



Ficsoft Technologies Pvt Ltd.

ISO 9001 : 2015 Certified Company

- Gateway to IT JOBS

.Net (dot-net) interview questions for Fresher and experienced. →

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. .net Supports More than 44 languages.

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.

What is an application server?



As defined in Wikipedia, an application server is a software engine that delivers applications to client computers or devices. The application server runs your server code. Some well known application servers are IIS (Microsoft), WebLogic Server (BEA), JBoss (Red Hat), WebSphere (IBM).

What is inheritance?



Inheritance represents the relationship between two classes where one type derives functionality from a second type and then extends it by adding new methods, properties, events, fields and constants. C# support two types of inheritance:

- Implementation inheritance
- Interface inheritance

What is implementation and interface inheritance?



When a class (type) is derived from another class(type) such that it inherits all the members of the base type it is Implementation Inheritance.

When a type (class or a struct) inherits only the signatures of the functions from another type it is Interface Inheritance.

In general Classes can be derived from another class, hence support Implementation inheritance. At the same time Classes can also be derived from one or more interfaces. Hence they support Interface inheritance.

What is inheritance hierarchy?



The class which derives functionality from a base class is called a derived class. A derived class can also act as a base class for another class. Thus it is possible to create a tree-like structure that illustrates the relationship between all related classes. This structure is known as the inheritance hierarchy.

How do you prevent a class from being inherited?



In VB.NET you use the NotInheritable modifier to prevent programmers from using the class as a base class. In C#, use the sealed keyword.

Define Overriding?



Overriding is a concept where a method in a derived class uses the same name, return type, and arguments as a method in its base class. In other words, if the derived class contains its own implementation of the method rather than using the method in the base class, the process is called overriding.

Can you use multiple inheritance in .NET?



.NET supports only single inheritance. However the purpose is accomplished using multiple interfaces.

What is an Interface?



An interface is a standard or contract that contains only the signatures of methods or events. The implementation is done in the class that inherits from this interface. Interfaces are primarily used to set a common standard or contract.

What is business logic?



It is the functionality which handles the exchange of information between database and a user interface.

What is a component?



Component is a group of logically related classes and methods. A component is a class that implements the IComponent interface or uses a class that implements IComponent interface.

What is a control?



A control is a component that provides user-interface (UI) capabilities.

What are design patterns?



Design patterns are common solutions to common design problems.

What is a connection pool?



A connection pool is a 'collection of connections' which are shared between the clients requesting one. Once the connection is closed, it returns back to the pool. This allows the connections to be reused.

What is a flat file?



A flat file is the name given to text, which can be read or written only sequentially.

What are functional and non-functional requirements?



Functional requirements defines the behavior of a system whereas non-functional requirements specify how the system should behave; in other words they specify the quality requirements and judge the behavior of a system. E.g. Functional - Display a chart which shows the maximum number of Customer in a region.

Non-functional – The data presented in the chart must be updated every 60 minutes.

What is the global assembly cache (GAC)?



GAC is a machine-wide cache of assemblies that allows .NET applications to share libraries. GAC solves some of the problems associated with dll's (DLL Hell).

What is Boxing/Unboxing?



Boxing is used to convert value types to object.

E.g. `int x=1;`

`object obj=x ;`

Unboxing is used to convert the object back to the value type.

E.g. `int y=(int)obj;`

Boxing/unboxing is quite an expensive operation.

What is globalization?



Globalization is the process of customizing applications that support multiple cultures and regions.

What is localization?



Localization is the process of customizing applications that support a given culture and regions.

What is MIME?



The definition of MIME or Multipurpose Internet Mail Extensions as stated in MSDN is “MIME is a standard that can be used to include content of various types in a single message. MIME extends the Simple Mail Transfer Protocol (SMTP) format of mail messages to include multiple content, both textual and non-textual. Parts of the message may be images, audio, or text in different character sets. The MIME standard derives from RFCs such as 2821 and 2822”.

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.

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 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 computer, 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 "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 types. 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'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.

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 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.