



This Download is from www.downloadmela.com . The main motto of this website is to provide free download links of ebooks,video tutorials,magazines,previous papers,interview related content. To download more visit the website.

If you like our services please help us in 2 ways.

1.Donate money.

Please go through the link to donate

<http://www.downloadmela.com/donate.html>

2.Tell about this website to your friends,relatives.

Thanks for downloading. Enjoy the reading.

What is .NET?

.NET is essentially a framework for software development. It is similar in nature to any other software development framework (J2EE etc) in that it provides a set of runtime containers/capabilities, and a rich set of pre-built functionality in the form of class libraries and APIs

The .NET Framework is an environment for building, deploying, and running Web Services and other applications. It consists of three main parts: the Common Language Runtime, the Framework classes, and ASP.NET.

How many languages .NET is supporting now?

When .NET was introduced it came with several languages. VB.NET, C#, COBOL and Perl, etc. The site DotNetLanguages.Net says 44 languages are supported.

How is .NET able to support multiple languages?

A language should comply with the Common Language Runtime standard to become a .NET language. In .NET, code is compiled to Microsoft Intermediate Language (MSIL for short). This is called as Managed Code. This Managed code is run in .NET environment. So after compilation to this IL the language is not a barrier. A code can call or use a function written in another language.

How ASP .NET different from ASP?

Scripting is separated from the HTML, Code is compiled as a DLL, these DLLs can be executed on the server.

What is smart navigation?

The cursor position is maintained when the page gets refreshed due to the server side validation and the page gets refreshed.

What is view state?

The web is stateless. But in ASP.NET, the state of a page is maintained in the in the page itself automatically. How? The values are encrypted and saved in hidden controls. this is done automatically by the ASP.NET. This can be switched off / on for a single control

Visit <http://www.downloadmela.com/> for more papers

How do you validate the controls in an ASP .NET page?

Using special validation controls that are meant for this. We have Range Validator, Email Validator.

Can the validation be done in the server side? Or this can be done only in the Client side?

Client side is done by default. Server side validation is also possible. We can switch off the client side and server side can be done.

How to manage pagination in a page?

Using pagination option in DataGrid control. We have to set the number of records for a page, then it takes care of pagination by itself.

What is ADO .NET and what is difference between ADO and ADO.NET?

ADO.NET is stateless mechanism. I can treat the ADO.Net as a separate in-memory database where in I can use relationships between the tables and select insert and updates to the database. I can update the actual database as a batch.

Observations between VB.NET and VC#.NET?

Choosing a programming language depends on your language experience and the scope of the application you are building. While small applications are often created using only one language, it is not uncommon to develop large applications using multiple languages.

For example, if you are extending an application with existing XML Web services, you might use a scripting language with little or no programming effort. For client-server applications, you would probably choose the single language you are most comfortable with for the entire application. For new enterprise applications, where large teams of developers create components and services for deployment across multiple remote sites, the best choice might be to use several languages depending on developer skills and long-term maintenance expectations.

The .NET Platform programming languages - including Visual Basic .NET, Visual C#, and Visual C++ with managed extensions, and many other programming languages from various vendors - use .NET Framework services and features through a common set of unified classes. The .NET unified classes provide a consistent method of accessing the platform's functionality. If you learn to use the class library, you will find that all tasks follow the same uniform architecture. You no longer need to learn and master different API architectures to write your applications.

In most situations, you can effectively use all of the Microsoft programming languages. Nevertheless, each programming language has its relative strengths and you will want to understand the features unique to each language. The following sections will help you choose the right programming language for your application.

Visual Basic .NET

Visual Basic .NET is the next generation of the Visual Basic language from Microsoft. With Visual Basic you can build .NET applications, including Web services and ASP.NET Web applications, quickly and easily. Applications made with Visual Basic are built on the services of the common language runtime and take advantage of the .NET Framework.

Visual Basic has many new and improved features such as inheritance, interfaces, and overloading that make it a powerful object-oriented programming language. Other new language features include free

threading and structured exception handling. Visual Basic fully integrates the .NET Framework and the common language runtime, which together provide language interoperability, garbage collection, enhanced security, and improved versioning support. A Visual Basic support single inheritance and creates Microsoft intermediate language (MSIL) as input to native code compilers.

Visual Basic is comparatively easy to learn and use, and Visual Basic has become the programming language of choice for hundreds of thousands of developers over the past decade. An understanding of Visual Basic can be leveraged in a variety of ways, such as writing macros in Visual Studio and providing programmability in applications such as Microsoft Excel, Access, and Word.

Visual Basic provides prototypes of some common project types, including:

- Windows Application.
- Class Library.
- Windows Control Library.
- ASP.NET Web Application.
- ASP.NET Web Service.
- Web Control Library.
- Console Application.
- Windows Service.
- Windows Service.

Visual C# .NET

Visual C# (pronounced C sharp) is designed to be a fast and easy way to create .NET applications, including Web services and ASP.NET Web applications. Applications written in Visual C# are built on the services of the common language runtime and take full advantage of the .NET Framework.

C# is a simple, elegant, type-safe, object-oriented language recently developed by Microsoft for building a wide range of applications. Anyone familiar with C and similar languages will find few problems in adapting to C#. C# is designed to bring rapid development to the C++ programmer without sacrificing the power and control that are a hallmark of C and C++. Because of this heritage, C# has a high degree of fidelity with C and C++, and developers familiar with these languages can quickly become productive in C#. C# provides intrinsic code trust mechanisms for a high level of security, garbage collection, and type safety. C# supports single inheritance and creates Microsoft intermediate language (MSIL) as input to native code compilers.

C# is fully integrated with the .NET Framework and the common language runtime, which together provide language interoperability, garbage collection, enhanced security, and improved versioning support. C# simplifies and modernizes some of the more complex aspects of C and C++, notably namespaces, classes, enumerations, overloading, and structured exception handling. C# also eliminates C and C++ features such as macros, multiple inheritance, and virtual base classes. For current C++ developers, C# provides a powerful, high-productivity language alternative.

Visual C# provides prototypes of some common project types, including:

- Windows Application.
- Class Library.

Visit <http://www.downloadmela.com/> for more papers

- Windows Control Library.
- ASP.NET Web Application.
- ASP.NET Web Service.
- Web Control Library.
- Console Application.
- Windows Service.

Advantages of migrating to VB.NET ?

Visual Basic .NET has many new and improved language features — such as inheritance, interfaces, and overloading that make it a powerful object-oriented programming language. As a Visual Basic developer, you can now create multithreaded, scalable applications using explicit multithreading. Other new language features in Visual Basic .NET include structured exception handling, custom attributes, and common language specification (CLS) compliance.

The CLS is a set of rules that standardizes such things as data types and how objects are exposed and interoperate. Visual Basic .NET adds several features that take advantage of the CLS. Any CLS-compliant language can use the classes, objects, and components you create in Visual Basic .NET. And you, as a Visual Basic user, can access classes, components, and objects from other CLS-compliant programming languages without worrying about language-specific differences such as data types.

CLS features used by Visual Basic .NET programs include assemblies, namespaces, and attributes.

These are the new features to be stated briefly:

Inheritance

Visual Basic .NET supports inheritance by allowing you to define classes that serve as the basis for derived classes. Derived classes inherit and can extend the properties and methods of the base class. They can also override inherited methods with new implementations. All classes created with Visual Basic .NET are inheritable by default. Because the forms you design are really classes, you can use inheritance to define new forms based on existing ones.

Exception Handling

Visual Basic .NET supports structured exception handling, using an enhanced version of the Try...Catch...Finally syntax supported by other languages such as C++.

Structured exception handling combines a modern control structure (similar to Select Case or While) with exceptions, protected blocks of code, and filters. Structured exception handling makes it easy to create and maintain programs with robust, comprehensive error handlers.

Overloading

Overloading is the ability to define properties, methods, or procedures that have the same name but use different data types. Overloaded procedures allow you to provide as many implementations as necessary to handle different kinds of data, while giving the appearance of a single, versatile procedure. **Overriding Properties and Methods** The Overrides keyword allows derived objects to override characteristics inherited from parent objects. Overridden members have the same arguments as the members inherited from the base class, but different implementations. A member's new implementation can call the original implementation in the parent class by preceding the member name with MyBase.

Constructors and Destructors

Constructors are procedures that control initialization of new instances of a class. Conversely, destructors are methods that free system resources when a class leaves scope or is set to Nothing. Visual Basic .NET supports constructors and destructors using the Sub New and Sub Finalize procedures.

Data Types

Visual Basic .NET introduces three new data types. The Char data type is an unsigned 16-bit quantity used to store Unicode characters. It is equivalent to the .NET Framework System.Char data type. The Short data type, a signed 16-bit integer, was named Integer in earlier versions of Visual Basic. The Decimal data type is a 96-bit signed integer scaled by a variable power of 10. In earlier versions of Visual Basic, it was available only within a Variant.

Interfaces

Interfaces describe the properties and methods of classes, but unlike classes, do not provide implementations. The Interface statement allows you to declare interfaces, while the Implements statement lets you write code that puts the items described in the interface into practice.

Delegates

Delegates objects that can call the methods of objects on your behalf are sometimes described as type-safe, object-oriented function pointers. You can use delegates to let procedures specify an event handler method that runs when an event occurs. You can also use delegates with multithreaded applications. For details, see Delegates and the AddressOf Operator.

Shared Members

Shared members are properties, procedures, and fields that are shared by all instances of a class. Shared data members are useful when multiple objects need to use information that is common to all. Shared class methods can be used without first creating an object from a class.

References

References allow you to use objects defined in other assemblies. In Visual Basic .NET, references point to assemblies instead of type libraries. For details, see References and the Imports Statement. Namespaces prevent naming conflicts by organizing classes, interfaces, and methods into hierarchies.

Assemblies

Assemblies replace and extend the capabilities of type libraries by, describing all the required files for a particular component or application. An assembly can contain one or more namespaces.

Attributes

Attributes enable you to provide additional information about program elements. For example, you can use an attribute to specify which methods in a class should be exposed when the class is used as a XML Web service.

Multithreading

Visual Basic .NET allows you to write applications that can perform multiple tasks independently. A task that has the potential of holding up other tasks can execute on a separate thread, a process known as multithreading. By causing complicated tasks to run on threads that are separate from your user

interface, multithreading makes your applications more responsive to user input.

Using ActiveX Control in .Net

ActiveX control is a special type of COM component that supports a User Interface. Using ActiveX Control in your .Net Project is even easier than using COM component. They are bundled usually in .ocx files. Again a proxy assembly is made by .Net utility AxImp.exe (which we will see shortly) which your application (or client) uses as if it is a .Net control or assembly.

Making Proxy Assembly For ActiveX Control: First, a proxy assembly is made using AxImp.exe (acronym for ActiveX Import) by writing following command on Command Prompt:

```
C:> AxImp C:MyProjectsMyControl.ocx
```

This command will make two dlls, e.g., in case of above command

MyControl.dll

AxMyControl.dll

The first file MyControl.dll is a .Net assembly proxy, which allows you to reference the ActiveX as if it were non-graphical object.

The second file AxMyControl.dll is the Windows Control, which allows u to use the graphical aspects of activex control and use it in the Windows Form Project.

Adding Reference of ActiveX Proxy Assembly in your Project Settings: To add a reference of ActiveX Proxy Assembly in our Project, do this:

- o Select Project A Add Reference (Select Add Reference from Project Menu).
 - o This will show you a dialog box, select .Net tab from the top of window.
 - o Click Browse button on the top right of window.
 - o Select the dll file for your ActiveX Proxy Assembly (which is MyControl.dll) and click OK
 - o Your selected component is now shown in the 'Selected Component' List Box. Click OK again
- Some More On Using COM or ActiveX in .Net

.Net only provides wrapper class or proxy assembly (Runtime Callable Wrapper or RCW) for COM or activeX control. In the background, it is actually delegating the tasks to the original COM, so it does not convert your COM/activex but just imports them.

A good thing about .Net is that when it imports a component, it also imports the components that are publically referenced by that component. So, if your component, say MyDataAcess.dll references ADODB.dll then .Net will automatically import that COM component too!

The Visual Studio.NET does surprise you in a great deal when u see that it is applying its intellisense (showing methods, classes, interfaces, properties when placing dot) even on your imported COM components!!!! Isn't it a magic or what?

When accessing thru RCW, .Net client has no knowledge that it is using COM component, it is presented just as another C# assembly.

U can also import COM component thru command prompt (for reference see Professional C# by Wrox)

U can also use your .Net components in COM, i.e., export your .net components (for reference see Professional C# by Wrox)

What is Machine.config?

Machine configuration file: The machine.config file contains settings that apply to the entire computer. This file is located in the %runtime install path%\Config directory. There is only one machine.config file on a computer. The Machine.Config file found in the "CONFIG" subfolder of your .NET Framework install directory (c:\WINNT\Microsoft.NETFramework\{Version Number}\CONFIG on Windows 2000 installations). The machine.config, which can be found in the directory \$WINDIR\Microsoft.NETFramework\v1.0.3705\CONFIG, is an XML-formatted configuration file that specifies configuration options for the machine. This file contains, among many other XML elements, a browserCaps element. Inside this element are a number of other elements that specify parse rules for the various User-Agents, and what properties each of these parsing supports.

For example, to determine what platform is used, a filter element is used that specifies how to set the platform property based on what platform name is found in the User-Agent string. Specifically, the machine.config file contains:

```
platform=Win95  
platform=Win98  
platform=WinNT  
...
```

That is, if in the User-Agent string the string "Windows 95" or "Win95" is found, the platform property is set to Win95. There are a number of filter elements in the browserCaps element in the machine.config file that define the various properties for various User-Agent strings.

Hence, when using the Request.Browser property to determine a user's browser features, the user's agent string is matched up to particular properties in the machine.config file. The ability for being able to detect a user's browser's capabilities, then, is based upon the honesty in the browser's sent User-Agent string. For example, Opera can be easily configured to send a User-Agent string that makes it appear as if it's IE 5.5. In this case from the Web server's perspective (and, hence, from your ASP.NET Web page's perspective), the user is visiting using IE 5.5, even though, in actuality, he is using Opera.

What is Web.config?

In classic ASP all Web site related information was stored in the metadata of IIS. This had the disadvantage that remote Web developers couldn't easily make Web-site configuration changes. For example, if you want to add a custom 404 error page, a setting needs to be made through the IIS admin tool, and you're Web host will likely charge you a flat fee to do this for you. With ASP.NET, however, these settings are moved into an XML-formatted text file (Web.config) that resides in the Web site's root directory. Through Web.config you can specify settings like custom 404 error pages, authentication and authorization settings for the Web site compilation options for the ASP.NET Web

pages, if tracing should be enabled, etc.

The Web.config file is an XML-formatted file. At the root level is the tag. Inside this tag you can add a number of other tags, the most common and useful one being the system.web tag, where you will specify most of the Web site configuration parameters. However, to specify application-wide settings you use the tag.

For example, if we wanted to add a database connection string parameter we could have a Web.config file like so.

What is the difference between ADO and ADO.NET?

ADO uses Recordsets and cursors to access and modify data. Because of its inherent design, Recordset can impact performance on the server side by tying up valuable resources. In addition, COM marshalling - an expensive data conversion process - is needed to transmit a Recordset. ADO.NET addresses three important needs that ADO doesn't address:

1. Providing a comprehensive disconnected data-access model, which is crucial to the Web environment
2. Providing tight integration with XML, and
3. Providing seamless integration with the .NET Framework (e.g., compatibility with the base class library's type system). From an ADO.NET implementation perspective, the Recordset object in ADO is eliminated in the .NET architecture. In its place, ADO.NET has several dedicated objects led by the DataSet object and including the DataAdapter, and DataReader objects to perform specific tasks. In addition, ADO.NET DataSets operate in disconnected state whereas the ADO RecordSet objects operated in a fully connected state.

In ADO, the in-memory representation of data is the RecordSet. In ADO.NET, it is the dataset. A RecordSet looks like a single table. If a RecordSet is to contain data from multiple database tables, it must use a JOIN query, which assembles the data from the various database tables into a single result table. In contrast, a dataset is a collection of one or more tables. The tables within a dataset are called data tables; specifically, they are DataTable objects. If a dataset contains data from multiple database tables, it will typically contain multiple DataTable objects. That is, each DataTable object typically corresponds to a single database table or view. In this way, a dataset can mimic the structure of the underlying database.

In ADO you scan sequentially through the rows of the RecordSet using the ADO MoveNext method. In ADO.NET, rows are represented as collections, so you can loop through a table as you would through any collection, or access particular rows via ordinal or primary key index. A cursor is a database element that controls record navigation, the ability to update data, and the visibility of changes made to the database by other users. ADO.NET does not have an inherent cursor object, but instead includes data classes that provide the functionality of a traditional cursor. For example, the functionality of a forward-only, read-only cursor is available in the ADO.NET DataReader object.

There is one significant difference between disconnected processing in ADO and ADO.NET. In ADO you communicate with the database by making calls to an OLE DB provider. In ADO.NET you communicate with the database through a data adapter (an OleDbDataAdapter, SqlDataAdapter, OdbcDataAdapter, or OracleDataAdapter object), which makes calls to an OLE DB provider or the APIs provided by the underlying data source.

What is the difference between VB and VB.NET?

Now VB.NET is object-oriented language. The following are some of the differences:

Data Type Changes

The .NET platform provides Common Type System to all the supported languages. This means that all the languages must support the same data types as enforced by common language runtime. This eliminates data type incompatibilities between various languages. For example on the 32-bit Windows platform, the integer data type takes 4 bytes in languages like C++ whereas in VB it takes 2 bytes. Following are the main changes related to data types in VB.NET:

- . Under .NET the integer data type in VB.NET is also 4 bytes in size.
- . VB.NET has no currency data type. Instead it provides decimal as a replacement.
- . VB.NET introduces a new data type called Char. The char data type takes 2 bytes and can store Unicode characters.
- . VB.NET do not have Variant data type. To achieve a result similar to variant type you can use Object data type. (Since every thing in .NET including primitive data types is an object, a variable of object type can point to any data type).
- . In VB.NET there is no concept of fixed length strings.
- . In VB6 we used the Type keyword to declare our user-defined structures. VB.NET introduces the structure keyword for the same purpose.

Declaring Variables

Consider this simple example in VB6:

```
Dim x,y as integer
```

In this example VB6 will consider x as variant and y as integer, which is somewhat odd behavior. VB.NET corrects this problem, creating both x and y as integers.

Furthermore, VB.NET allows you to assign initial values to the variables in the declaration statement itself:

```
br> Dim str1 as string = Hello
```

VB.NET also introduces Read-Only variables. Unlike constants Read-Only variables can be declared without initialization but once you assign a value to it, it cannot be changes.

Initialization here

```
Dim readonly x as integer
```

In later code

```
X=100
```

Now x can't be changed

```
X=200 ***** Error *****
```

Property Syntax

In VB.NET, we anymore don't have separate declarations for Get and Set/Let. Now, everything is done in a single property declaration. This can be better explained by the following example.

```
Public [ReadOnly | WriteOnly] Property PropertyName as Datatype
```

```

Get
Return m_var
End Get
Set
M_var = value
End Set
End Property
Example:
Private _message as String
Public Property Message As String
Get
Return _message
End Get
Set
_message = Value
End Set
End Property

```

ByVal is the default - This is a crucial difference between VB 6.0 and VB.NET, where the default in VB 6.0 was by reference. But objects are still passed by reference.

Invoking Subroutines In previous versions of VB, only functions required the use of parentheses around the parameter list. But in VB.NET all function or subroutine calls require parentheses around the parameter list. This also applies, even though the parameter list is empty.

User-Defined Types - VB.NET does away with the keyword Type and replaces it with the keyword Structure

```

Public Structure Student
Dim strName as String
Dim strAge as Short
End Structure

```

Procedures and Functions

In VB6 all the procedure parameters are passed by reference (ByRef) by default. In VB.NET they are passed by value (ByVal) by default. Parentheses are required for calling procedures and functions whether they accept any parameters or not. In VB6 functions returned values using syntax like: FunctionName = return_value. In VB.NET you can use the Return keyword (Return return_value) to return values or you can continue to use the older syntax, which is still valid.

Scoping VB.NET now supports block-level scoping of variables. If your programs declare all of the variables at the beginning of the function or subroutine, this will not be a problem. However, the following VB 6.0 will cause an issue while upgrading to VB .NET

```

Do While objRs.EOF
Dim J as Integer
J=0

```

```

If objRs("flag")="Y" then
J=1
End If
objRs.MoveNext
Wend
If J Then
Msgbox "Flag is Y"
End If

```

In the above example the variable J will become out of scope just after the loop, since J was declared inside the While loop.

Exception Handling

The most wanted feature in earlier versions of VB was its error handling mechanism. The older versions relied on error handlers such as "On Error GoTo and On Error Resume Next. VB.NET provides us with a more structured approach. The new block structure allows us to track the exact error at the right time. The new error handling mechanism is referred to as Try...Throw...Catch...Finally. The following example will explain this new feature.

```

Sub myOpenFile()
Try
Open "myFile" For Output As #1
Write #1, myOutput
Catch
Kill "myFile"
Finally
Close #1
End try
End Sub

```

The keyword SET is gone - Since everything in VB.NET is an object. So the keyword SET is not at all used to differentiate between a simple variable assignment and an object assignment. So, if you have the following statement in VB 6.0

```

Set ObjConn = Nothing
Should be replaced as
ObjConn = Nothing.
Constructor and Destructor

```

The constructor procedure is one of the many new object-oriented features of VB.NET. The constructor in VB.NET replaces the Class_Initialize in VB 6.0. All occurrence of Class_Initialize in previous versions of VB should now be placed in a class constructor. In VB.NET, a constructor is added to a class by adding a procedure called New. We can also create a class destructor, which is equivalent to Class_Terminate event in VB 6.0, by adding a sub-procedure called Finalize to our class. Usage of Return In VB.NET, we can use the keyword return to return a value from any function. In previous

versions, we used to assign the value back with the help of the function name itself. The following example explains this:

```
Public Function Sum (intNum1 as Integer, intNum2 as Integer) as Integer
Dim intSum as Integer
intSum = intNum1 + intNum2
Return intSum
End Function
Static Methods
```

VB.NET now allows you to create static methods in your classes. Static methods are methods that can be called without requiring the developer to create instance of the class. For example, if you had a class named Foo with the non-static method NonStatic() and the static method Static(), you could call the Static() method like so:

```
Foo.Static()
```

However, non-static methods require than an instance of the class be created, like so:

```
Create an instance of the Foo class
Dim objFoo as New Foo()
Execute the NonStatic() method
ObjFoo.NonStatic()
```

To create a static method in a VB.NET, simply prefix the method definition with the keyword Shared.

What is a Strong Name?

A strong name consists of the assembly's identity its simple text name, version number, and culture information (if provided) plus a public key and a digital signature. It is generated from an assembly file (the file that contains the assembly manifest, which in turn contains the names and hashes of all the files that make up the assembly), using the corresponding private key. Assemblies with the same strong name are expected to be identical.

Strong names guarantee name uniqueness by relying on unique key pairs. No one can generate the same assembly name that you can, because an assembly generated with one private key has a different name than an assembly generated with another private key.

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.

There are two ways to sign an assembly with a strong name:

1. Using the Assembly Linker (Al.exe) provided by the .NET Framework SDK.
2. Using assembly attributes to insert the strong name information in your code. You can use either the

AssemblyKeyFileAttribute or the AssemblyKeyNameAttribute, depending on where the key file to be used is located.

To create and sign an assembly with a strong name using the Assembly Linker, at the command prompt, type the following command:
al /out: /keyfile:

In this command, assembly name is the name of the assembly to sign with a strong name, module name is the name of the code module used to create the assembly, and file name is the name of the container or file that contains the key pair.

The following example signs the assembly MyAssembly.dll with a strong name using the key file sgKey.snk.

```
al /out:MyAssembly.dll MyModule.netmodule /keyfile:sgKey.snk
```

To sign an assembly with a strong name using attributes

In a code module, add the AssemblyKeyFileAttribute or the AssemblyKeyNameAttribute, specifying the name of the file or container that contains the key pair to use when signing the assembly with a strong name. The following code example uses the AssemblyKeyFileAttribute with a key file called sgKey.snk.

[Visual Basic]

[C#]

```
[assembly:AssemblyKeyFileAttribute(@"....sgKey.snk")]
```

What is a Manifest?

An assembly manifest contains all the metadata needed to specify the assembly's version requirements and security identity, and all metadata needed to define the scope of the assembly and resolve references to resources and classes. The assembly manifest can be stored in either a PE (Portable Executable) file (an .exe or .dll) with Microsoft intermediate language (MSIL) code or in a standalone PE (Portable Executable) file that contains only assembly manifest information. The following table shows the information contained in the assembly manifest. The first four items the assembly name, version number, culture, and strong name information make up the assembly's identity.

Assembly name: A text string specifying the assembly's name.

Version number: A major and minor version number, and a revision and build number. The common language runtime uses these numbers to enforce version policy.

Culture: Information on the culture or language the assembly supports. This information should be used only to designate an assembly as a satellite assembly containing culture- or language-specific information. (An assembly with culture information is automatically assumed to be a satellite assembly.)
Strong name information: The public key from the publisher if the assembly has been given a strong name.
List of all files in the assembly:

A hash of each file contained in the assembly and a file name. Note that all files that make up the assembly must be in the same directory as the file containing the assembly manifest.

Type reference information: Information used by the runtime to map a type reference to the file that contains its declaration and implementation. This is used for types that are exported from the assembly.

Information on referenced assemblies: A list of other assemblies that are statically referenced by the assembly. Each reference includes the dependent assembly's name, assembly metadata (version, culture, operating system, and so on), and public key, if the assembly is strong named.

Creating a Key Pair?

You can create a key pair using the Strong Name tool (Sn.exe). Key pair files usually have an .snk extension. To create a key pair At the command prompt, type the following command:

```
sn k
```

In this command, file name is the name of the output file containing the key pair. The following example creates a key pair called sgKey.snk.

```
sn -k sgKey.snk
```

What is the difference between "using System.Data;" and directly adding the reference from "Add References Dialog Box"?

When u compile a program using command line, u add the references using /r switch. When you compile a program using Visual Studio, it adds those references to our assembly, which are added using "Add Reference" dialog box. While "using" statement facilitates us to use classes without using their fully qualified names.

For example: if u have added a reference to "System.Data.SqlClient" using "Add Reference" dialog box then u can use SqlConnection class like this:

```
System.Data.SqlClient.SqlConnection
```

But if u add a "using System.Data.SqlClient" statement at the start of ur code then u can directly use SqlConnection class.

On the other hand if u add a reference using "using System.Data.SqlClient" statement, but don't add it using "Add Reference" dialog box, Visual Studio will give error message while we compile the program.

What is GAC?

The global assembly cache stores assemblies specifically designated to be shared by several applications on the computer. You should share assemblies by installing them into the global assembly cache only when you need to. Assemblies deployed in the global assembly cache must have a strong name. When an assembly is added to the global assembly cache, integrity checks are performed on all files that make up the assembly. The cache performs these integrity checks to ensure that an assembly has not been tampered with, for example, when a file has changed but the manifest does not reflect the

change. Use a developer tool called the Global Assembly Cache tool (Gacutil.exe), provided by the .NET Framework SDK or Use Windows Explorer to drag assemblies into the cache. To install a strong-named assembly into the global assembly cache At the command prompt, type the following command:

```
gacutil I
```

In this command, assembly name is the name of the assembly to install in the global assembly cache.

What is a Metadata?

Metadata is information about a PE. In COM, metadata is communicated through non-standardized type libraries.

In .NET, this data is contained in the header portion of a COFF-compliant PE and follows certain guidelines;

it contains information such as the assembly's name, version, language (spoken, not computera.k.a., culture), what external types are referenced, what internal types are exposed, methods, properties, classes, and much more.

The CLR uses metadata for a number of specific purposes. Security is managed through a public key in the PE's header.

Information about classes, modules, and so forth allows the CLR to know in advance what structures are necessary. The class loader component of the CLR uses metadata to locate specific classes within assemblies, either locally or across networks.

Just-in-time (JIT) compilers use the metadata to turn IL into executable code.

Other programs take advantage of metadata as well.

A common example is placing a Microsoft Word document on a Windows 2000 desktop. If the document file has completed comments, author, title, or other Properties metadata, the text is displayed as a tool tip when a user hovers the mouse over the document on the desktop. You can use the Ildasm.exe utility to view the metadata in a PE. Literally, this tool is an IL disassembler.

What is managed code and managed data?

Managed code is code that is written to target the services of the Common Language Runtime.

In order to target these services, the code must provide a minimum level of information (metadata) to the runtime.

All C#, Visual Basic .NET, and JScript .NET code is managed by default.

Visual Studio .NET C++ code is not managed by default, but the compiler can produce managed code by specifying a command-line switch (/CLR).

Closely related to managed code is managed data--data that is allocated and de-allocated by the Common Language Runtime's garbage collector. C#, Visual Basic, and JScript .NET data is managed by default.

C# data can, however, be marked as unmanaged through the use of special keywords.

Visual Studio .NET C++ data is unmanaged by default (even when using the /CLR switch), but when

using Managed Extensions for C++, a class can be marked as managed using the `__gc` keyword. As the name suggests, this means that the memory for instances of the class is managed by the garbage collector.

In addition, the class becomes a full participating member of the .NET Framework community, with the benefits and restrictions that it brings. An example of a benefit is proper interoperability with classes written in other languages (for example, a managed C++ class can inherit from a Visual Basic class).

An example of a restriction is that a managed class can only inherit from one base class.

What is .NET / .NET Framework?

It is a Framework in which Windows applications may be developed and run. The Microsoft .NET Framework is a platform for building, deploying, and running Web Services and applications. It provides a highly productive, standards-based, multi-language environment for integrating existing investments with next-generation applications and services as well as the agility to solve the challenges of deployment and operation of Internet-scale applications. The .NET Framework consists of three main parts: the common language runtime, a hierarchical set of unified class libraries, and a componentized version of Active Server Pages called ASP.NET. The .NET Framework provides a new programming model and rich set of classes designed to simplify application development for Windows, the Web, and mobile devices. It provides full support for XML Web services, contains robust security features, and delivers new levels of programming power. The .NET Framework is used by all Microsoft languages including Visual C#, Visual J#, and Visual C++.

What is Reflection?

It extends the benefits of metadata by allowing developers to inspect and use it at runtime. For example, dynamically determine all the classes contained in a given assembly and invoke their methods. Reflection provides objects that encapsulate assemblies, modules, and types. You can use reflection to dynamically create an instance of a type, bind the type to an existing object, or get the type from an existing object. You can then invoke the type's methods or access its fields and properties.

Namespace: System.Reflection

What is "Common Type System" (CTS)?

CTS defines all of the basic types that can be used in the .NET Framework and the operations performed on those type.

All this time we have been talking about language interoperability, and .NET Class Framework. None of this is possible without all the language sharing the same data types. What this means is that an int should mean the same in VB, VC++, C# and all other .NET compliant languages. This is achieved through introduction of Common Type System (CTS).

What is "Common Language Specification" (CLS)?

CLS is the collection of the rules and constraints that every language (that seeks to achieve .NET compatibility) must follow. It is a subsection of CTS and it specifies how it shares and extends one another libraries.

What is "Common Language Runtime" (CLR)?

CLR is .NET equivalent of Java Virtual Machine (JVM). It is the runtime that converts a MSIL code into the host machine language code, which is then executed appropriately. The CLR is the execution engine for .NET Framework applications. It provides a number of services, including:

- Code management (loading and execution)

- Application memory isolation
- Verification of type safety
- Conversion of IL to native code.
- Access to metadata (enhanced type information)
- Managing memory for managed objects
- Enforcement of code access security
- Exception handling, including cross-language exceptions
- Interoperation between managed code, COM objects, and pre-existing DLL's (unmanaged code and data)
- Automation of object layout
- Support for developer services (profiling, debugging, and so on).

What are Attributes?

Attributes are declarative tags in code that insert additional metadata into an assembly. There exist two types of attributes in the .NET Framework: Predefined attributes such as `AssemblyVersion`, which already exist and are accessed through the Runtime Classes; and custom attributes, which you write yourself by extending the `System.Attribute` class.

What are the Types of Assemblies?

Assemblies are of two types:

1. Private Assemblies
2. Shared Assemblies

Private Assemblies: The assembly is intended only for one application. The files of that assembly must be placed in the same folder as the application or in a sub folder. No other application will be able to make a call to this assembly. The advantage of having a private assembly is that, it makes naming the assembly very easy, since the developer need not worry about name clashes with other assemblies. As long as the assembly has a unique name within the concerned application, there won't be any problems.

Shared Assemblies: If the assembly is to be made into a Shared Assembly, then the naming conventions are very strict since it has to be unique across the entire system. The naming conventions should also take care of newer versions of the component being shipped. These are accomplished by giving the assembly a Shared Name. Then the assembly is placed in the global assembly cache, which is a folder in the file system reserved for shared assemblies.

What is an Intermediate language?

Assemblies are made up of IL code modules and the metadata that describes them. Although programs may be compiled via an IDE or the command line, in fact, they are simply translated into IL, not machine code. The actual machine code is not generated until the function that requires it is called.

This is the just-in-time, or JIT, compilation feature of .NET. JIT compilation happens at runtime for a variety of reasons, one of the most ambitious being Microsoft's desire for cross-platform .NET adoption. If a CLR is built for another operating system (UNIX or Mac), the same assemblies will run in addition to the Microsoft platforms. The hope is that .NET assemblies are write-once-run-anywhere applications. This is a .NET feature that works behind-the-scenes, ensuring that developers are not limited to writing applications for one single line of products. No one has demonstrated whether or not this promise will ever truly materialize.

CTS/CLS

The MSIL Instruction Set Specification is included with the .NET SDK, along with the IL Assembly

Language Programmers Reference. If a developer wants to write custom .NET programming languages, these are the necessary specifications and syntax. The CTS and CLS define the types and syntaxes that every .NET language needs to embrace. An application may not expose these features, but it must consider them when communicating through IL.

ASP.NET Authentication Providers and IIS Security

ASP.NET implements authentication using authentication providers, which are code modules that verify credentials and implement other security functionality such as cookie generation. ASP.NET supports the following three authentication providers:

Forms Authentication: Using this provider causes unauthenticated requests to be redirected to a specified HTML form using client side redirection. The user can then supply logon credentials, and post the form back to the server. If the application authenticates the request (using application-specific logic), ASP.NET issues a cookie that contains the credentials or a key for reacquiring the client identity. Subsequent requests are issued with the cookie in the request headers, which means that subsequent authentications are unnecessary.

Passport Authentication: This is a centralized authentication service provided by Microsoft that offers a single logon facility and membership services for participating sites. ASP.NET, in conjunction with the Microsoft® Passport software development kit (SDK), provides similar functionality as Forms Authentication to Passport users.

Windows Authentication: This provider utilizes the authentication capabilities of IIS. After IIS completes its authentication, ASP.NET uses the authenticated identity's token to authorize access.

To enable a specified authentication provider for an ASP.NET application, you must create an entry in the application's configuration file as follows:

```
// web.config file
```

What is the difference between ASP and ASP.NET?

ASP is interpreted. ASP.NET Compiled event base programming.

Control events for text button can be handled at client javascript only. Since we have server controls events can handle at server side.

More error handling.

ASP .NET has better language support, a large set of new controls and XML based components, and better user authentication.

ASP .NET provides increased performance by running compiled code.

ASP .NET code is not fully backward compatible with ASP.

ASP .NET also contains a new set of object oriented input controls, like programmable list boxes, validation controls. A new data grid control supports sorting, data paging, and everything you expect from a dataset control. The first request for an ASP.NET page on the server will compile the ASP .NET code and keep a cached copy in memory. The result of this is greatly increased performance.

ASP .NET is not fully compatible with earlier versions of ASP, so most of the old ASP code will need some changes to run under ASP .NET. To overcome this problem,

ASP .NET uses a new file extension ".aspx". This will make ASP .NET applications able to run side by side with standard ASP applications on the same server.

Using COM Component in .Net ?

As most of you know that .Net does not encourage the development of COM components and provides a different solution to making reusable components through Assemblies. But, there are a lot of COM components present which our .Net application might need to use. Fortunately, .Net provides an extremely simple approach to achieve this. This is achieved by using 'Wrapper Classes' and 'Proxy Components'. .Net wraps the COM component into .Net assembly technically called 'Runtime Callable Wrapper' or RCW. Then u can call and use your COM component just as a .Net (or C#, if u are using C#) Assembly.

What is an assembly?

An assembly is the primary building block of a .NET Framework application. It is a collection of functionality that is built, versioned, and deployed as a single implementation unit (as one or more files). All managed types and resources are marked either as accessible only within their implementation unit, or as accessible by code outside that unit. .NET Assembly contains all the metadata about the modules, types, and other elements it contains in the form of a manifest. The CLR loves assemblies because differing programming languages are just perfect for creating certain kinds of applications. For example, COBOL stands for Common Business-Oriented Language because it's tailor-made for creating business apps. However, it's not much good for creating drafting programs. Regardless of what language you used to create your modules, they can all work together within one Portable Executable Assembly. There's a hierarchy to the structure of .NET code. That hierarchy is Assembly -> Module -> Type -> Method." Assemblies can be static or dynamic. Static assemblies can include .NET Framework types (interfaces and classes), as well as resources for the assembly (bitmaps, JPEG files, resource files, and so on). Static assemblies are stored on disk in portable executable (PE) files. You can also use the .NET Framework to create dynamic assemblies, which are run directly from memory and are not saved to disk before execution. You can save dynamic assemblies to disk after they have executed.

What is a Web Service?

A web service is a software component that exposes itself through the open communication channels of the Internet. Applications running on remote machines, on potentially different platforms, can access these components in a language and platform-independent manner. A Web Service is a group of functions, packaged together for use in a common framework throughout a network.

webFarm Vs webGardens

A web farm is a multi-server scenario. So we may have a server in each state of US. If the load on one server is in excess then the other servers step in to bear the brunt.

How they bear it is based on various models.

1. RoundRobin. (All servers share load equally)
2. NLB (economical)
3. HLB (expensive but can scale up to 8192 servers)
4. Hybrid (of 2 and 3).

5. CLB (Component load balancer).

A web garden is a multi-processor setup. i.e., a single server (not like the multi server above).

How to implement webfarms in .Net:

Go to web.config and Here for mode = you have 4 options.

a) Say mode=inproc (non web farm but fast when you have very few customers).

b) Say mode=StateServer (for webfarm)

c) Say mode=SqlServer (for webfarm)

Whether to use option b or c depends on situation. StateServer is faster but SqlServer is more reliable and used for mission critical applications.

How to use webgardens in .Net:

Go to web.config and Change the false to true. You have one more attribute that is related to webgarden in the same tag called cpuMask.

What is the difference between a namespace and assembly name?

A namespace is a logical naming scheme for types in which a simple type name, such as MyType, is preceded with a dot-separated hierarchical name. Such a naming scheme is completely under control of the developer. For example, types MyCompany.FileAccess.A and MyCompany.FileAccess.B might be logically expected to have functionally related to file access. The .NET Framework uses a hierarchical naming scheme for grouping types into logical categories of related functionality, such as the ASP.NET application framework, or remoting functionality. Design tools can make use of namespaces to make it easier for developers to browse and reference types in their code. The concept of a namespace is not related to that of an assembly. A single assembly may contain types whose hierarchical names have different namespace roots, and a logical namespace root may span multiple assemblies. In the .NET Framework, a namespace is a logical design-time naming convenience, whereas an assembly establishes the name scope for types at run time.

What's a Windows process?

It's an application that's running and had been allocated memory.

What's typical about a Windows process in regards to memory allocation?

Each process is allocated its own block of available RAM space, no process can access another process' code or data. If the process crashes, it dies alone without taking the entire OS or a bunch of other applications down.

Explain what relationship is between a Process, Application Domain, and Application?

Each process is allocated its own block of available RAM space, no process can access another process' code or data. If the process crashes, it dies alone without taking the entire OS or a bunch of other applications down.

A process is an instance of a running application. An application is an executable on the hard drive or network. There can be numerous processes launched of the same application (5 copies of Word running), but 1 process can run just 1 application.

What are possible implementations of distributed applications in .NET?

.NET Remoting and ASP.NET Web Services. If we talk about the Framework Class Library, noteworthy classes are in System.Runtime.Remoting and System.Web.Services.

What are the consideration in deciding to use .NET Remoting or ASP.NET Web Services?

Remoting is a more efficient communication exchange when you can control both ends of the application involved in the communication process. Web Services provide an open-protocol-based exchange of information. Web Services are best when you need to communicate with an external

organization or another (non-.NET) technology.

What's a proxy of the server object in .NET Remoting?

It's a fake copy of the server object that resides on the client side and behaves as if it was the server. It handles the communication between real server object and the client object. This process is also known as marshaling.

What are remotable objects in .NET Remoting?

Remotable objects are the objects that can be marshaled across the application domains. You can marshal by value, where a deep copy of the object is created and then passed to the receiver. You can also marshal by reference, where just a reference to an existing object is passed.

What are channels in .NET Remoting?

Channels represent the objects that transfer the other serialized objects from one application domain to another and from one computer to another, as well as one process to another on the same box. A channel must exist before an object can be transferred.

What security measures exist for .NET Remoting in System.Runtime.Remoting?

None. Security should be taken care of at the application level. Cryptography and other security techniques can be applied at application or server level.

What is a formatter?

A formatter is an object that is responsible for encoding and serializing data into messages on one end, and deserializing and decoding messages into data on the other end.

Choosing between HTTP and TCP for protocols and Binary and SOAP for formatters, what are the trade-offs?

Binary over TCP is the most efficient, SOAP over HTTP is the most interoperable.

What's SingleCall activation mode used for?

If the server object is instantiated for responding to just one single request, the request should be made in SingleCall mode.

What's Singleton activation mode?

A single object is instantiated regardless of the number of clients accessing it. Lifetime of this object is determined by lifetime lease.

How do you define the lease of the object?

By implementing ILease interface when writing the class code.

Can you configure a .NET Remoting object via XML file?

Yes, via machine.config and application level .config file (or web.config in ASP.NET). Application-level XML settings take precedence over machine.config.

How can you automatically generate interface for the remotable object in .NET with Microsoft tools?

Use the Soapsuds tool.

What is Delegation?

A delegate acts like a strongly type function pointer. Delegates can invoke the methods that they reference without making explicit calls to those methods.

Delegate is an entity that is entrusted with the task of representation, assign or passing on information. In code sense, it means a Delegate is entrusted with a Method to report information back to it when a

certain task (which the Method expects) is accomplished outside the Method's class.

What is "Microsoft Intermediate Language" (MSIL)?

A .NET programming language (C#, VB.NET, J# etc.) does not compile into executable code; instead it compiles into an intermediate code called Microsoft Intermediate Language (MSIL). As a programmer one need not worry about the syntax of MSIL - since our source code is automatically converted to MSIL. The MSIL code is then send to the CLR (Common Language Runtime) that converts the code to machine language, which is, then run on the host machine. MSIL is similar to Java Byte code. MSIL is the CPU-independent instruction set into which .NET Framework programs are compiled. It contains instructions for loading, storing, initializing, and calling methods on objects. Combined with metadata and the common type system, MSIL allows for true cross- language integration Prior to execution, MSIL is converted to machine code. It is not interpreted.

Differences between Datagrid, Datalist and Repeater?

1. Datagrid has paging while Datalist doesnt.
 2. Datalist has a property called repeat. Direction = vertical/horizontal. (This is of great help in designing layouts). This is not there in Datagrid.
 3. A repeater is used when more intimate control over html generation is required.
 4. When only checkboxes/radiobuttons are repeatedly served then a checkboxlist or radiobuttonlist are used as they involve fewer overheads than a Datagrid.
- The Repeater repeats a chunk of HTML you write, it has the least functionality of the three. DataList is the next step up from a Repeater; accept you have very little control over the HTML that the control renders. DataList is the first of the three controls that allow you Repeat-Columns horizontally or vertically. Finally, the DataGrid is the motherload. However, instead of working on a row-by-row basis, you're working on a column-by-column basis. DataGrid caters to sorting and has basic paging for your disposal. Again you have little contro, over the HTML. NOTE: DataList and DataGrid both render as HTML tables by default. Out of the 3 controls, I use the Repeater the most due to its flexibility w/ HTML. Creating a Pagination scheme isn't that hard, so I rarely if ever use a DataGrid. Occasionally I like using a DataList because it allows me to easily list out my records in rows of three for instance.

I am constantly writing the drawing procedures with System.Drawing.Graphics, but having to use the try and dispose blocks is too time-consuming with Graphics objects. Can I automate this?

Yes, the code

```
System.Drawing.Graphics canvas = new System.Drawing.Graphics();
try
{
//some code
}
finally
canvas.Dispose();
```

is functionally equivalent to

```
using (System.Drawing.Graphics canvas = new System.Drawing.Graphics())
{
```

```
//some code  
} //canvas.Dispose() gets called automatically
```

How do you trigger the Paint event in System.Drawing?

Invalidate the current form, the OS will take care of repainting. The Update method forces the repaint.

With these events, why wouldn't Microsoft combine Invalidate and Paint, so that you wouldn't have to tell it to repaint, and then to force it to repaint?

Painting is the slowest thing the OS does, so usually telling it to repaint, but not forcing it allows for the process to take place in the background.

How can you assign an RGB color to a System.Drawing.Color object?

Call the static method FromArgb of this class and pass it the RGB values.

What class does Icon derive from? Isn't it just a Bitmap with a wrapper name around it?

No, Icon lives in System.Drawing namespace. It's not a Bitmap by default, and is treated separately by .NET. However, you can use ToBitmap method to get a valid Bitmap object from a valid Icon object.

Before in my VB app I would just load the icons from DLL. How can I load the icons provided by .NET dynamically?

By using System.Drawing.SystemIcons class, for example System.Drawing.SystemIcons.Warning produces an Icon with a warning sign in it.

When displaying fonts, what's the difference between pixels, points and ems?

A pixel is the lowest-resolution dot the computer monitor supports. Its size depends on user's settings and monitor size. A point is always 1/72 of an inch. An em is the number of pixels that it takes to display the letter M.

What is the difference between VB 6 and VB.NET?

Answer1

VB

- 1, Object-based Language
- 2, Does not support Threading
- 3, Not powerful Exception handling mechanism
- 4, Does not have support for the console based applications
- 5, Cannot use more than one version of COM objects in VB application called DLL error
- 6, Does not support for the Disconnected data source.

VB.Net

- 1, Object-oriented Language
- 2, supports Threading
- 3, powerful Exception handling mechanism
- 4, having support for the console based applications
- 5, More than one version of DLL is supported
- 6, supports the Disconnected data source by using Dataset class

Answer2

VB:

1. Object-based language
2. Does not support inheritance
3. ADO.Net does not give support for disconnected data architecture
4. No interoperability function
5. No support for threading

VB.Net

1. Object-Oriented Programming language
2. ADO.Net gives support for disconnected data architecture
3. It provides interoperability
4. It uses managed code
5. supports threading
6. provides access to third-party controls like COM, DCOM

Answer2

1. The concept of the complete flow of execution of a program from start to finish: Visual Basic hides this aspect of programs from you, so that the only elements of a Visual Basic program you code are the event handlers and any methods in class modules. C# makes the complete program available to you as source code. The reason for this has to do with the fact that C# can be seen, philosophically, as next-generation C++. The roots of C++ go back to the 1960s and predate windowed user interfaces and sophisticated operating systems. C++ evolved as a low-level, closeto- the-machine, all-purpose language. To write GUI applications with C++ meant that you had to invoke the system calls to create and interact with the windowed forms. C# has been designed to build on this tradition while simplifying and modernizing C++, to combine the low-level performance benefits of C++ with the ease of coding in Visual Basic. Visual Basic, on the other hand, is designed specifically for rapid application development of Windows GUI applications. For this reason, in Visual Basic all the GUI boilerplate code is hidden, and all the Visual Basic programmer implements are the event handlers. In C# on the other hand, this boilerplate code is exposed as part of your source code.

2. Classes and inheritance: C# is a genuine object-oriented language, unlike Visual Basic, requiring all code to be a part of a class. It also includes extensive support for implementation inheritance. Indeed, most well-designed C# programs will be very much designed around this form of inheritance, which is completely absent in Visual Basic.

What are the authentication methods in .NET?

There are 4 types of authentications.

1. WINDOWS AUTHENTICATION
2. FORMS AUTHENTICATION
3. PASSPORT AUTHENTICATION
4. NONE/CUSTOM AUTHENTICATION

The authentication option for the ASP.NET application is specified by using the tag in the Web.config file, as shown below:

other authentication options

1. WINDOWS AUTHENTICATION Schemes
- I. Integrated Windows authentication

- II. Basic and basic with SSL authentication
- III. Digest authentication
- IV. Client Certificate authentication

2. FORMS AUTHENTICATION

You, as a Web application developer, are supposed to develop the Web page and authenticate the user by checking the provided user ID and password against some user database

3.PASSPORT AUTHENTICATION

A centralized service provided by Microsoft, offers a single logon point for clients. Unauthenticated users are redirected to the Passport site

4 NONE/CUSTOM AUTHENTICATION:

If we don't want ASP.NET to perform any authentication, we can set the authentication mode to "none". The reason behind this decision could be: We don't want to authenticate our users, and our Web site is open for all to use. We want to provide our own custom authentication

What is Serialization in .NET?

Answer1

The serialization is the process of converting the objects into stream of bytes. they or used for transport the objects(via remoting) and persist objects(via files and databases)

Answer2

When developing smaller applications that do not have a database (or other formal storage mechanism) or data that doesn't need to be stored in a database (such as the state of a web application), you often still would like to save the data for later retrieval. There are many ways to do this, but many of them are subject to a lot of extra code (work) and extra time spent debugging. With .NET, there is now an easy way to add this functionality to your code with only a few lines of easily tested code. This easy way is called serialization.

Serialization is the process of storing an object, including all of its public and private fields, to a stream. Deserialization is the opposite – restoring an object's field values from a stream. The stream is generally in the form of a FileStream, but does not have to be. It could be a memory stream or any other object that is of type IO.Stream. The format can be anything from XML to binary to SOAP.

What's the use of System.Diagnostics.Process class?

By using System.Diagnostics.Process class, we can provide access to the files which are presented in the local and remote system.

Example: System.Diagnostics.Process("c:\mlaks\example.txt") — local file

System.Diagnostics.Process("http://www.mlaks.com\example.txt") — remote file

What are the authentication methods in .NET?

Abstract class: This class has abstract methods (no body). This class cannot be instantiated. One needs to provide the implementation of the methods by overriding them in the derived class. No Multiple Inheritance.

Interfaces: Interface class contains all abstract methods which are public by default. All of these methods must be implemented in the derived class. One can inherit from from more than one interface thus provides for Multiple Inheritance.

re-clarification of object based:

VB6 DOES support polymorphism and interface inheritance. It also supports the “Implements” keyword. What is not supported in vb6 is implementation inheritance.

Also, from above, vb6 DOES “provides access to third-party controls like COM, DCOM ” That is not anything new in .NET.

How to achieve Polymorphism in VB.Net?

We can achieve polymorphism in .Net i.e Compile time polymorphism and Runtime polymorphism. Compiletime Polymorphism achieved by method overloading. Runtime polymorphism achieved by Early Binding or Late Binding. Provide the function pointer to the object at compile time called as Early Binding.

provide the function pointer to the object at runtime called as Late Binding

class emp having the method display()

class dept having the method display()

create objects as in the main function

```
// Early binding
```

```
dim obj as new emp
```

```
dim ob as new dept
```

obj.display()-to call the display method of emp class

ob.display()-to call the display method of the dept class

```
// Late binding
```

create object in the main class as

```
object obj
```

```
obj=new emp
```

```
obj.display()-to call the display of emp class
```

```
obj=new dept
```

```
obj.display()-to call the display of dept class
```

Difference between Class And Interface

Class is logical representation of object. It is collection of data and related sub procedures with definition.

Interface is also a class containing methods which is not having any definitions.

Class does not support multiple inheritance. But interface can support.

What doesu mean by .NET framework?

The .NET Framework is an environment for building, deploying, and running Web Services and other applications. It consists of three main parts: the Common Language Runtime, the Framework classes, and ASP.NET

What is assembly?

It is a single deployable unit that contains all the information about the implementation of classes, structures and interfaces

What is namespaces?

It is a logical group of related classes and interfaces and that can be used by any language targeting the .net framework.

.NET framework programming interview questions

.NET framework overview

1. Has own class libraries. System is the main namespace and all other namespaces are subsets of this.
2. It has CLR(Common language runtime, Common type system, common language specification)
3. All the types are part of CTS and Object is the base class for all the types.
4. If a language said to be .net complaint, it should be compatible with CTS and CLS.
5. All the code compiled into an intermediate language by the .Net language compiler, which is nothing but an assembly.
6. During runtime, JIT of CLR picks the IL code and converts into PE machine code and from there it processes the request.
7. CTS, CLS, CLR
8. Garbage Collection
9. Dispose, finalize, suppress finalize, IDisposable interface
10. Assemblies, Namespace: Assembly is a collection of class/namespaces. An assembly contains Manifest, Metadata, Resource files, IL code
11. Com interoperability, adding references, web references
12. Database connectivity and providers

Application Domain

1. Class modifiers: public, private, friend, protected, protected friend, mustinherit, NotInheritable
2. Method modifiers: public, private
3. Overridable
4. Shadows
5. Overloadable
6. Overrides
7. Overloads
8. Set/Get Property
9. IIF
10. Inheritance
11. Polymorphism
12. Delegates
13. Events
14. Reflection
15. Boxing
16. UnBoxing

ASP.Net

1. Web Controls: Data grid (templates, sorting, paging, bound columns, unbound columns, data binding), Data list, repeater controls
2. HTML Controls
3. Code behind pages, system.web.ui.page base class
4. Web.config: App settings, identity (impersonate), authentication (windows, forms, anonymous, passport), authorization

5. Databind.eval
6. Trace, Debug
7. Output cache
8. Session management
9. Application, Session
10. Global.asax httpapplication
11. User controls, custom controls, custom rendered controls (postback event, postdatachanged event)
usercontrol is the base class
12. Directives

ADO.Net

1. Command object (ExecuteNonQuery, ExecuteReader, ExecuteXMLReader, ExecuteScalar)
2. DataAdapter object (Fill)
3. Dataset (collection of tables)
4. CommandBuiler object
5. Transaction Object
6. Isolation levels

What is the base class of .NET?

Base class provides a base set of methods that all derived classes can use

Explain assemblies.

Answer 1:

Assemblies are similar to dll files. Both has the reusable pieces of code in the form of classes/ functions. Dll needs to be registered but assemblies have its own metadata.

Answer 2:

Assembly is a single deployable unit that contains information about the implementation of classes, structures and interfaces. it also stores the information about itself called metadata and includes name and verison of the assembly, security information, information about the dependencies and the list of files that constitute the assembly.

Assembly also contains namespaces. In the .Net Framework, applications are deployed in the form of assemblies.

Answer 3:

An assembly is a single deployable unit that contains all the information about the implementation of :

- classes
- structures and
- interfaces

An assembly stores all the information about itself. This information is called METADATA and include the name and the verison number of the assembly, security information, information about the dependencies and a lost of files that constitute the assembly.

All the application developed using the .NET framework are made up of assemblies.

Namespaces are also stored in assemblies

Answer 4:

In the Microsoft .NET framework an assembly is a partially compiled code library for use in deployment, versioning and security. In the Microsoft Windows implementation of .NET, an assembly is a PE (portable executable) file. There are two types, process assemblies (EXE) and library assemblies (DLL). A process assembly represents a process which will use classes defined in library assemblies. In version 1.1 of the CLR classes can only be exported from library assemblies; in version 2.0 this restriction is relaxed. The compiler will have a switch to determine if the assembly is a process or library and will set a flag in the PE file. .NET does not use the extension to determine if the file is a process or library. This means that a library may have either .dll or .exe as its extension.

The code in an assembly is partially compiled into CIL, which is then fully compiled into machine language at runtime by the CLR.

An assembly can consist of one or more files. Code files are called modules. An assembly can contain more than one code module and since it is possible to use different languages to create code modules this means that it is technically possible to use several different languages to create an assembly. In practice this rarely happens, principally because Visual Studio only allows developers to create assemblies that consist of a single code module.

Name some of the languages .NET support?

Some of the languages that are supported by .NET

1. Visual Basic.NET
2. Visual C#
3. Visual C++

ADO.NET features? Benefits? Drawbacks?

Answer 1:

1. Data will be retrieved through Datasets
2. Scalability

Answer 2:

1. Disconnected Data Architecture
2. Data cached in Datasets
3. Data transfer in XML format
4. Interaction with the database is done through data commands

How many types of exception handlers are there in .NET?

Answer 1:

From

MSDN>gt; "How the Runtime Manages Exceptions"

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconexceptionsoverview.asp>

The exception information table represents four types of exception handlers for protected blocks:

A finally handler that executes whenever the block exits, whether that occurs by normal control flow or by an unhandled exception.

A fault handler that must execute if an exception occurs, but does not execute on completion of normal control flow.

A type-filtered handler that handles any exception of a specified class or any of its derived classes.

A user-filtered handler that runs user-specified code to determine whether the exception should be handled by the associated handler or should be passed to the next protected block.

Answer 2:

1. Unstructured Exception Handling
2. Structured Exception Handling

Difference between Panel and GroupBox classes?

Answer 1:

Panel and Group box both can used as container for other controls like radio buttons and check box. the difference in panel and group box are Panel

- 1) In case of panel captions cannot be displayed
- 2) Can have scroll bars.

Group box

- 1) Captions can be displayed.
- 2) Cannot have a scroll bar

Answer 2:

Panel is scrollable. In panel you can't set caption like Group box.

What is the base class of Button control?

Listing from visual studio .net > Button Class

System.Object

System.MarshalByRefObject

System.ComponentModel.Component

System.Windows.Forms.Control

System.Windows.Forms.ButtonBase

System.Windows.Forms.Button

What is Response object? How is it related to ASP's Response object?

Response object allows the server to communicate with the client(browser). It is useful for displaying information to the user (or) redirecting the client.

Eg: Response.Write("Hello World")

hat is IIS? Have you used it?

IIS - Internet Information Server

IIS is used to access the ASP.Net web applications

Yes, I used in ASP.NET web applications.

Main differences between ASP and ASP.NET.

Answer 1:

1. ASP: Code is Interpreted
ASP.NET: Code is Compiled

2. ASP: Business Logic and Presentation Logic are in a single file

ASP.NET: Business Logic and Presentation Logic are in separate files (.cs or .vb) and (.aspx) respectively.

3. ASP: No Web Server Controls

ASP.NET: Web Server Controls supported by strong .NET Framework

4. ASP: No RAD in Classic ASP

ASP.NET: Supports RAD

Answer 2:

1. Asp is interpreted

Asp.net is compiled which is faster than asp.

2 Asp.net maintains its own CLR and is managed as it runs by CLR

Where as asp is unmanaged

3 We can maintain sessions in state server and sql server which is Outproc, where in asp sessions will be lost if we restart webserver or make changes.

4 In asp.net we can configure each application using web.config file which is available in application itself and we have machine.config where we can configure all applications.

In asp we cannot configure single application

5 Asp.net we have autopostback event which is not in asp

6 In asp.net we have global.asax where we can handle some global things which is not in asp.

7 We have well built GUI to work in asp.net

8 We have ado.net and as well as disconnected architecture in asp.net

9 We have Xcopy deployment in asp.net

10. We can work with any language as code behind technique in asp.net that supports .net framework

Answer 3:

a) asp.net is compiled but ASP is an interpreter or script only.

b) asp.net is supported more control than the asp.

c) asp.net is more supported even control than the asp.

d) In asp.net if update any component then no need to shutdown the computer but in asp if loaded any component then need to be shutdown the computer.

d) So lastly an asp.net is faster than asp .

What are the advantages and drawbacks of using ADO.NET?

Pros

ADO.NET is rich with plenty of features that are bound to impress even the most skeptical of programmers. If this weren't the case, Microsoft wouldn't even be able to get anyone to use the Beta. What we've done here is come up with a short list of some of the more outstanding benefits to using the ADO.NET architecture and the System.Data namespace.

* Performance – there is no doubt that ADO.NET is extremely fast. The actual figures vary depending on who performed the test and which benchmark was being used, but ADO.NET performs much, much faster at the same tasks than its predecessor, ADO. Some of the reasons why ADO.NET is faster than ADO are discussed in the ADO versus ADO.NET section later in this chapter.

* Optimized SQL Provider – in addition to performing well under general circumstances, ADO.NET includes a SQL Server Data Provider that is highly optimized for interaction with SQL Server. It uses SQL Server's own TDS (Tabular Data Stream) format for exchanging information. Without question, your SQL Server 7 and above data access operations will run blazingly fast utilizing this optimized Data Provider.

* XML Support (and Reliance) – everything you do in ADO.NET at some point will boil down to the use of XML. In fact, many of the classes in ADO.NET, such as the DataSet, are so intertwined with XML that they simply cannot exist or function without utilizing the technology. You’ll see later when we compare and contrast the “old” and the “new” why the reliance on XML for internal storage provides many, many advantages, both to the framework and to the programmer utilizing the class library.

* Disconnected Operation Model – the core ADO.NET class, the DataSet, operates in an entirely disconnected fashion. This may be new to some programmers, but it is a remarkably efficient and scalable architecture. Because the disconnected model allows for the DataSet class to be unaware of the origin of its data, an unlimited number of supported data sources can be plugged into code without any hassle in the future.

* Rich Object Model – the entire ADO.NET architecture is built on a hierarchy of class inheritance and interface implementation. Once you start looking for things you need within this namespace, you’ll find that the logical inheritance of features and base class support makes the entire system extremely easy to use, and very customizable to suit your own needs. It is just another example of how everything in the .NET framework is pushing toward a trend of strong application design and strong OOP implementations.

Cons

=====
Hard as it may be to believe, there are a couple of drawbacks or disadvantages to using the ADO.NET architecture. I’m sure others can find many more faults than we list here, but we decided to stick with a short list of some of the more obvious and important shortcomings of the technology.

* Managed-Only Access – for a few obvious reasons, and some far more technical, you cannot utilize the ADO.NET architecture from anything but managed code. This means that there is no COM interoperability allowed for ADO.NET. Therefore, in order to take advantage of the advanced SQL Server Data Provider and any other feature like DataSets, XML internal data storage, etc, your code must be running under the CLR.

* Only Three Managed Data Providers (so far) – unfortunately, if you need to access any data that requires a driver that cannot be used through either an OLEDB provider or the SQL Server Data Provider, then you may be out of luck. However, the good news is that the OLEDB provider for ODBC is available for download from Microsoft. At that point the down-side becomes one of performance, in which you are invoking multiple layers of abstraction as well as crossing the COM InterOp gap, incurring some initial overhead as well.

* Learning Curve – despite the misleading name, ADO.NET is not simply a new version of ADO, nor should it even be considered a direct successor. ADO.NET should be thought of more as the data access class library for use with the .NET framework. The difficulty in learning to use ADO.NET to its fullest is that a lot of it does seem familiar. It is this that causes some common pitfalls. Programmers need to learn that even though some syntax may appear the same, there is actually a considerable amount of difference in the internal workings of many classes. For example (this will be discussed in

far more detail later), an ADO.NET DataSet is nothing at all like a disconnected ADO RecordSet. Some may consider a learning curve a drawback, but I consider learning curves more like scheduling issues. There's a learning curve in learning anything new; it's just up to you to schedule that curve into your time so that you can learn the new technology at a pace that fits your schedule.

Why The JavaScript Validation Not Run on the Asp.Net Button But Run Successfully On The HTML Button

The Asp.Net Button Is post backed on the server & not yet Submit & when It goes to the server its states is lost So if we r using javascript in our application so we always use the Input Button in the asp Button

what is the difference between user control an custom control? advantages/disadvantages?

Web user controls Vs Web custom controls

Easier to create Vs Harder to create

Limited support for consumers who use a visual design tool Vs Full visual design tool support for consumers

A separate copy of the control is required in each application Vs Only a single copy of the control is required, in the global assembly cache

Cannot be added to the Toolbox in Visual Studio Vs Can be added to the Toolbox in Visual Studio

Good for static layout Vs Good for dynamic layout

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

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

What is the use of ErrorProvider Control?

The ErrorProvider control is used to indicate invalid data on a data entry form. Using this control, you can attach error messages that display next to the control when the data is invalid, as seen in the following image. A red circle with an exclamation point blinks, and when the user mouses over the icon, the error message is displayed as a tooltip.

What is CLR?

Answer 1:

CLR(Common Language Runtime) is the main resource of .Net Framework. it is collection of services like garbage collector, exception handler, jit compilers etc. with the CLR cross language integration is possible.

Answer 2:

The .NET Framework provides a runtime environment which runs the code and provides services that make the development process easier. This runtime environment in .NET Framework is known as Common Language Runtime (CLR). The CLR sits at the very heart of managed code. Common Language Runtime is the generalized multi-language, reflective execution engine on which code originally written in various languages runs. At a higher level, CLR is simply an engine that takes in Intermediate Language (IL) instructions, translates them into machine instructions, and executes them. Although the common language runtime provides many standard runtime services, managed code is never interpreted. A feature called just-in-time (JIT) compiling enables all managed code to run in the native machine language of the system on which it is executing. The CLR shares much in common with a traditional operating system.

Quote:

Visit <http://www.downloadmela.com/> for more papers

Managed code is the term used for any code that is running on .NET Framework.

The CLR provides the infrastructure that enables managed code to execute as well provides variety of services during execution. When a method, for which IL has been generated, is called for the first time the CLR compiles the IL into native code that is specific to the processor the Environment it is running on (This process is known as Just in Time Compilation or JIT). If the same method is called next time, the existing JIT compiled code is reused. During execution managed code receives variety of services from the runtime environment.

Quote:

When compiling to managed code, the compiler translates your source code into Microsoft intermediate language (MSIL), which is a CPU-independent set of instructions that can be efficiently converted to native code. Intermediate Language is a binary assembly language that is compiled at runtime down to whatever machine language is appropriate for the host CPU. This runtime compilation is called Just-In-Time Compiling or JIT-compiling.

Advantages of Managed Execution Environments

In unmanaged environments the compiler and linker directly compile the source code in to native instructions that are targeted at a specific processor. The disadvantage of this process is that each time you want to run your executable on a different platform you will have to re-compile the code using a compiler and linker that will compile the code that is targeted at the specific hardware. This means that each time you want your application to run on a different platform, you will have to ship the compiled instructions again and again. As this leads to compiling and maintaining multiple versions of the same application, the companies try to create a more generalized compiled version in order to target most of the environments. This process is known as the Lowest Common Denominator approach. This leads to a more generalized program which is not optimized properly and does not take advantages of the underlying hardware infrastructure (processor, cache, etc). Because the CLR supplies one or more Just in Time Compiler for each computer architecture it supports, the same set of MSIL can be JIT-compiled and run on any supported architecture. This

CLR provides the following benefits for developers:

- Vastly simplified development.
- Seamless integration of code written in various languages.
- Evidence-based security with code identity.
- Assembly-based deployment that eliminates DLL Hell.
- Side-by-side versioning of reusable components.
- Code reuse through implementation inheritance.
- Automatic object lifetime management.
- Code access security.
- Cross Language Integration.
- Self describing objects.

The CLR automatically handles object layout and manages references to objects, releasing them when they are no longer being used. This automatic memory management resolves the two most common application errors, memory leaks and invalid memory references. This process is known as Garbage Collection. The CLR also manages thread execution, code execution, code safety verification,

compilation, and other system services.

The CLR is designed for the software of the future, and it also supports software of today and yesterday. Interoperability between managed and unmanaged code provided by CLR helps developers continue to use necessary COM components and DLLs.

What is Delegate and what is it used for ?

Delegate is kinda like a pointer to a function in C++ or like an event handler in Java

You can use it to “multicast” which means running multiple functions in different instances of object already created.

This is useful when you want your objects to “register” to an event raised by another object.

The way it works is the object you are registered to listen to receives the delegate of the function it is supposed to run in your object, the delegate is then run from it. (if you switch the word delegate for pointer, this would be much simpler)

How is meant by DLL ?

A DLL (Dynamic Link Library) is a file that can be loaded and executed by programs dynamically.

Basically it’s an external code repository for programs. Since usually several different programs reuse the same DLL instead of having that code in their own file, this dramatically reduces required storage space. A synonym for a DLL would be library.

Which DLL translate XML to SQL in IIS?

Sqlisapi.dll

Can anyone tell me about Secure Socket Layer? How to make use of the technology?

Secure Sockets Layer (SSL) and Transport Layer Security (TLS), its successor, are cryptographic protocols which provide secure communications on the Internet. There are slight differences between SSL 3.0 and TLS 1.0, but the protocol remains substantially the same. The term “SSL” as used here applies to both protocols unless clarified by context.

What is the Difference Between Response.write & response.output.Write

In ASP.NET the Response object is of type HttpResponse and when you say Response.Write you’re really saying (basically) HttpContext.Current.Response.Write and calling one of the many overloaded Write methods of HttpResponse.

Response.Write then calls .Write() on it’s internal TextWriter object:

```
public void Write(object obj){ this._writer.Write(obj);}
```

HttpResponse also has a Property called Output that is of type, yes, TextWriter, so:

```
public TextWriter get_Output(){ return this._writer; }
```

Which means you can to the Response whatever a TextWriter will let you. Now, TextWriters support a Write() method ala String.Format, so you can do this:

```
Response.Output.Write(“Scott is {0} at {1:d}”, “cool”,DateTime.Now);
```

But internally, of course, this this is happening:

```
public virtual void Write(string format, params object[] arg)
{
this.Write(string.Format(format, arg));
}
```

Which dll is required to translate XML to SQL in IIS ?

Microsoft.data.sqlxml.dll

What is an interface and what is an abstract class? Please, expand by examples of using both. Explain why.

Answers1:

In a interface class, all methods are abstract without implementation where as in an abstract class some methods we can define concrete. In interface, no accessibility modifiers are allowed. An abstract class may have accessibility modifiers. Interface and abstract class are basically a set of rules which u have to follow in case u r using them(inheriting them).

Answers2:

Abstract classes are closely related to interfaces. They are classes that cannot be instantiated, and are frequently either partially implemented, or not at all implemented. One key difference between abstract classes and interfaces is that a class may implement an unlimited number of interfaces, but may inherit from only one abstract (or any other kind of) class. A class that is derived from an abstract class may still implement interfaces. Abstract classes are useful when creating components because they allow you specify an invariant level of functionality in some methods, but leave the implementation of other methods until a specific implementation of that class is needed. They also version well, because if additional functionality is needed in derived classes, it can be added to the base class without breaking code.

Answers3:

Abstract Classes

An abstract class is the one that is not used to create objects. An abstract class is designed to act as a base class (to be inherited by other classes). Abstract class is a design concept in program development and provides a base upon which other classes are built. Abstract classes are similar to interfaces. After declaring an abstract class, it cannot be instantiated on it's own, it must be inherited. Like interfaces, abstract classes can specify members that must be implemented in inheriting classes. Unlike interfaces, a class can inherit only one abstract class. Abstract classes can only specify members that should be implemented by all inheriting classes.

Answers4:

An interface looks like a class, but has no implementation. They're great for putting together plug-n-play like architectures where components can be interchanged at will. Think Firefox Plug-in extension implementation. If you need to change your design, make it an interface. However, you may have abstract classes that provide some default behavior. Abstract classes are excellent candidates inside of application frameworks.

Answers5:

One additional key difference between interfaces and abstract classes (possibly the most important one) is that multiple interfaces can be implemented by a class, but only one abstract class can be inherited by any single class.

Some background on this: C++ supports multiple inheritance, but C# does not. Multiple inheritance in C++ has always be controversial, because the resolution of multiple inherited implementations of the same method from different base classes is hard to control and anticipate. C# decided to avoid this problem by allowing a class to implement multiple interfaces, which do not contain method implementations, but restricting a class to have at most a single parent class. Although this can result in redundant implementations of the same method when different classes implement the same interface, it

is still an excellent compromise.

Another difference between interfaces and abstract classes is that an interface can be implemented by an abstract class, but no class, abstract or otherwise, can be inherited by an interface.

Answers6:

What is an Abstract class?

An abstract class is a special kind of class that cannot be instantiated. So the question is why we need a class that cannot be instantiated? An abstract class is only to be sub-classed (inherited from). In other words, it only allows other classes to inherit from it but cannot be instantiated. The advantage is that it enforces certain hierarchies for all the subclasses. In simple words, it is a kind of contract that forces all the subclasses to carry on the same hierarchies or standards.

What is an Interface?

An interface is not a class. It is an entity that is defined by the word Interface. An interface has no implementation; it only has the signature or in other words, just the definition of the methods without the body. As one of the similarities to Abstract class, it is a contract that is used to define hierarchies for all subclasses or it defines specific set of methods and their arguments. The main difference between them is that a class can implement more than one interface but can only inherit from one abstract class. Since C# doesn't support multiple inheritance, interfaces are used to implement multiple inheritance.

What is serialization, how it works in .NET?

Serialization is when you persist the state of an object to a storage medium so an exact copy can be re-created at a later stage.

Serialization is used to save session state in ASP.NET.

Serialization is to copy objects to the Clipboard in Windows Forms

Serialization is used by remoting to pass objects by value from one application domain to another

What should one do to make class serializable?

Answers1:

To make a class serializable is to mark it with the Serializable attribute as follows.

```
[Serializable]
public class MyObject {
public int n1 = 0;
public int n2 = 0;
public String str = null;
}
```

What exactly is being serialized when you perform serialization?

The object's state (values)

How does output caching work in ASP.NET?

Output caching is a powerful technique that increases request/response throughput by caching the content generated from dynamic pages. Output caching is enabled by default, but output from any given response is not cached unless explicit action is taken to make the response cacheable.

To make a response eligible for output caching, it must have a valid expiration/validation policy and public cache visibility. This can be done using either the low-level OutputCache API or the high-level @ OutputCache directive. When output caching is enabled, an output cache entry is created on the first GET request to the page. Subsequent GET or HEAD requests are served from the output cache entry

until the cached request expires.

The output cache also supports variations of cached GET or POST name/value pairs.

The output cache respects the expiration and validation policies for pages. If a page is in the output cache and has been marked with an expiration policy that indicates that the page expires 60 minutes from the time it is cached, the page is removed from the output cache after 60 minutes. If another request is received after that time, the page code is executed and the page can be cached again. This type of expiration policy is called absolute expiration - a page is valid until a certain time.

What is connection pooling and how do you make your application use it?

Opening database connection is a time consuming operation.

Connection pooling increases the performance of the applications by reusing the active database connections instead of create new connection for every request.

Connection pooling Behaviour is controlled by the connection string parameters.

Following the the 4 parameters that control most of the connection pooling behaviour.

1. Connect Timeout
2. Max Pool Size
3. Min Pool Size
4. Pooling

What are different methods of session maintenance in ASP.NET?

3 types:

In-process storage.

Session State Service.

Microsoft SQL Server.

In-Process Storage

The default location for session state storage is in the ASP.NET process itself.

Session State Service

As an alternative to using in-process storage for session state, ASP.NET provides the ASP.NET State Service. The State Service gives you an out-of-process alternative for storing session state that is not tied quite so closely to ASP. Net's own process.

To use the State Service, you need to edit the sessionState element in your ASP.NET application's web.config file:

You'll also need to start the ASP.NET State Service on the computer that you specified in the stateConnectionString attribute. The .NET Framework installs this service, but by default it's set to manual startup. If you're going to depend on it for storing session state, you'll want to change that to automatic startup by using the Services MMC plug-in in the Administrative Tools group.

If you make these changes, and then repeat the previous set of steps, you'll see slightly different behavior: session state persists even if you recycle the ASP.NET process.

There are two main advantages to using the State Service. First, it is not running in the same process as ASP.NET, so a crash of ASP.NET will not destroy session information. Second, the stateConnectionString that's used to locate the State Service includes the TCP/IP address of the service, which need not be running on the same computer as ASP.NET. This allows you to share state

information across a web garden (multiple processors on the same computer) or even across a web farm (multiple servers running the application). With the default in-process storage, you can't share state information between multiple instances of your application.

The major disadvantage of using the State Service is that it's an external process, rather than part of ASP.NET. That means that reading and writing session state is slower than it would be if you kept the state in-process. And, of course, it's one more process that you need to manage. As an example of the extra effort that this can entail, there is a bug in the initial release of the State Service that allows a determined attacker to crash the ASP.NET process remotely. If you're using the State Service to store session state, you should install the patch from Microsoft Security Bulletin MS02-66, or install SP2 for the .NET Framework.

Microsoft SQL Server

The final choice for storing state information is to save it in a Microsoft SQL Server database. To use SQL Server for storing session state, you need to perform several setup steps:

Run the InstallSqlState.sql script on the Microsoft SQL Server where you intend to store session state. This script will create the necessary database and database objects. The .NET Framework installs this script in the same folder as its compilers and other tools—for example, C:\WINNT\Microsoft.NET\Framework\v1.0.3705 on a Windows 2000 computer with the 1.0 version of the Framework. Edit the sessionState element in the web.config file for your ASP.NET application as follows:

Supply the server name, user name, and password for a SQL Server account that has access to the session state database in the sqlConnectionString attribute.

Like the State Service, SQL Server lets you share session state among the processors in a web garden or the servers in a web farm. But you also get the additional benefit of persistent storage. Even if the computer hosting SQL Server crashes and is restarted, the session state information will still be present in the database, and will be available as soon as the database is running again. That's because SQL Server, being an industrial-strength database, is designed to log its operations and protect your data at (almost) all costs. If you're willing to invest in SQL Server clustering, you can keep the session state data available transparently to ASP.NET even if the primary SQL Server computer crashes.

Like the State Service, SQL Server is slower than keeping session state in process. You also need to pay additional licensing fees to use SQL Server for session state in a production application. And, of course, you need to worry about SQL Server-specific threats such as the "Slammer" worm.

What is Viewstate?

A server control's view state is the accumulation of all its property values. In order to preserve these values across HTTP requests, ASP.NET server controls use this property, which is an instance of the StateBag class, to store the property values.

Can any object be stored in a Viewstate?

An object that either is serializable or has a TypeConverter defined for it can be persisted in ViewState

What should you do to store an object in a Viewstate?

Do serialization of convert the object to string

Explain how ViewState is being formed and how it's stored on client.

The type of ViewState is System.Web.UI.StateBag, which is a dictionary that stores name/value pairs. ViewState is persisted to a string variable by the ASP.NET page framework and sent to the client and back as a hidden variable. Upon postback, the page framework parses the input string from the hidden variable and populates the ViewState property of each control. If a control uses ViewState for property data instead of a private field, that property automatically will be persisted across round trips to the client. (If a property is not persisted in ViewState, it is good practice to return its default value on postback.)

What do you know about ADO.NET's objects and methods?

ADO.NET provides consistent access to data sources such as Microsoft SQL Server, as well as data sources exposed through OLE DB and XML.

Data-sharing consumer applications can use ADO.NET to connect to these different data sources and retrieve, manipulate, and update data.

ADO.NET provides first-class support for the disconnected, n-tier programming environment for which many new applications are written.

Explain DataSet.AcceptChanges and DataAdapter.Update methods.

DataAdapter.Update method Calls the respective INSERT, UPDATE, or DELETE statements for each inserted, updated, or deleted row in the DataSet.

DataSet.AcceptChanges method Commits all the changes made to this row since the last time AcceptChanges was called.

When we go for html server controls and when we go for web server controls?

Server controls are a part of ASP.net. When a server control is used there will be an extra overhead on the server to create the control at runtime and accordingly set the values. HTML controls are static controls and are easy to use. They are supported in ASP.net.

As a rule, if there is a corresponding HTML control available instead of the server control, you should always go for the HTML control as it enhances the server performance and ensures faster response.

Server controls should be used when it is found that the available HTML controls are not sufficient to achieve the task.

How many languages .NET is supporting now?

When .NET was introduced it came with several languages. VB.NET, C#, COBOL and Perl, etc. 44 languages are supported.

How is .NET able to support multiple languages?

A language should comply with the Common Language Runtime standard to become a .NET language. In .NET, code is compiled to Microsoft Intermediate Language (MSIL for short). This is called as Managed Code. This Managed code is run in .NET environment. So after compilation to this IL the language is not a barrier. A code can call or use a function written in another language.

How ASP .NET different from ASP?

Scripting is separated from the HTML, Code is compiled as a DLL, these DLLs can be executed on the server.

What is smart navigation?

The cursor position is maintained when the page gets refreshed due to the server side validation and the page gets refreshed.

What is view state?

The web is stateless. But in ASP.NET, the state of a page is maintained in the in the page itself automatically. How? The values are encrypted and saved in hidden controls. this is done automatically by the ASP.NET. This can be switched off / on for a single control

How do you validate the controls in an ASP .NET page?

Using special validation controls that are meant for this. We have Range Validator, Email Validator.

Can the validation be done in the server side? Or this can be done only in the Client side?

Client side is done by default. Server side validation is also possible. We can switch off the client side and server side can be done.

How to manage pagination in a page?

Using pagination option in DataGrid control. We have to set the number of records for a page, then it takes care of pagination by itself.

What is ADO .NET and what is difference between ADO and ADO.NET?

ADO.NET is stateless mechanism. I can treat the ADO.Net as a separate in-memory database where in I can use relationships between the tables and select insert and updates to the database. I can update the actual database as a batch.

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

Server side scripting means that all the script will be executed by the server and interpreted as needed. ASP doesn't have some of the functionality like sockets, uploading, etc. For these you have to make a custom components usually in VB or VC++. Client side scripting means that the script will be executed immediately in the browser such as form field validation, clock, email validation, etc. Client side scripting is usually done in VBScript or JavaScript. Download time, browser compatibility, and visible code - since JavaScript and VBScript code is included in the HTML page, then anyone can see the code by viewing the page source. Also a possible security hazards for the client computer.

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

C#

Should validation (did the user enter a real date) occur server-side or client-side? Why?

Client-side validation because there is no need to request a server side date when you could obtain a date from the client machine.

What does the "EnableViewState" property do? Why would I want it on or off?

Enable ViewState turns on the automatic state management feature that enables server controls to re-populate their values on a round trip without requiring you to write any code. This feature is not free however, since the state of a control is passed to and from the server in a hidden form field. You should be aware of when ViewState is helping you and when it is not. For example, if you are binding a control to data on every round trip (as in the datagrid example in tip #4), then you do not need the control to maintain it's view state, since you will wipe out any re-populated data in any case. ViewState is enabled for all server controls by default. To disable it, set the EnableViewState property of the control to false.

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

Why would I choose one over the other? Server.Transfer() : client is shown as it is on the requesting page only, but the all the content is of the requested page. Data can be persist across the pages using Context.Item collection, which is one of the best way to transfer data from one page to another keeping the page state alive. Response.Dedirect() :client know the physical location (page name and query

string as well). Context.Items loses the persistence when navigate to destination page. In earlier versions of IIS, if we wanted to send a user to a new Web page, the only option we had was Response.Redirect. While this method does accomplish our goal, it has several important drawbacks. The biggest problem is that this method causes each page to be treated as a separate transaction. Besides making it difficult to maintain your transactional integrity, Response.Redirect introduces some additional headaches. First, it prevents good encapsulation of code. Second, you lose access to all of the properties in the Request object. Sure, there are workarounds, but they're difficult. Finally, Response.Redirect necessitates a round trip to the client, which, on high-volume sites, causes scalability problems. As you might suspect, Server.Transfer fixes all of these problems. It does this by performing the transfer on the server without requiring a roundtrip to the client.

Can you give an example of when it would be appropriate to use a web service as opposed to a non-serviced .NET component?

When to Use Web Services:

* Communicating through a Firewall When building a distributed application with 100s/1000s of users spread over multiple locations, there is always the problem of communicating between client and server because of firewalls and proxy servers. Exposing your middle tier components as Web Services and invoking the directly from a Windows UI is a very valid option.

* Application Integration When integrating applications written in various languages and running on disparate systems. Or even applications running on the same platform that have been written by separate vendors.

* Business-to-Business Integration This is an enabler for B2B integration which allows one to expose vital business processes to authorized supplier and customers. An example would be exposing electronic ordering and invoicing, allowing customers to send you purchase orders and suppliers to send you invoices electronically.

* Software Reuse This takes place at multiple levels. Code Reuse at the Source code level or binary component-based reuse. The limiting factor here is that you can reuse the code but not the data behind it. Webservice overcome this limitation. A scenario could be when you are building an app that aggregates the functionality of several other Applications. Each of these functions could be performed by individual apps, but there is value in perhaps combining the multiple apps to present a unified view in a Portal or Intranet.

* When not to use Web Services: Single machine Applications When the apps are running on the same machine and need to communicate with each other use a native API. You also have the options of using component technologies such as COM or .NET Components as there is very little overhead.

* Homogeneous Applications on a LAN If you have Win32 or Winforms apps that want to communicate to their server counterpart. It is much more efficient to use DCOM in the case of Win32 apps and .NET Remoting in the case of .NET Apps.

Can you explain the difference between an ADO.NET Dataset and an ADO Recordset?

In ADO, the in-memory representation of data is the RecordSet. In ADO.NET, it is the dataset. There are important differences between them.

* A RecordSet looks like a single table. If a recordset is to contain data from multiple database tables, it must use a JOIN query, which assembles the data from the various database tables into a single result table. In contrast, a dataset is a collection of one or more tables. The tables within a dataset are called data tables; specifically, they are DataTable objects. If a dataset contains data from multiple database tables, it will typically contain multiple DataTable objects. That is, each DataTable object typically corresponds to a single database table or view. In this way, a dataset can mimic the structure of the underlying database. A dataset usually also contains relationships. A relationship within a dataset is analogous to a foreign-key relationship in a database —that is, it associates rows of the tables with each other. For example, if a dataset contains a table about investors and another table about each investor's stock purchases, it could also contain a relationship connecting each row of the investor table with the corresponding rows of the purchase table. Because the dataset can hold multiple, separate tables and maintain information about relationships between them, it can hold much richer data structures than a recordset, including self-relating tables and tables with many-to-many relationships.

* In ADO you scan sequentially through the rows of the recordset using the ADO MoveNext method. In ADO.NET, rows are represented as collections, so you can loop through a table as you would through any collection, or access particular rows via ordinal or primary key index. DataRelation objects maintain information about master and detail records and provide a method that allows you to get records related to the one you are working with. For example, starting from the row of the Investor table for "Nate Sun," you can navigate to the set of rows of the Purchase table describing his purchases. A cursor is a database element that controls record navigation, the ability to update data, and the visibility of changes made to the database by other users. ADO.NET does not have an inherent cursor object, but instead includes data classes that provide the functionality of a traditional cursor. For example, the functionality of a forward-only, read-only cursor is available in the ADO.NET DataReader object. For more information about cursor functionality, see Data Access Technologies.

* Minimized Open Connections: In ADO.NET you open connections only long enough to perform a database operation, such as a Select or Update. You can read rows into a dataset and then work with them without staying connected to the data source. In ADO the recordset can provide disconnected access, but ADO is designed primarily for connected access. There is one significant difference between disconnected processing in ADO and ADO.NET. In ADO you communicate with the database by making calls to an OLE DB provider. In ADO.NET you communicate with the database through a data adapter (an OleDbDataAdapter, SqlDataAdapter, OdbcDataAdapter, or OracleDataAdapter object), which makes calls to an OLE DB provider or the APIs provided by the underlying data source. The important difference is that in ADO.NET the data adapter allows you to control how the changes to the dataset are transmitted to the database — by optimizing for performance, performing data validation checks, or adding any other extra processing. Data adapters, data connections, data commands, and data readers are the components that make up a .NET Framework data provider. Microsoft and third-party providers can make available other .NET Framework data providers that can be integrated into Visual Studio.

* Sharing Data Between Applications. Transmitting an ADO.NET dataset between applications is much easier than transmitting an ADO disconnected recordset. To transmit an ADO disconnected recordset from one component to another, you use COM marshalling. To transmit data in ADO.NET, you use a dataset, which can transmit an XML stream.

* Richer data types.COM marshalling provides a limited set of data types — those defined by the COM standard. Because the transmission of datasets in ADO.NET is based on an XML format, there is no restriction on data types. Thus, the components sharing the dataset can use whatever rich set of data types they would ordinarily use.

* Performance. Transmitting a large ADO recordset or a large ADO.NET dataset can consume network resources; as the amount of data grows, the stress placed on the network also rises. Both ADO and ADO.NET let you minimize which data is transmitted. But ADO.NET offers another performance advantage, in that ADO.NET does not require data-type conversions. ADO, which requires COM marshalling to transmit records sets among components, does require that ADO data types be converted to COM data types.

* Penetrating Firewalls.A firewall can interfere with two components trying to transmit disconnected ADO recordsets. Remember, firewalls are typically configured to allow HTML text to pass, but to prevent system-level requests (such as COM marshalling) from passing.

Can you give an example of what might be best suited to place in the Application_Start and Session_Start subroutines?

The Application_Start event is guaranteed to occur only once throughout the lifetime of the application. It's a good place to initialize global variables. For example, you might want to retrieve a list of products from a database table and place the list in application state or the Cache object. SessionStateModule exposes both Session_Start and Session_End events.

If I'm developing an application that must accommodate multiple security levels though secure login and my ASP.NET web application is spanned across three web-servers (using round-robin load balancing) what would be the best approach to maintain login-in state for the users?

What are ASP.NET Web Forms? How is this technology different than what is available though ASP?

Web Forms are the heart and soul of ASP.NET. Web Forms are the User Interface (UI) elements that give your Web applications their look and feel. Web Forms are similar to Windows Forms in that they provide properties, methods, and events for the controls that are placed onto them. However, these UI elements render themselves in the appropriate markup language required by the request, e.g. HTML. If you use Microsoft Visual Studio .NET, you will also get the familiar drag-and-drop interface used to create your UI for your Web application.

How does VB.NET/C# achieve polymorphism?

By using Abstract classes/functions.

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

Inheritance is a fundamental feature of an object oriented system and it is simply the ability to inherit data and functionality from a parent object. Rather than developing new objects from scratch, new code can be based on the work of other programmers, adding only new features that are needed.

How would you implement inheritance using VB.NET/C#?

When we set out to implement a class using inheritance, we must first start with an existing class from which we will derive our new subclass. This existing class, or base class, may be part of the .NET system class library framework, it may be part of some other application or .NET assembly, or we may create it as part of our existing application. Once we have a base class, we can then implement one or

more subclasses based on that base class. Each of our subclasses will automatically have all of the methods, properties, and events of that base class ? including the implementation behind each method, property, and event. Our subclass can add new methods, properties, and events of its own - extending the original interface with new functionality. Additionally, a subclass can replace the methods and properties of the base class with its own new implementation - effectively overriding the original behavior and replacing it with new behaviors. Essentially inheritance is a way of merging functionality from an existing class into our new subclass. Inheritance also defines rules for how these methods, properties, and events can be merged.

What's an assembly?

Assemblies are the building blocks of .NET Framework applications; they form the fundamental unit of deployment, version control, reuse, activation scoping, and security permissions. An assembly is a collection of types and resources that are built to work together and form a logical unit of functionality. An assembly provides the common language runtime with the information it needs to be aware of type implementations. To the runtime, a type does not exist outside the context of an assembly.

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

ASP.NET supports two modes of page development: Page logic code that is written inside <script runat=server> blocks within an .aspx file and dynamically compiled the first time the page is requested on the server. Page logic code that is written within an external class that is compiled prior to deployment on a server and linked "behind" the .aspx file at run time.

Explain what a diffgram is, and a good use for one?

A DiffGram is an XML format that is used to identify current and original versions of data elements. The DataSet uses the DiffGram format to load and persist its contents, and to serialize its contents for transport across a network connection. When a DataSet is written as a DiffGram, it populates the DiffGram with all the necessary information to accurately recreate the contents, though not the schema, of the DataSet, including column values from both the Original and Current row versions, row error information, and row order.

Where would you use an IHttpModule, and what are the limitations of any approach you might take in implementing one?

One of ASP.NET's most useful features is the extensibility of the HTTP pipeline, the path that data takes between client and server. You can use them to extend your ASP.NET applications by adding pre- and post-processing to each HTTP request coming into your application. For example, if you wanted custom authentication facilities for your application, the best technique would be to intercept the request when it comes in and process the request in a custom HTTP module.

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

Every Page object (which your .aspx page is) has nine events, most of which you will not have to worry about in your day to day dealings with ASP.NET. The three that you will deal with the most are: Page_Init, Page_Load, Page_PreRender.

Which method do you invoke on the DataAdapter control to load your generated dataset with data?

System.Data.Common.DataAdapter.Fill(System.Data.DataSet);

If my DataAdapter is sqlDataAdapter and my DataSet is dsUsers then it is called this way:
sqlDataAdapter.Fill(dsUsers);

Which template must you provide, in order to display data in a Repeater control?

ItemTemplate

How can you provide an alternating color scheme in a Repeater control?

AlternatingItemTemplate Like the ItemTemplate element, but rendered for every other row (alternating items) in the Repeater control. You can specify a different appearance for the AlternatingItemTemplate element by setting its style properties.

What property must you set, and what method must you call in your code, in order to bind the data from some data source to the Repeater control?

You must set the DataMember property which Gets or sets the specific table in the DataSource to bind to the control and the DataBind method to bind data from a source to a server control. This method is commonly used after retrieving a data set through a database query.

What base class do all Web Forms inherit from?

System.Web.UI.Page

What method do you use to explicitly kill a user's session?

The Abandon method destroys all the objects stored in a Session object and releases their resources. If you do not call the Abandon method explicitly, the server destroys these objects when the session times out.

Syntax: Session.Abandon

How do you turn off cookies for one page in your site?

Use the Cookie.Discard Property which Gets or sets the discard flag set by the server. When true, this property instructs the client application not to save the Cookie on the user's hard disk when a session ends.

Which two properties are on every validation control?

ControlToValidate & ErrorMessage properties

How do you create a permanent cookie?

Setting the Expires property to MinValue means that the Cookie never expires.

Which method do you use to redirect the user to another page without performing a round trip to the client?

Server.transfer()

What is the transport protocol you use to call a Web service?

Answer1:

SOAP. Transport Protocols: It is essential for the acceptance of Web Services that they are based on established Internet infrastructure. This in fact imposes the usage of of the HTTP, SMTP and FTP protocols based on the TCP/IP family of transports. Messaging Protocol: The format of messages exchanged between Web Services clients and Web Services should be vendor neutral and should not carry details about the technology used to implement the service. Also, the message format should allow for extensions and different bindings to specific transport protocols. SOAP and ebXML Transport are specifications which fulfill these requirements. We expect that the W3C XML Protocol Working Group defines a successor standard.

Answer2:

SOAP is not the transport protocol. SOAP is the data encapsulation protocol that is used but the

transport protocol is fairly unlimited. Generally HTTP is the most common transport protocol used though you could conceivably use things like SMTP or any others. SOAP is not dependant on any single transport protocol or OS, it is a syntactical and logical definition, not a transport protocol.

True or False: A Web service can only be written in .NET.?

False.

What does WSDL stand for?

Web Services Description Language

Where on the Internet would you look for Web services?

UDDI repositories like uddi.microsoft.com, IBM UDDI node, UDDI Registries in Google Directory, enthusiast sites like XMethods.net.

What tags do you need to add within the asp:datagrid tags to bind columns manually?

Column tag and an ASP:databound tag.

How is a property designated as read-only?

In VB.NET:

```
Public ReadOnly Property PropertyName As Return Type
```

```
Get 'Your Property Implementation goes in here
```

```
End Get
```

```
End Property
```

in C#

```
public returntype PropertyName
```

```
{
```

```
get{
```

```
//property implementation goes here
```

```
}
```

```
// Do not write the set implementation
```

```
}
```

Which control would you use if you needed to make sure the values in two different controls matched?

Use the CompareValidator control to compare the values of 2 different controls.

True or False: To test a Web service you must create a windows application or Web application to consume this service?

False.

How many classes can a single .NET DLL contain?

Unlimited.

Describe session handling in a webfarm, how does it work and what are the limits?

Set the sessionState mode in the web.config file to "StateServer".

StateServer mode uses an out-of-process Windows NT Server to store state information.

It solves the session state loss problem in InProc mode.

Allows a webfarm to store session on a central server.

It provides a Single point of failure at the State Server.

Follow these simple steps:

- In a web farm, make sure you have the same in all your web servers.
- Also, make sure your objects are serializable.
- For session state to be maintained across different web servers in the web farm, the Application Path of the website in the IIS Metabase should be identical in all the web servers in the web farm.

What are the disadvantages of viewstate/what are the benefits?

Answer1:

Disadvantage of viewstate is that additional data is sent to the browser. The benefits are that you do not have to manually manage refreshing the page fields after a submit, (when re-displaying the same page).

Answer2:

Automatic view-state management is a feature of server controls that enables them to repopulate their property values on a round trip (without you having to write any code). This feature does impact performance, however, since a server control's view state is passed to and from the server in a hidden form field. You should be aware of when view state helps you and when it hinders your page's performance.

What tags do you need to add within the asp:datagrid tags to bind columns manually?

Answer1:

Set AutoGenerateColumns Property to false on the datagrid tag

Answer2:

tag and either or tags (with appropriate attributes of course)

What is State Management in .Net and how many ways are there to maintain a state in .Net?

What is view state?

Web pages are recreated each time the page is posted to the server. In traditional Web programming, this would ordinarily mean that all information associated with the page and the controls on the page would be lost with each round trip.

To overcome this inherent limitation of traditional Web programming, the ASP.NET page framework includes various options to help you preserve changes — that is, for managing state. The page framework includes a facility called view state that automatically preserves property values of the page and all the controls on it between round trips.

However, you will probably also have application-specific values that you want to preserve. To do so, you can use one of the state management options.

Client-Based State Management Options:

View State

Hidden Form Fields

Cookies

Query Strings

Server-Based State Management Options

Application State

Session State

Database Support

What tag do you use to add a hyperlink column to the DataGrid?

Depends on who's definition of hyperlink you're using. Manually a standard HTML anchor tag (a) will work or you can use the micro-magical tag

What is the standard you use to wrap up a call to a Web service?

Several possible answers depending on your interpretation of the question, but I think you were aiming for SOAP (with the caveat that this is MS's version of SOAP)

What is the difference between boxing and unboxing ?

Boxing allows us to convert value types to reference types. Basically, the runtime creates a temporary reference-type box for the object on heap.

Eg:

```
int i=20;  
object o=i;
```

Describe the difference between a Thread and a Process?

Answer1:

Thread - is used to execute more than one program at a time.
process - executes single program

Answer2:

A thread is a path of execution that runs on CPU, a process is a collection of threads that share the same virtual memory. A process has at least one thread of execution, and a thread always runs in a process context.

Answer3:

The operating system creates a process for the purpose of running a program. Each process executes a single program. Processes own resources allocated by the operating system. Resources include memory, file handles, sockets, device handles, and windows. Processes do not share address spaces or file resources except through explicit methods such as inheriting file handles or shared memory segments, or mapping the same file in a shared way.

Threads allow a program to do multiple things concurrently. At least one thread exists within each process. If multiple threads can exist within a process, then they share the same memory and file resources.

Answer4:

Thread is a light weight process, which is initialized itself by a process. Light weight processes do not load resources required by it itself, these are loaded by its parent process which has generated it.

What is a Windows Service and how does its lifecycle differ from a "standard" EXE?

Windows Service applications are long-running applications that are ideal for use in server environments. The applications do not have a user interface or produce any visual output; it is instead used by other programs or the system to perform operations. Any user messages are typically written to the Windows Event Log. Services can be automatically started when the computer is booted. This makes services ideal for use on a server or whenever you need long-running functionality that does not interfere with other users who are working on the same computer. They do not require a logged in user in order to execute and can run under the context of any user including the system. Windows Services are controlled through the Service Control Manager where they can be stopped, paused, and started as needed.

What is the difference between an EXE and a DLL?

An EXE can run independently, whereas DLL will run within an EXE. DLL is an in-process file and EXE is an out-process file

What is strong-typing versus weak-typing? Which is preferred? Why?

Strong type is checking the types of variables as soon as possible, usually at compile time. While weak typing is delaying checking the types of the system as late as possible, usually to run-time. Which is preferred depends on what you want. For scripts & quick stuff you'll usually want weak typing, because you want to write as much less code as possible. In big programs, strong typing can reduce errors at compile time.

What are PDBs? Where must they be located for debugging to work?

Answer1:

To debug precompiled components such as business objects and code-behind modules, you need to generate debug symbols. To do this, compile the components with the debug flags by using either Visual Studio .NET or a command line compiler such as Csc.exe (for Microsoft Visual C# .NET) or Vbc.exe (for Microsoft Visual Basic .NET).

Using Visual Studio .NET

1. Open the ASP.NET Web Application project in Visual Studio .NET.
2. Right-click the project in the Solution Explorer and click Properties.
3. In the Properties dialog box, click the Configuration Properties folder.
4. In the left pane, select Build.
5. Set Generate Debugging Information to true.
6. Close the Properties dialog box.
7. Right-click the project and click Build to compile the project and generate symbols (.pdb files).

Answer2:

A program database (PDB) file holds debugging and project state information that allows incremental linking of a Debug configuration of your program.

The linker creates project.PDB, which contains debug information for the project's EXE file. The project.PDB contains full debug information, including function prototypes, not just the type information found in VCx0.PDB. Both PDB files allow incremental updates.

They should be located at bin\Debug directory

What is cyclomatic complexity and why is it important?

Cyclomatic complexity is a computer science metric (measurement) developed by Thomas McCabe used to generally measure the complexity of a program. It directly measures the number of linearly independent paths through a program's source code.

The concept, although not the method, is somewhat similar to that of general text complexity measured by the Flesch-Kincaid Readability Test.

Cyclomatic complexity is computed using a graph that describes the control flow of the program. The nodes of the graph correspond to the commands of a program. A directed edge connects two nodes, if the second command might be executed immediately after the first command. By definition,

$$CC = E - N + P$$

where

CC = cyclomatic complexity

E = the number of edges of the graph

N = the number of nodes of the graph

P = the number of connected components.

What is FullTrust? Do GAC'ed assemblies have FullTrust?

Your code is allowed to do anything in the framework, meaning that all (.Net) permissions are granted. The GAC has FullTrust because it's on the local HD, and that has FullTrust by default, you can change that using caspol

What does this do? gacutil /l | find /i "about"

Answer1:

This command is used to install strong typed assembly in GAC

Answer2:

gacutil.exe is used to install strong typed assembly in GAC. gacutil.exe /l is used to lists the contents of the global assembly cache. |(pipe) symbol is used to filter the output with another command. find /i "about" is to find the text "about" on gacutil output. If any lines contains the text "about" then that line will get displayed on console window.

Contrast OOP and SOA. What are tenets of each

Service Oriented Architecture. In SOA you create an abstract layer that your applications use to access various "services" and can aggregate the services. These services could be databases, web services, message queues or other sources. The Service Layer provides a way to access these services that the applications do not need to know how the access is done. For example, to get a full customer record, I might need to get data from a SGL Server database, a web service and a message queue. The Service layer hides this from the calling application. All the application knows is that it asked for a full customer record. It doesn't know what system or systems it came from or how it was retrieved.

How does the XmlSerializer work? What ACL permissions does a process using it require?

XmlSerializer requires write permission to the system's TEMP directory.

Why is catch(Exception) almost always a bad idea?

Well, if at that point you know that an error has occurred, then why not write the proper code to handle that error instead of passing a new Exception object to the catch block? Throwing your own exceptions signifies some design flaws in the project.

What is the difference between Debug. Write and Trace. Write? When should each be used?

Answer1:

The Debug. Write call won't be compiled when the DEBUG symbol is not defined (when doing a release build). Trace. Write calls will be compiled. Debug. Write is for information you want only in debug builds, Trace. Write is for when you want it in release build as well. And in any case, you should use something like log4net because that is both faster and better

Answer2:

Debug. Write & Trace. write - both works in Debug mode, while in Release Mode,Trace.write only will work .Try changing the Active Config property of Solution in Property page and find the difference. Debug.write is used while debugging a project and Trace.write is used in Released version of

Applications.

What is the difference between a Debug and Release build? Is there a significant speed difference? Why or why not?

Debug build contain debug symbols and can be debugged while release build doesn't contain debug symbols, doesn't have [Conditional("DEBUG")] methods calls compiled, can't be debugged (easily, that is), less checking, etc. There should be a speed difference, because of disabling debug methods, reducing code size etc but that is not a guarantee (at least not a significant one)

Contrast the use of an abstract base class against an interface?

Answer1:

In the interface all methods must be abstract, in the abstract class some methods can be concrete. In the interface no accessibility modifiers are allowed, which is ok in abstract classes

Answer2:

Whether to Choose VB.NET/C#.

Both the languages are using same classes and namespaces. Once it compile and generates MSIL, there is no meaning of which language it was written. If you are Java/C++ programmer better to choose C# for same coding style otherwise you can choose VB.net.

What is the difference between a.Equals(b) and a == b?

Answer1:

a=b is used for assigning the values (rather than comparison) and a==b is for comparison.

Answer2:

a == b is used to compare the references of two objects

a.Equals(b) is used to compare two objects

Answer3:

A equals b -> copies contents of b to a

a == b -> checks if a is equal to b

Answer4:

Equals method compares both type and value of the variable, while == compares value.

```
int a = 0;
```

```
bool b = 0
```

```
if(a.Equals(b))
```

Answer5:

a.Equals(b) checks whether the Type of a is equal to b or not! Put it in another way,

```
Dim a As Integer = 1
```

```
Dim b As Single = 1
```

a.Equals(b) returns false. The Equals method returns a boolean value.

a == b is a simple assignment statement.

Answer6:

a.Equals(b) will check whether the “b” has same type as “a” has and also has the same data as “a” has.
a==b will do the same thing.

if you have done this in c++ under “operator overloading” than you guys must be aware of this syntaxs.
they are doing the same thing there is only syntactical difference.

let me explain it in different manner.

a==b : means compare “b” with “a”. always left hand side expression evaluated first so here in this case
“a” (considered an object) will call the overloaded operator “=” which defines “Equals(object)” method
in it’s class. thus, ultimately a.Equals(b) goanna called.

so the answer is: both will perform the same task. they are different by syntax

Answer7:

Difference b/w a==b,a.Equals(b)

a.Equals(b):

The default implementation of Equals supports reference equality only, but derived classes can override
this method to support value equality.

For reference types, equality is defined as object equality; that is, whether the references refer to the
same object. For value types, equality is defined as bitwise equality

== :

For predefined value types, the equality operator (==) returns true if the values of its operands are
equal, false otherwise. For reference types other than string, == returns true if its two operands refer to
the same object. For the string type, == compares the values of the strings.

How would one do a deep copy in .NET?

Answer1:

System.Array.CopyTo() - Deep copies an Array

Answer2:

How would one do a deep copy in .NET?

The First Approach.

1.Create a new instance.

2.Copy the properties from source instance to newly created instance.

[Use reflection if you want to write a common method to achive this]

The Second Approach.

1. Serialize the object and deserialize the output.

: Use binary serialization if you want private variables to be copied.

: Use xml Serialization if you dont want private variable to be copied.

What is boxing?

Boxing is an implicit conversion of a value type to the type object

```
int i = 123; // A value type
```

```
Object box = i // Boxing
```

Unboxing is an explicit conversion from the type object to a value type

```
int i = 123; // A value type object box = i; // Boxing
```

```
int j = (int)box; // Unboxing
```

Is string a value type or a reference type?

Answer1:

String is Reference Type.

Value type - bool, byte, char, decimal, double, enum, float, int, long, sbyte, short, ushort, ushort

Value types are stored in the Stack

Reference type - class, delegate, interface, object, string

Reference types are stored in the Heap

Answer2:

Yes String is reference type. C# gives two types of variable reference and value type. string and object are reference type.

How does the lifecycle of Windows services differ from Standard EXE?

Windows services lifecycle is managed by "Service Control Manager" which is responsible for starting and stopping the service and the applications do not have a user interface or produce any visual output, but "Standard executable" doesn't require Control Manager and is directly related to the visual output

What's wrong with a line like this? DateTime.Parse(myString)

the result returned by this function is not assigned to anything, should be something like var x = DateTime.Parse(myString)

NET is Compile Time OR RunTime Environment?

.Net's framework has CLS,CTS and CLR.CTS checks declaration of types at the time when u write code and CLS defines some rules and restrictions.and CLR compile everything at runtime with following benefits: Vastly simplified development Seamless integration of code written in various languages Evidence-based security with code identity Assembly-based deployment that eliminates DLL Hell Side-by-side versioning of reusable components Code reuse through implementation inheritance Automatic object lifetime management Self describing objects

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

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

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

The latter one allows you to write formatted output.

What methods are fired during the page load?

Init() - when the page is

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

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

- when page finishes loading.

Where does the Web page belong in the .NET Framework class hierarchy?

System.Web.UI.Page

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

System.Web.UI.Page.Culture

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

CodeBehind is relevant to Visual Studio.NET only.

What's a bubbled event?

When you have a complex control, like DataGrid, writing an event processing routine for each object (cell, button, row, etc.) is quite tedious. The controls can bubble up their event handlers, allowing the main DataGrid event handler to take care of its constituents.

Suppose you want a certain ASP.NET function executed on MouseOver over a certain button.

Where do you add an event handler?

It's the Attributes property, the Add function inside that property. So

```
btnSubmit.Attributes.Add("onMouseOver","someClientCode();")
```

A simple "Javascript:ClientCode();" in the button control of the .aspx page will attach the handler (javascript function) to the onmouseover event.

What data type does the RangeValidator control support?

Integer,String and Date.

To test a Web Service you must create a windows application or web application to consume this service? It is True/False?

FALSE

How many classes can a single.NET DLL contain?

Answer1:

As many

Answer2:

One or more

What are good ADO.NET object(s) to replace the ADO Recordset object?

The differences includes

In ADO, the in-memory representation of data is the Recordset.

In ADO.net, it is the dataset

A recordset looks like a single table in ADO

In contrast, a dataset is a collection of one or more tables in ADO.net

ADO is designed primarily for connected access

ADO.net the disconnected access to the database is used

In ADO you communicate with the database by making calls to an OLE DB provider.

In ADO.NET you communicate with the database through a data adapter (an OleDbDataAdapter, SqlDataAdapter, OdbcDataAdapter, or OracleDataAdapter object), which makes calls to an OLE DB provider or the APIs provided by the underlying data source.

In ADO you can't update the database from the recordset. ADO.NET the data adapter allows you to control how the changes to the dataset are transmitted to the database.

On order to get assembly info which namespace we should import?

System.Reflection Namespace

How do you declare a static variable and what is its lifetime? Give an example.

Answer1

static int Myint–The life time is during the entire application.

br> Answer2

The static modifier is used to declare a static member, which belongs to the type itself rather than to a specific object. The static modifier can be used with fields, methods, properties, operators, events and constructors, but cannot be used with indexers, destructors, or types. In C#, the static keyword indicates a class variable. In VB, the equivalent keyword is Shared. Its scoped to the class in which it occurs.

Example

a. Static int var //in c#.net

b. static void Time() //in c#.net

How do you get records number from 5 to 15 in a dataset of 100 records? Write code.

Answer1

```
DataSet ds1=new DataSet(); String strCon="data source=IBM-6BC8A0DACEF;initial catalog=pubs;integrated security=SSPI;persist" +" security info=False;user id=sa;workstation id=IBM-6BC8A0DACEF;packet size=4096?;
```

```
String strCom1="SELECT * FROM employee";  
SqlDataAdapter sqlDa1=new SqlDataAdapter(strCom1,strCon);  
ds1.Tables.Add("employee");  
sqlDa1.Fill(ds1,40,50,ds1.Tables["employee"].TableName);  
DataGrid dg1.DataSource=ds1.Tables["employee"].DefaultView;  
dg1.DataBind();
```

Answer2

```
OleDbConnection1.Open()  
OleDbDataAdapter1.Fill(DataSet21, 5, 15, "tab")  
This will fill the dataset with the records starting at 5 to 15
```

How do you call and execute a Stored Procedure in .NET? Give an example.

Answer1

```
ds1=new DataSet();
```

```
sqlCon1=new SqlConnection(connectionstring);
```

```
String strCom1="byroyalty";
```

```
sqlCom1=new SqlCommand(strCom1,sqlCon1);  
sqlCom1.CommandType=CommandType.StoredProcedure;  
sqlDa1=new SqlDataAdapter(sqlCom1);  
SqlParameter myPar=new SqlParameter("@percentage",SqlDbType.Int);  
sqlCom1.Parameters.Add (myPar);
```

```
myPar.Value=40;
sqlDa1.Fill(ds1);
dg1.DataSource=ds1;
dg1.DataBind();
```

Answer2

Yes

```
Dim cn as new OleDbConnection ( "Provider=Microsoft.Jet.OLEDB.4.0;" + _
"Data Source=C:\Documents and Settings\User\My Documents\Visual Studio
Projects\1209\db1.mdb"+ _
"User ID=Admin;" + _
"Password=;");
Dim cmd As New OleDbCommand("Products", cn)
cmd.CommandType = CommandType.StoredProcedure
```

```
Dim da As New OleDbDataAdapter(cmd)
Dim ds As New DataSet()
da.Fill(ds, "Products")
DataGrid1.DataSource = ds.Tables("Products")
```

What is the maximum length of a varchar in SQL Server?

Answer1

VARCHAR[(n)]

Null-terminated Unicode character string of length n,
with a maximum of 255 characters. If n is not supplied, then 1 is assumed.

Answer2

8000

Answer3

The business logic is the aspx.cs or the aspx.vb where the code is being written. The presentation logic is done with .aspx extension.

How do you define an integer in SQL Server?

We define integer in Sql server as

```
var_name int
```

How do you separate business logic while creating an ASP.NET application?

There are two level of asp.net debugging

1. Page level debugging

For this we have to edit the page level debugging enable the trace to true in the line in the html format of the page.

```
%@ Page Language="vb" trace="true"AutoEventWireup="false" Codebehind="WebForm1.aspx.vb"
Inherits="WebApplication2.WebForm1?>
```

2. You can enable the debugging in the application level for this

Edit the following trace value in web.config file

Enable trace enabled=true.

If there is a calendar control to be included in each page of your application, and we do not intend to use the Microsoft-provided calendar control, how do you develop it? Do you copy and paste the code into each and every page of your application?

Create the Calendar User Control

The control we will create will contain a calendar control and a label which has the corresponding date and time written

Steps are:-

Creating a CalenderControl

- 1) To begin, open Visual Studio .NET and begin a new C# Windows Control Library.
- 2) You may name it whatever you like, for this sample the project name will be CalenderControl

Using the Calender Control in a Windows Application

It's just like adding any other control like a button or a label.

- 1) First, create a new Windows Application project named: CustomControl.
- 2) Add a reference to the Calender Control DLL named: CalenderControl.dll.
- 3) Now you can customize the Toolbox:
Right-Click the Toolbox> .NET Framework Components> Browse> select the CalenderControl.dll.
- 4) The Calender Control is now added to the Toolbox and can be inserted in Windows Form as any other control. The control itself will take care of the date display

How can you deploy an asp.net application ?

You can deploy an ASP.NET Web application using any one of the following three deployment options.

- a) Deployment using VS.NET installer
- b) Using the Copy Project option in VS .NET
- c) XCOPY Deployment

Explain similarities and differences between Java and .NET?

Comparing Java and .NET is comparing apples and oranges. Either the question needs to be to compare Java and C# or J2EE and .NET.

What are the XML files that are important in developing an ASP.NET application?

The XML file necessary for the for developing an asp.net application is Web.config

Specify the best ways to store variables so that we can access them in various pages of ASP.NET application?

Declare the variables in Global.aspx

How many objects are there in ASP?

Answer1

8 objects, they are request,response, server,application,session,file, dictionary, textstream.

Answer2

There are 6 objects in ASP.net

- a) Server

- b) Session
- c) Application
- d) ObjectContext
- e) Response
- f) Request

Which DLL file is needed to be registered for ASP?

The dll needed for the ASP.net is SYSTEM.WEB.dll

Is there any inbuilt paging (for example shopping cart, which will show next 10 records without refreshing) in ASP? How will you do paging?

Use DataGrid control which has in-built paging features for the purpose.

What does Server.MapPath do?

Answer1

server.mappath() maps the path given in the argument to the server's physical path.

Answer2

It returns the complete(absolute) path of the file used in parameter.

Answer3

It returns a string containing the physical path in the server's file system that corresponds to the virtual or relative path specified by the Path argument.

Name atleast three methods of response object other than Redirect.

Answer1

a) Response.Clear()

Clears the content of the current output stream.

b) Response.Close()

Closes the network socket for the current response.

c) Response.End()

Stops processing the current request and sends all buffered content to the client immediately.

Answer2

methods of Response is Redirect a. Transfer

Name atleast two methods of response object other than Transfer.

a) Response.ClearContent()

Clears the content of the current output stream.

b) Response.ClearHeaders()

Clears the HTTP headers from the current output stream.

What is State?

It is the property of the web forms.

ASP.NET provides four types of state:

Application state

Session state

Cookie state
View state.

Explain differences between ADO and DAO.

dao- can access only access database
ado- can access any databases

How many types of cookies are there?

2 types, persistent and non-persistent.

How many types of cookies are there?

Answer1

Two types of cookies.

- a) single valued eg request.cookies("UserName").value="Mahesh"
- b) Multivalued cookies. These are used in the way collections are used.

e.g.

```
request.cookies("CookieName")("UserName")="Mahesh"
```

```
request.cookies("CookieName")("UserID")="ABC003"
```

remember no value method in multivalued cookie

Answer2

There are two types of cookies:

Session cookies

Persistent cookies

Tell few steps for optimizing (for speed and resource) ASP page/application.

Avoid mixing html code with asp code

Which command using Query Analyzer will give you the version of SQL Server and Operating System?

@@VERSION

Returns version, processor architecture, build date, and operating system for the current installation of SQL Server.

How to find the SQL server version from Query Analyzer ?

Answer1

To determine which version of Microsoft SQL Server 2005 is running, connect to SQL Server 2005 by using SQL Server Management Studio, and then run the following Transact-SQL statement:

```
SELECT SERVERPROPERTY('productversion'), SERVERPROPERTY ('productlevel'),  
SERVERPROPERTY ('edition')
```

The results are:

- The product version (for example, "9.00.1399.06?")
- The product level (for example, "RTM").
- The edition (for example, "Enterprise Edition").

For example, the result looks similar to:

9.00.1399.06 RTM Enterprise Edition

How to determine which version of SQL Server 2000 is running

To determine which version of SQL Server 2000 is running, connect to SQL Server 2000 by using Query Analyzer, and then run the following code:

```
SELECT SERVERPROPERTY('productversion'), SERVERPROPERTY ('productlevel'),  
SERVERPROPERTY ('edition')
```

The results are:

- The product version (for example, 8.00.534).
- The product level (for example, "RTM" or "SP2?").
- The edition (for example, "Standard Edition"). For example, the result looks similar to

:

8.00.534 RTM Standard Edition

Answer2

One can also use SELECT @@Version where the result would look like

Microsoft SQL Server 2005 - 9.00.1399.06 (Intel X86)

Oct 14 2005 00:33:37

Copyright (c) 1988-2005 Microsoft Corporation

Express Edition on Windows NT 5.1 (Build 2600: Service Pack 2)

Using query analyzer, name 3 ways you can get an accurate count of the number of records in a table.

Answer1.

a. Select count(*) from table1

b. SELECT object_name(id),rowcnt FROM sysindexes WHERE indid IN (1,0) AND
OBJECTPROPERTY(id, 'IsUserTable') = 1

c. exec sp_table_validation @table = 'authors'

Answer2.

```
SELECT count( * ) as totalrecords FROM employee
```

This will display total records under the name totalrecords in the table employee

use COUNT_BIG

Returns the number of items in a group.

@@ROWCOUNT

Returns the number of rows affected by the last statement.

Use this statement after an SQL select * statement, to retrieve the total number of rows in the table

What is the purpose of using COLLATE in a query?

Answer1.

Collation refers to a set of rules that determine how data is sorted and compared. Character data is sorted using rules that define the correct character sequence, with options for specifying case-sensitivity, accent marks, kana character types and character width.

Answer2.

COLLATE is a clause that can be applied to a database definition or a column definition to define the

collation, or to a character string expression to apply a collation cast.

What is one of the first things you would do to increase performance of a query? For example, a boss tells you that “a query that ran yesterday took 30 seconds, but today it takes 6 minutes”?

Answer1.

Use Storedprocedure for any optimized result, because it is an compiled code.

Answer2.

One of the best ways to increase query performance is to use indexes.

What is an execution plan? When would you use it? How would you view the execution plan?

The Query Analyzer has a feature called Show Execution Plan. This option allows you to view the execution plan used by SQL Server's Query Optimizer to actually execute the query. This option is available from the Query menu on the main menu of Query Analyzer, and must be turned on before the query is executed. Once the query is executed, the results of the execution plan are displayed in graphical format in a separate window, available from a tab that appears below the query results window on the screen.

What is the STUFF function and how does it differ from the REPLACE function? Answer1:

stuff-> inserts into it without removing any thing. Replace->replace the given text with the new one.

Answer2:

STUFF - it deletes a specified length of characters and inserts another set of characters at a specified starting point. REPLACE -Replaces all occurrences of a specified string value with another string value.

What does it mean to have quoted _identifier on? What are the implications of having it off?

SET QUOTED_IDENTIFIER ON- Causes SQL Server to follow the SQL-92 rules regarding quotation mark delimiting identifiers and literal strings. Identifiers delimited by double quotation marks can be either Transact-SQL reserved keywords or can contain characters not usually allowed by the Transact-SQL syntax rules for identifiers.

What is the difference between a Local temporary table and a Global temporary table? How is each one used?

Answer1:

Local temprary table will have a single # (#tablename) appended with the table name.Global temprary table will have Double # (##tablename) appended with the table name.

Ex:create table #table1

local temp. table will be available until the session who created it logs out, but global temp. table is available till the last session gets close in SQLServer.

Answer1:

Local temporary tables are visible only in the current session; global temporary tables are visible to all sessions.Prefix local temporary table names with single number sign (#table_name), and prefix global temporary table names with a double number sign (##table_name).

What are cursors? Name four type of cursors and when each one would be applied?

Opening a cursor on a result set allows processing the result set one row at a time.

The four API server cursor types supported by SQL Server are:

a) Static cursors

- b) Dynamic cursors
- c) Forward-only cursors
- d) Keyset-driven cursors

What is the purpose of UPDATE STATISTICS?

UPDATE STATISTICS- it updates information about the distribution of key values for one or more statistics groups (collections) in the specified table or indexed view.

How do you use DBCC statements to monitor various ASpects of a SQL Server installation?

Database Consistency Checker (DBCC) - Is a statement used to check the logical and physical consistency of a database, check memory usage, decrease the size of a database, check performance statistics, and so on. Database consistency checker (DBCC) ensures the physical and logical consistency of a database, but is not corrective. DBCC can help in repairing or checking the installation in case of any failure.

What is referential integrity and how can we achieve it?

Referential integrity preserves the defined relationships between tables when records are entered or deleted. In SQL Server, referential integrity is based on relationships between foreign keys and primary keys or between foreign keys and unique keys. Referential integrity ensures that key values are consistent across tables. Such consistency requires that there be no references to nonexistent values and that if a key value changes, all references to it change consistently throughout the database. We can achieve this by using foreign key.

What is indexing?

If we give proper indexes on a table so that any queries written against this table can run efficiently. As your data sets grow over time, SQL Server will continue to rebuild indexes and move data around as efficiently as possible. This property is known as Indexing.

Explain differences between Server.Transfer and server.execute method?

Answer1:

server.transfer-> transfers the server's control to the requested page given in the parameter.

server.Execute-> executes the requested page from the current page itself, with no change in the address bar. after execution the next line of code is executed in the current page.

Answer2.

Execute method returns control to the page in which it is called once the page specified in the Execute method finishes processing, the Transfer method does not return control to the calling page.

What is de-normalization? When do you do it and how?

De-normalization is the process of attempting to optimize the performance of a database by adding redundant data. It's used To introduce redundancy into a table in order to incorporate data from a related table. The related table can then be eliminated. De-normalization can improve efficiency and performance by reducing complexity in a data warehouse schema.

Explain features of SQL Server like Scalability , Availability, Integration with Internet.

Scalability - The same Microsoft SQL Server 2000 database engine operates on Microsoft Windows 2000 Professional, Microsoft Windows 2000 Server, Microsoft Windows 2000 Advanced Server, Windows 98, and Windows Millennium Edition. It also runs on all editions of Microsoft Windows NT version 4.0. The database engine is a robust server that can manage terabyte-sized databases accessed by thousands of users. Availability - SQL Server 2000 can maintain the extremely high levels of

availability required by large Web sites and enterprise systems. Integration -The SQL Server 2000 TCP/IP Sockets communications support can be integrated with Microsoft Proxy Server to implement secure Internet and intranet communications.

What is Data Warehousing?

A data warehouse is a collection of data gathered and organized so that it can easily be analyzed, extracted, synthesized, and otherwise be used for the purposes of further understanding the data.

What is OLAP?

OLAP is an acronym for On Line Analytical Processing. It is an approach to quickly provide the answer to analytical queries that are dimensional in nature.

How do we upgrade SQL Server 7.0 to 2000?

Run the installation of the SQL Server 2000

In the Existing Installation dialog box, click Upgrade your existing installation, and then click Next. In the Upgrade dialog box, you are prompted as to whether you want to proceed with the requested upgrade. Click Yes, upgrade my to start the upgrade process, and then click Next. The upgrade runs until finished.

In the Connect to Server dialog box, select an authentication mode, and then click Next.

If you are not sure which mode to use, accept the default: The Windows account information I use to log on to my computer with (Windows). In Start Copying Files dialog box, click Next.

Now your Sql Server would be upgraded.

What is job?

It can be defined as a task performed by a computer system. For example, printing a file is a job. Jobs can be performed by a single program or by a collection of programs.

What is Task?

Whenever you execute a program, the operating system creates a new task for it. The task is like an envelope for the program: it identifies the program with a task number and attaches other bookkeeping information to it.

How do you find the error, how can you know the number of rows affected by last SQL Statement?

Answer1

@@errors->give the last error occurred in the current DB.

Ans. select @@rowcount

Answer2.

Use @@ERROR which returns the error number for the last Transact-SQL statement executed from knowing the error.

Use @@ROWCOUNT which returns the number of rows affected by the last statement for finding the no of rows affected.

What are the advantages/disadvantages of viewstate?

Disadvantages - Because the view state for a given page must be kept on the server, it is possible for the current state to be out of synchronization with the current page of the browser, if the user uses the Back feature on the browser to go back in the history. Advantages - On ordinary Web Forms pages, their view state is sent by the server as a hidden variable in a form, as part of every response to the client, and is returned to the server by the client as part of a postback. However, to reduce bandwidth

demand when using mobile controls, ASP.NET does not send a page's view state to the client. Instead, the view state is saved as part of a user's session on the server. Where there is a view state, a hidden field that identifies this page's view state is sent by the server as part of every response to the client, and is returned to the server by the client as part of the next request.

Describe session handling in webform. How does it work and what are the limits?

Session management in ASP.NET can be done in two ways:

Using Cookies

Encoding of URLs with Session ID

Explain differences between framework 1.0 and framework 1.1?

1. Native Support for Developing Mobile Web Applications
2. Unified Programming Model for Smart Client Application Development
3. Enable Code Access Security for ASP.NET Applications
4. Native Support for Communicating with ODBC and Oracle Databases
5. Supports for IPv6

If we write any code for dataGrid methods, what is the access specifier used for that methods in the code behind file and why and how? Give an example.

We use Friends Modifier for the dataGrid methods. Friend WithEvents DataGrid1 As System.Windows.Forms.DataGrid

What is the use of trace utility?

Tracing is a very important monitoring and debugging tool for distributed, multitier applications. Such applications often contain problems that can only be observed when the application is under a heavy load and the inherent randomness of a real-life environment. Trace utility allows developers and administrators to monitor the health of applications running in real-life settings.

What are the differences between User control and Web control and Custom control?

Answer1:

Usercontrol-> control that is created as u wish.

Web Control-> any control placed in web page (web application page)

Custom Control-> same as user control with some difference.

user control custome control

1.easy to create difficult

2.no full suport for customers using

Visual studio tools Full support

3. Seperate copy of the control in each

assembly only one copy in global assembly.

4. best for static layout best for dynamic layout.

Answer2

User control

1) Reusability web page

2) We can't add to toolbox

3) Just drag and drop from solution explorer to page (aspx)

4) Good for static layout

- 5) Easier to create
- 6) Not compiled into DLL

Custom controls

- 1) Reusability of control (or extend functionalities of existing control)
- 2) We can add toolbox
- 3) Just drag and drop from toolbox
- 4) You can register user control to. Aspx page by Register tag
- 5) A single copy of the control is required in each application
- 6) Good for dynamic layout
- 7) Hard to create
- 8) Compiled in to dll

Custom controls

- 1) Reusability of control
- 2) Pre defined Control
- 3) Just drag and drop from toolbox

If I have more than one version of one assemblies, then how will I use old version in my application? Give an example.

Change the assembly version number in the AssemblyInfo.vb file

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

Answer1:

add the component in add reference window, click .NETCOM tab.

Answer1:

While adding the refferences we can handle the COM components in other .Net programming languages.

How will you register COM+ services?

Through X-Copy Deployment.

How do u call and execute a stored procedure in .NET?

```
system.Data;  
system.Data.SqlClient;
```

```
SqlConnection sqCon = new SqlConnection("connection string");  
SqlCommand sqCmd = new SqlCommand();  
sqCmd.Connection = sqCon;  
sqCmd.CommandText = procedure_name;  
sqCmd.CommandType = CommandType.StoredProcedure;  
sqComd.ExecuteReader();
```

What are the different types of replication? How are they used?

Replication is used for distributing data and the execution of stored procedures across an enterprise. The replication technology allows you to make duplicate copies of your data, move those copies to different locations, and synchronize the data automatically so that all copies have the same data values.

The different types of replications are

- a) transactional replication
- b) merge replication

How do SQL Server 2000 and XML linked? What is SQL Server agent?

Every Request or the Response to or from SQL Server is converted into XML format. Its purpose is to ease the implementation of tasks for the DBA, with its full-function scheduling engine, which allows you to schedule your own jobs and scripts.

How do you create thread in .NET?

- 1) Import System.Threading
- 2) Create a new thread using new Thread() and assign the address of the method
- 3) Use Thread.Start method to start the execution

```
using System;  
using System.Threading;
```

```
public class Test  
{  
    static void Main()  
    {  
        ThreadStart job = new ThreadStart(ThreadJob);  
        Thread thread = new Thread(job);  
        thread.Start();  
    }  
}
```

Which of the following languages is NOT included in the default .NET Framework installation?

- * C#
- * VB.NET
- * JScript.NET
- * VBScript.NET

VBScript.NET

What are the different types of serialization supported in .NET Framework

- * XmlSerializer
- * SoapFormatter
- * XPathNavigator
- * HttpFormatter

xmlserializer

The CLR uses which format for assembly version numbers

- * Major:Minor:Revision:Build
- * Major:Build:Minor:Revision
- * Major:Revision:Minor:Build
- * Major:Minor:Build:Revision

Major:Minor:Build:Revision

What tool is used to manage the GAC?

- * **GacMgr.exe**
- * **GacSvr32.exe**
- * **GacUtil.exe**
- * **RegSvr.exe**

GacUtil.exe

State True or False: A single .NET dll can contain unlimited classes

- * **True**
- * **False**

True

State True or False: ASP.NET can currently run only on Windows Platform

- * **True**
- * **False**

True

Which one of the following best describes “Type-Safe”

- * **It ensures that the data type are safely handled**
- * **It ensures that types are not mismatched when they are called so uses a typecasting before referencing any data object**
- * **It ensures that an object only references memory locations that it’s allowed to, preventing data corruption and the accidental misuse of object types**
- * **All of the above**

All of the above

The number of objects in ASP.NET is

- * **6**
- * **7**
- * **9**
- * **10**

Answer1:

7

Answer2:

10

The code used to turn off buffering is

- * **Buffering = false**
- * **OutputBuffer = false**
- * **BufferOutput = false**
- * **Buffer = Off**

Answer1:

Buffer=true

Answer2:

Buffer=false

Can you have two applications on the same machine one which is using .NET Framework 1.1 and the other using 2.0 ?

- * Yes
- * No
- * Depends on System configuration

Yes

Which of the following DOT.NET tools manages certificates, certificate trust lists (CTLs), and certificate revocation lists (CRLs)?

- * sn.exe
- * certnet.exe
- * certmgr.exe
- * gacutil.exe

certmgr.exe

You need to generate a public/private key pair for using in creating a shared assembly. Given the above scenario, which .NET SDK utility should be used?

- * certmgr.exe
- * gacutil.exe
- * sn.exe
- * resgen.exe

sn.exe

The object that contains all the properties and methods for every ASP.NET page, that is built is

- * Page Object
- * HTTPPage Object
- * WebPage Object
- * System.Web.UI.Page

Page Object

In C#, which character is used to indicate a verbatim string literal?

- * @
- * !
- * "
- * #

@

Which of the following operators has the highest precedence?

- * Pre Increment (++x)
- * Shift bits left: <<
- * Bitwise Or: |

* **Post Increment (x++)**

Shift bits left

The uniqueId that gets generated at the start of the Session is stored in

* **Client computer as a cookie**

* **Server machine**

* **Passed to and fro on each and every request and response**

* **Both a and b are correct**

Both a and b are correct

State True or False: C# supports multiple-inheritance

* **True**

* **False**

False

Bitwise AND operator in C# is

* **&**

* **&&**

* **AND**

* **XAND**

Answer1:

&&

Answer2:

&

Bitwise OR operator in C# is

* **OR**

* **||**

* **|**

* **XOR**

Answer1:

||

Answer2:

|

What's the .NET datatype that allows the retrieval of data by a unique key?

* **Primary**

* **Integer**

* **Unique Identifier**

* **HashTable**

HashTable

The keyword 'int' maps to one of the following .NET types

- * **System.Int16**
- * **System.Int32**
- * **System.Int64**
- * **System.Int128**

System.Int32

What can be achieved in IL which is not possible in C# ?

- * **Creation of an ArrayList**
- * **Creation of a Dictionary**
- * **Creation of a two dimensional array**
- * **Creation of a non-zero based array**

Answer1:

Creation of a dictionary

Answer2:

Creation of a non-zero based array

Which of the following is the correct code for setting a Session timeout of 30 minutes

- * **Session.Timeout = 108000**
- * **Session.Timeout = 1800**
- * **Session.Timeout = 300**
- * **Session.Timeout = 30**

Answer1:

Session.Timeout = 1800

Answer2:

Session.Timeout = 30

The process that ASP.NET uses to keep track of Sessions without cookies is

- * **Cookie Munging**
- * **Cookie Monking**
- * **Cookie Mocking**
- * **Cookie Munching**

Cookie munging

The method that transfers ASP.NET execution to another page, but returns to the original page when it is done is

- * **Server.Transfer()**
- * **Server.Redirect()**
- * **Server.Execute()**
- * **Server.Return()**

Answer1:

Server.Redirect()

Answer2:

Server.Execute()

A structure in C# can be derived from one or more

- * **class**
- * **interface**
- * **both**
- * **none**

interface.

State True or False: Static method cannot be overridden

- * **True**
- * **False**

True

The Equivalent HTML Control for the <input type="button"> tag is

- * **HtmlInput**
- * **HtmlButton**
- * **HtmlPushButton**
- * **HtmlInputButton**

HtmlInputButton

The Equivalent Html Control for the <input type="checkbox"> tag is

- * **HtmlCheckBox**
- * **HtmlInputChkBox**
- * **HtmlInputCheckBox**
- * **HtmlInputTypeChkBox**

HtmlInputCheckBox

Which operator is used for connecting a event with a procedure in C#?

- * **+=**
- * **=**
- * **both**
- * **none**

+=

The Equivalent Html Control for the <select> tag is

- * **<HtmlSelectInput>**
- * **<HtmlSelect>**
- * **<HtmlInputSelect>**
- * **<HtmlSelectControl>**

<HtmlSelect>

State True or False: Events in Web forms are processed before the “Page Load” event

- * True
- * False

True

What namespaces are necessary to create a localized application?

- * System.Globalization
- * System.Globalization and System.Resources.
- * System.Resources.
- * System.Array

System.Resources

A new server-side control can be created by implementing the class

- * System.Web.WebControl
- * System.Web.UI.WebControl
- * System.Web.UI.WebControls.WebControl
- * Any one of the above

System.Web.UI.WebControls.WebControl

The parameter “clienttarget = downlevel” does one of the following

- * Adds aliases for specific user agents to an internal collection of user agent aliases
- * Indicates the useragents level of validating the controls
- * Disables all DHTML events(including Validation controls)
- * None of the above

None of the above

The methods in C# can be overloaded in which of the following ways

- * By having the same method name and specifying different number of parameters
- * By giving different method names and same number of parameters
- * By having the same method name and specifying different types of parameters
- * By giving different method names and same types of parameters

By having the same method name and specifying different types of parameters

The RangeValidator control supports the following datatype

- * Integer and String
- * Integer, Float, String, XMLDatatypes
- * Integer, String and Date
- * Integer, Boolean, Short, String and Date

Integer, String and Date

What is the difference between Convert.ToInt32 and int.Parse?

- * Both are same and both can handle null value.
- * Convert.ToInt32 can handle null value but int.Parse throws ArgumentException error.
- * int.Parse can handle null value but Convert.ToInt32 throws ArgumentException error.

*** Both Convert.ToInt32 and int.Parse cannot handle null value.**

Convert.ToInt32 can handle null value but int.Parse throws ArgumentNullException error.

State True or False: Any ODBC-compliant database can be accessed through ASP.NET

*** True**

*** False**

True

You need to select a .NET language that has auto-documenting features built into the source code and compiler. Given the scenario above what language compiler should be selected?

*** vbc.exe**

*** cl.exe**

*** ilasm.exe**

*** cs.exe**

cs.exe

A set of tables are maintained in a Dataset as

*** TablesCollection object**

*** DataTableCollection object**

*** DataRowsCollection object**

*** TableRowCollection object**

TablesCollection object

The namespaces needed to use data mechanisms in ASP.NET pages are

*** System.Data, System.Data.OleDb or System.Data.SQL**

*** System.Data, System.Data.ODBC**

*** System.Data, System.Data.DataRow**

*** System.Data.DataRowCollection, System.Data.DbType**

System.Data, System.Data.OleDb or System.Data.SQL

What are the different methods to access Database in .NET ?

*** OleDb,ODBC,SQLClient**

*** OleDb, JDBC, SQLServer**

*** ODBC-JDBC, DataSet, SQLClient**

*** Datasource, DataSet, DSN**

OleDb,ODBC,SQLClient

The two properties of a DataGrid that has to be specified to turn on sorting and paging respectively are

*** EnableSorting = "true" and EnablePaging = "true"**

*** DisableSorting = "false" and DisablePaging = "false"**

*** AllowSorting = "true" and AllowPaging = "true"**

*** Sorting = "true" and Paging = "true"**

AllowSorting = "true" and AllowPaging = "true"

Which one of the following objects is used to create a foreign key between two DataTables?

- * **DataRelation**
- * **DataRelationship**
- * **DataConstraint**
- * **DataKey**

DataRelation

The Syntax for data-binding expressions is

- * **<%# property or collection %>**
- * **<%# property or collection #%>**
- * **<%\$ property or collection %>**
- * **<%@ property or collection %>**

<%# property or collection #%>

The method that need to be invoked on the DataAdapter control to load the generated dataset with data is

- * **Bind()**
- * **Fill()**
- * **FillData()**
- * **SetData()**

Fill()

Which of the following operations can you NOT perform on an ADO.NET DataSet?

- * **A DataSet can be synchronised with the database.**
- * **A DataSet can be synchronised with a RecordSet.**
- * **A DataSet can be converted to XML.**
- * **You can infer the schema from a DataSet**

A DataSet can be synchronised with a RecordSet.

Which is the correct statement to set the alias name for namespace in C#?

- * **using System Data.OracleClient = aliasName;**
- * **using aliasName = System.Data.OracleClient;**
- * **string aliasName = using.System.Data.OracleClient;**
- * **Both a & b**

using aliasName = System.Data.OracleClient;

The property that indicates whether existing database constraints should be observed when performing updates

- * **EnforceConstraints**
- * **Constraints**
- * **GetConstraints**
- * **ConstraintsEnforce**

EnforceConstraints

State True or False: If you set AutoGenerateColumns=True and still provide custom column definitions, the DataGrid will render both

- * True
- * False

False.

The data from an XSL Transform with XmlReader can be returned in one of the following ways

- * objReader = objXslt.Transform(objNav, nothing)
- * objXslt.Transform(objNav, nothing)
- * objReader = objXslt.Transform(objNav, nothing, objWriter)
- * objXslt.Transform(objNav, nothing, objWriter)

objXslt.Transform(objNav, nothing)

Pick the command line that would result in the C# compiler generating an XML documentation file

- * csc /doc:NewHome.xml NewHome.cs
- * c /doc /docfile: NewHome.xml NewHome.cs
- * csc /doc /out: NewHome.xml NewHome.cs
- * csc /xml NewHome.cs

csc /doc:NewHome.xml NewHome.cs

What is the comment syntax for C#'s XML-based documentation? * / and **/ * /// * /// * /****

///

When creating a C# Class Library project, what is the name of the supplementary file that Visual Studio.NET creates that contains General Information about the assembly?

- * AssemblyInfo.xml
- * AssemblyInfo.cs
- * AssemblyInformation.cs
- * AssemblyAttributes.cs

AssemblyInfo.cs

Which of the following is the C# escape character for Null?

- * \n
- * \0
- * \f
- * \v

\0

What is the exception that is thrown when there is an attempt to dynamically access a method that does not exist?

- * MissingMethodException

- * **TypeLoadException**
- * **MethodLoadException**
- * **MethodAccessException**

MissingMethodException

What method(s) must be used with the Application object to ensure that only one process accesses a variable at a time?

- * **Synchronize()**
- * **Lock() and UnLock()**
- * **Lock()**
- * **Asynchoize()**

Lock()

After capturing the SelectedIndexChanged event for a ListBox control, you find that the event handler doesn't execute. What could the problem be?

- * **The AutoEventWireup attribute is set to False**
- * **The AutomaticPostBack attribute is set to False**
- * **The codebehind module is not properly compiled**
- * **The ListBox must be defined WithEvents**

AutoPostBack attribute is set to False

What method must be overridden in a custom control?

- * **The Paint() method**
- * **The Control_Build() method**
- * **The Render() method**
- * **The default constructor**

The Render() method

What is used to validate complex string patterns like an e-mail address?

- * **Extended expressions**
- * **Regular expressions**
- * **Irregular expressions**
- * **Basic expressions**

Regular Expression.

The following is a valid statement in ASP.NET <%@ Page Language="C" %>

- * **True**
- * **False**

False

A valid comment block in ASP.NET is

- * **<!-- - - Comment - - - >**
- * **<!-- - Comment - - >**
- * **<% - - Comment - - %>**

* <% ! -- Comment -- >

<% -- Comment -- %>

The event handlers that can be included in the Global.asax file are

- * **Application Start and Session Start event handlers only**
- * **Application End and Session End event handlers only**
- * **Per-request and Non-deterministic event handlers only**
- * **Application Start and End ,**

Application Start and End

A Few of the Namespaces that get imported by default in an ASPX file are

- * **System, System.Data, System.Drawing, System.Globalization**
- * **System, System.IO, System.Management, System.Globalization**
- * **System, System.Collections, System.Text, System.Web**
- * **System, System.NET, System.Reflection, System.Web**

System, System.Collections, System.Text, System.Web

The Assemblies that can be referenced in an ASPX file without using @Assembly Directive is

- * **System.dll, System.Data.dll, System.Web.dll, System.Xml.dll,**
- * **System.dll, System.Collections.dll, System.IO.dll**
- * **System.dll, System.Reflection.dll, System.Globalization.dll,**
- * **System.Drawing.dll, System.Assembly.dll, System.Text.dll**

System.dll, System.Data.dll, System.Web.dll, System.Xml.dll

An .ASHX file contains the following

- * **Code-behind that are used in the code**
- * **Server Controls that can be called from a code-behind file**
- * **HTTP handlers-software modules that handle raw HTTP requests received by ASP.NET**
- * **Contains normal ASP.NET code and can be used as an include file**

HTTP handlers-software modules that handle raw HTTP requests received by ASP.NET

What is the output for the following code snippet:

```
public class testClass
```

```

{
public static void Main(string[] args)
{
System.Console.WriteLine(args[1]);
} //end Main
} //end class testClass

```

- * **Compiler Error**
- * **Runtime Error**
- * **Hello C# world**
- * **None of the above**

Runtime Error

One of the possible way of writing an ASP.NET handler that works like an ISAPI filter- that is, that sees requests and responses and modifies them also, is by,

- * **writing a module that extends FormsAuthenticatonModule and using it**
- * **writing a component class that extends HttpModuleCollection and using it**
- * **writing an HTTP module-a Class that implements IHttpModule and registering it in Web.Config**
- * **All of the above**

writing an HTTP module-a Class that implements IHttpModule and registering it in Web.Config

The ASP.NET directive that lets you cache different versions of a page based on varying input parameters, HTTP headers and browser type is

- * **@OutputCache**
- * **@CacheOutput**
- * **@PageCache**
- * **@CacheAll**

@ OutputCache

If we develop an application that must accommodate multiple security levels through secure login and ASP.NET web application is spanned across three web-servers (using round-robin load balancing) what would be the best approach to maintain login-in state for the users?

- * **<SessionState mode="InProc" stateConnectionString=" tcpip=127.0.0.1:42424" sqlConnectionString=" data source=127.0.0.1;user id=sa;password=" cookieless="false" timeout="30" />**
- * **<SessionState mode="OutProc" stateConnectionString="tcpip=127.0.0.1:42424" sqlConnectionString="data source=127.0.0.1;user id=sa;password=" cookieless="false" timeout="30" />**
- * **<SessionState mode="stateserver" stateConnectionString="tcpip=127.0.0.1:42424" sqlConnectionString="data source=127.0.0.1;user id=sa;password=" cookieless="false" timeout="30" />**

SessionState mode="stateserver" stateConnectionString="tcpip=127.0.0.1:42424?
sqlConnectionString="data source=127.0.0.1;user id=sa;password=" cookieless="false" timeout="30?"

Visit <http://www.downloadmela.com/> for more papers

What is the output for the below mentioned compilation command `>csc /addmodule:A.Exe B.Cs`

- * A.exe
- * B.exe
- * A.dll
- * B.dll

B.exe

How can be the web application get configured with the following authorization rules

* Anonymous users must not be allowed to access the application.

* All persons except David and John must be allowed to access the application.

o `<authorization><allow roles = "*" ><deny roles = "?" ></authorization>`

o `<authorization><deny users = "applicationname\David; applicationname\John" ><deny users = "*" ></authorization>`

o `<authorization><deny users = "applicationname\David, applicationname\John" ><deny users = "?" ><allow users = "*" ></authorization>`

o `<authorization><allow users = "*" ><deny users = "applicationname\David, applicationname\John" ></authorization>`

`authorization deny users = "applicationname\David, applicationname\John" deny users = "?" allow users = "*" /authorization`

What will be the output of the following code snippet?

```
using System;
```

```
class MainClass
```

```
{
```

```
static void Main()
```

```
{
```

```
new MainClass().Display( 3.56 );
```

```
}
```

```
private void Display( float anArg )
```

```
{
```

```
Console.Write( "{0} {1}", anArg.GetType(), anArg );
```

```
}
```

```
double Display( double anArg )
```

```
{
```

```
Console.Write( "{0} {1}", anArg.GetType(), anArg );
```

```
return anArg;
```

```
}
```

```
public decimal Display( decimal anArg )
```

```
{
```

```
Console.Write( "{0} {1}", anArg.GetType(), anArg ); return anArg;
```

```
}
```

```
}
```

- * **System.Single 3.56**
- * **System.Float 3.56**
- * **System.Double 3.56**
- * **System.Decimal 3.56**

System.Double 3.56

What will be output for the given code?

Dim I as integer = 5

Do

I = I + 2

Response.Write (I & " ")

Loop Until I > 10

* **5 8**

* **5 7 9**

* **7 9 11**

* **Errors out**

7 9 11 (if corrected)

Describe the advantages of writing a managed code application instead of unmanaged one.

What's involved in certain piece of code being managed?

The advantages include automatic garbage collection, memory management, support for versioning and security. These advantages are provided through .NET FCL and CLR, while with the unmanaged code similar capabilities had to be implemented through third-party libraries or as a part of the application itself.

Are COM objects managed or unmanaged?

Since COM objects were written before .NET, apparently they are unmanaged.

Any code not written in the Microsoft .NET framework environment is UNMANAGED. So naturally COM+ is unmanaged because it is written in Visual Basic 6.

So can a COM object talk to a .NET object?

Yes, through Runtime Callable Wrapper (RCW) or PInvoke.

How do you generate an RCW from a COM object?

Use the Type Library Import utility shipped with SDK. `tlbimp COMObject.dll /out:.NETObject.dll` or reference the COM library from Visual Studio in your project.

I can't import the COM object that I have on my machine. Did you write that object?

You can only import your own objects. If you need to use a COM component from another developer, you should obtain a Primary Interop Assembly (PIA) from whoever authored the original object.

The answer to (5) is only partially correct. You *can* import COM objects, even if they are not your own. It is recommended that you obtain the Primary Interop from the vendor, but not required. The most common problem importing a COM DLL is that it exposes objects that form part of a separate COM DLL. You can generate interops for these additional DLLs, and then refer to them when

importing the problem DLL.

How do you call unmanaged methods from your .NET code through PInvoke?

Supply a DllImport attribute. Declare the methods in your .NET code as static extern. Do not implement the methods as they are implemented in your unmanaged code, you're just providing declarations for method signatures.

Can you retrieve complex data types like structs from the PInvoke calls?

Yes, just make sure you re-declare that struct, so that managed code knows what to do with it.

I want to expose my .NET objects to COM objects. Is that possible?

Yes, but few things should be considered first. Classes should implement interfaces explicitly. Managed types must be public. Methods, properties, fields, and events that are exposed to COM must be public. Types must have a public default constructor with no arguments to be activated from COM. Types cannot be abstract.

Can you inherit a COM class in a .NET application?

The .NET Framework extends the COM model for reusability by adding implementation inheritance. Managed types can derive directly or indirectly from a COM coclass; more specifically, they can derive from the runtime callable wrapper generated by the runtime. The derived type can expose all the method and properties of the COM object as well as methods and properties implemented in managed code. The resulting object is partly implemented in managed code and partly implemented in unmanaged code.

Suppose I call a COM object from a .NET application, but COM object throws an error. What happens on the .NET end?

COM methods report errors by returning HRESULTs; .NET methods report them by throwing exceptions. The runtime handles the transition between the two. Each exception class in the .NET Framework maps to an HRESULT.

What do you know about .NET assemblies?

Assemblies are the smallest units of versioning and deployment in the .NET application. Assemblies are also the building blocks for programs such as Web services, Windows services, serviced components, and .NET Remoting applications.

What's the difference between private and shared assembly?

Private assembly is used inside an application only and does not have to be identified by a strong name. Shared assembly can be used by multiple applications and has to have a strong name.

What's a strong name?

A strong name includes the name of the assembly, version number, culture identity, and a public key token.

How can you tell the application to look for assemblies at the locations other than its own install?

Use the directive in the XML .config file for a given application.

```
<probing privatePath="c:\mylibs; bin\debug" />
```

should do the trick. Or you can add additional search paths in the Properties box of the deployed application.

How can you debug failed assembly binds?

Use the Assembly Binding Log Viewer (fuslogvw.exe) to find out the paths searched.

Where are shared assemblies stored?

Global assembly cache.

How can you create a strong name for a .NET assembly?

With the help of Strong Name tool (sn.exe).

Where's global assembly cache located on the system?

Usually C:\winnt\assembly or .

Can you have two files with the same file name in GAC?

Yes, remember that GAC is a very special folder, and while normally you would not be able to place two files with the same name into a Windows folder, GAC differentiates by version number as well, so it's possible for MyApp.dll and MyApp.dll to co-exist in GAC if the first one is version 1.0.0.0 and the second one is 1.1.0.0.

So let's say I have an application that uses MyApp.dll assembly, version 1.0.0.0. There is a security bug in that assembly, and I publish the patch, issuing it under name MyApp.dll 1.1.0.0. How do I tell the client applications that are already installed to start using this new MyApp.dll?

Use publisher policy. To configure a publisher policy, use the publisher policy configuration file, which uses a format similar app.config file. But unlike the app.config file, a publisher policy file needs to be compiled into an assembly and placed in the GAC.

What is delay signing?

Delay signing allows you to place a shared assembly in the GAC by signing the assembly with just the public key. This allows the assembly to be signed with the private key at a later stage, when the development process is complete and the component or assembly is ready to be deployed. This process enables developers to work with shared assemblies as if they were strongly named, and it secures the private key of the signature from being accessed at different stages of development.