for the controls contained within the user control. These event procedures do the bulk of the work for the user control.  
**3.** If the property or method needs to retain a setting between page displays, write code to save and restore settings from the control’s ViewState.  
  
**What happens when you drag a user control from solution explorer and drop it on a web form?**  
When you drag a user control from solution explorer and drop it on a web form, Visual Studio .NET generates a @Register directive and HTML tags to create the control on the Web form.

**Interview Questions on ASP.NET Custom Controls**

**What are composite custom controls?**  
Composite custom controls combine one or more server or HTML controls within a single control class, which can be compiled along with other control classes to create an assembly (.dll) that contains a custom control library. Once created, the custom control library can be loaded into Visual Studio .NET and used in the same way as the standard server and HTML controls.  
  
Composite custom controls are functionally similar to user controls, but they reside in their own assemblies, so you can share the same control among multiple projects without having to copy the control to each project, as you must do with user controls. However, composite controls are somewhat more difficult to create because you can’t draw them visually using the Visual Studio .NET Designer.  
  
**What are the steps to follow create and use a custom control in a Web application?**  
**1.** Create a solution containing a custom control project.  
**2.** Add a Web application project to the solution, and set it as the startup project. You will use the Web application project to test the custom control during development.  
**3.** Add a project reference from the Web application to the custom control project, and add an HTML @Register directive and control element to use the custom control on a Web form.  
**4.** Create the custom control’s visual interface by adding existing controls to it through the custom control’s CreateChildControls method.  
**5.** Add the properties, methods, and events that the custom control provides.  
**6.** Build and test the custom control.  
  
  
**In general what is the base class for every composite custom control?**  
System.Web.UI.WebControls.WebControl  
  
**Which directive is used to add a custom control to a Web form?**  
Register directive.  
  
**What are the 3 Register directive's attributes?**  
**TagPrefix**  
This name identifies the group that the user control belongs to. For example, the tag prefix for ASP.NET server controls is “asp”. You use this prefix to create a naming convention to organize your custom controls.  
**Namespace**  
This is the project name and namespace within the custom control assembly that contains the controls to register. Microsoft Visual Basic .NET uses the project name as an implicit namespace, so for controls written in Visual Basic .NET, use the project name.  
**Assembly**  
This is the name of the assembly (.dll) containing the custom controls. The control assembly must be referenced by the Web application. Referencing the assembly maintains a copy of it in the Web application’s /bin directory.  
  
**What are the differences between User Controls and Custom Controls?**  
**1.** User Controls are easy to create where as Custom Controls are difficult to create.  
**2.** User Controls cannot be compiled into an assembly, where as Custom Controls can be compiled into an assembly.  
**3.** User Controls cannot be added to tool box, where as Custom controls can be added to the toolbox.  
**4.** You need to have a copy of user control in every project where you want to use it, where as this is not the case with custom controls. You can install a single copy of the Web custom control in the global assembly cache and share it between applications, which makes maintenance easier.  
**5.** User controls are used for reusing existing user interface elements and code, but are not useful for developing reusable components for multiple web applications.

**Interview Questions on cascading style sheets**

**What are the 2 ways provided by ASP.NET to format output in a Web application?**  
**1.** Use **cascading style sheets**(CSS) to control the appearance of elements on a Web form.These styles can set the color, size, font, and behavior of the HTML elements on a Web page.  
  
**2.** Use **Extensible Stylesheet Language Transformations** (XSLT) to convert information from an Extensible Markup Language (XML) file to HTML output and position that information on a Web form. XSLT puts data from the XML file into HTML elements and applies styles to those elements.  
  
**What are Cascading style sheets?**  
**Cascading style sheets**(CSS) collect and organize all of the formatting information applied to HTML elements on a Web form. Because they keep this information in a single location, style sheets make it easy to adjust the appearance of Web applications.  
  
**What are the 3 levels at which formatting can be applied with in a web application?  
1.** Styles can be defined in a style sheet file. Styles in a style sheet can be applied to all webforms referencing the style sheet.  
  
**2.** You can also define styles in the page’s head element. These styles can be applied to all elements on the current page.  
  
**3.** You can also define styles inline, in the HTML tag itself. Inline styles are applicable only to the HTML element in which these styles are defined.  
  
**Inline formatting**takes precedence over local formatting, which, in turn, takes precedence over global formatting. These precedence rules are the reason style sheets are referred to as **cascading**.  
  
**What are the advantages of storing style definitions in a style sheet file (.css) rather than locally in each Web form or inline with each HTML element?**  
**1.** Formatting can be maintained in one location so that you make changes only once for an entire application.  
  
**2.** Several sets of parallel formatting rules can be maintained in separate style sheets for formatting output on different devices or for different user needs. For example, an application might provide standard, enlarged-type, and printer-friendly style sheets that the user can select at run time.  
  
**3.** In general, you should use page and inline styles only when you have a really good reason to override the global styles. Relying heavily on page and inline styles can make it difficult to maintain the formatting in a Web application.  
  
**What HTML element is used to reference a style sheet on webform?**To reference a style sheet on webform you must add a link element to the page’s head element, as shown below.  
  
**<link href="Styles.css" type="text/css" rel="stylesheet">**  
**What is the use of Style Builder?  
Style Builder** is used to change the appearance of any of the styles in a style sheet. Changes to the style sheet change the appearance of all Web forms that reference that style sheet.  
  
**How do you modify a style sheet using style builder?**To modify a style sheet using style builder, follow these steps:  
  
**1.** Open the style sheet in Visual Studio. Visual Studio .NET displays the style definitions in the Document window and an outline of the style sheet in the Tool window  
  
**2.** Select the style to modify from the Tool window. Visual Studio .NET displays the definition for that style in the Document window.  
  
**3.** Right-click in the style definition or right-click the style in the Tool window, and select Build Style from the shortcut menu. Visual Studio .NET displays the Style Builder Wizard.  
  
**4.** Use the Style Builder to compose the formatting that you want to add or modify in the selected style, and then click OK.  
  
**5.** When you have finished, you’ll see that the Style Builder adds the new or modified style attributes to the style definition.  
  
**Can you apply styles using class names or element IDs?**Yes, Using **class names**allows you to apply a single style to a number of different elements or to style the same element differently, depending on how the element is used. Using **element IDs**allows you to apply a style to a unique element on one or more Web forms.  
  
When you create a style rule for a class, Visual Studio .NET adds a style definition to the style sheet using a **.classname**identifier.  
  
You apply the style class to HTML elements by using the **class** attribute. You apply the style to server controls by using the**CssClass**attribute.  
  
**Can you change style sheets at run time?**Yes.

.Net Versions and Features: see the [pdf] file in the directory of the current document.