**The five pillars of c# interviews**

Below are some questions which are asked from OOP perspective:-

1. **Principles of OOP**
   1. Inheritance
   2. Polymorphism
   3. Abstraction
   4. Encapsulation
2. **Abstract VS encapsulation**

Abstraction: Abstraction is the act of representing essential information without including background details and explanations.

Encapsulation: Encapsulation is the act of wrapping up of attributes (represented by data members) and operations (represented by functions) under one single unit (represented by class).

Taking **Real world example**



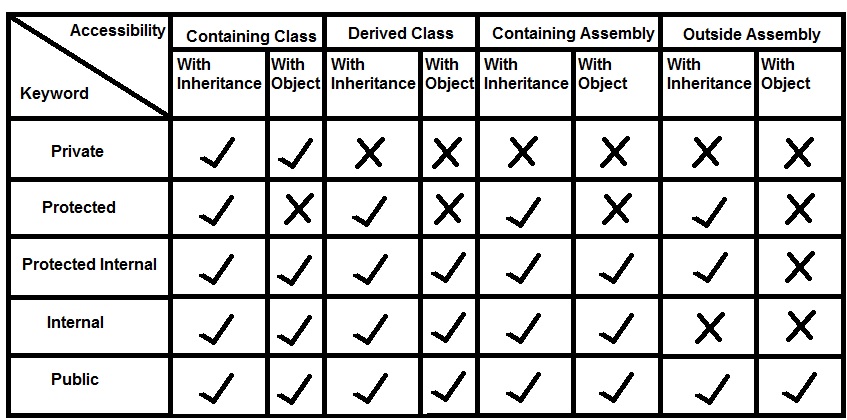
Suppose you go to an *automatic cola vending machine* and request for a *cola*. The machine processes your request and gives the cola.

* Here automatic cola vending machine is a class. It contains both data i.e. Cola can and operations i.e. service mechanism and they are wrapped/integrated under a single unit Cola Vending Machine. This is called **Encapsulation.**
* You need not know how the machine is working. This is called **Abstraction.**
* You can interact with cola can only through service mechanism. You cannot access the details about internal data like how much cans it contains, mechanism etc. This is **Data Hiding.**
* You cannot pick the can directly. You request for cola through proper instructions and request mechanism (i.e. by paying amount and filling request) and get that cola only through specified channel. This is **message passing.**

The working and data is hidden from you. This is possible because that Vending machine is made (or Encapsulated or integrated) so. Thus, we can say **Encapsulation is a way to implement Abstraction.**

1. **How did you implement encapsulation?**

Encapsulation is implemented by using **access specifiers**. An **access specifier** defines the scope and visibility of a class member. C# supports the following access specifiers −



1. **Difference between shadowing and overriding?**

There is a major difference in shadowing and overriding which is normally when we override a virtual method in derived class and create an instance of derived class, and then if we hold reference to the derived class object as a base class object, and call that member, it always calls derived class implementation which is supposed to happen. In shadowing the case is different, if for the same virtual member, we shadow it in the derived class using a new keyword and we call the implementation as above, it will call base class implementation when we have reference to an object of type base class. And if we have reference to the same object of derived type it will call derived type implementation, so base class and derived class implementation are hidden from each other. The method of choosing which implementation to be called depends upon if we are calling the member using reference of base type or derived type.

Shadowing is actually hiding overridden method implementation in derived class and call the parent call implementation using derived class object.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

Logger logger = new Logger();

logger.Log("Log started!");

logger.Log("Base: Log Continuing");

logger.LogCompleted();

Console.ReadKey();

}

}

public abstract class BaseLogger

{

public virtual void Log(string message)

{

Console.WriteLine("Base: " + message);

}

public void LogCompleted()

{

Console.WriteLine("Completed");

}

}

public class Logger : BaseLogger

{

public override void Log(string message)

{

Console.WriteLine(message);

}

public new void LogCompleted()

{

Console.WriteLine("Finished");

}

}

}

Output:

Log started!

Base: Log Continuing

Completed

1. Different types of polymorphism in c#.

The second category is .NET fundamentals. In this section interviewer expects you to know things like delegates, events, garbage collector, reflection etc. As a c# developer everything we develop using .NET framework, so again this is a crucial part of c# interviews. Some most asked questions from .NET fundamental sections which are asked:-

* CLR, CTS, CAS,IL code etc.
* Delegates , events , difference between them
* Reflection,threading, application domain,boxing, unboxing etc.

Again .NET framework has lot of versions and the interviewer will expect that you important features of each version.

The third category of those five pillars which is important is ASP.NET. Almost all software projects are now built as web applications and ASP.NET is the framework for creating web application in Microsoft technology. So again this section becomes compulsory and important. The most favorite questions in ASP.NET revolve around page life cycle, authentication, authorization, sessions, view state and caching. Below goes most common topics for ASP.Net interviews questions:-

* Page life cycle.
* Authentication and Authorization
* Session , cache , view state and applications.
* Server controls and user controls.

The fourth category is SQL server. RDBMS forms one of the important parts of any software project. SQL Server is the primary RDBMS product. Again this section you need to prepare thoroughly. I have seen many c# interviewers very particular about SQL queries. In this section mostly I have seen interviewer throwing some tricky SQL queries, asking some database design questions and some questions around stored procedures,triggers, views and cursors. Below are some sections which are discussed during SQL server questions:-

* Database normalization (first normal form, second normal form and third normal form).
* SQL Joins
* Triggers, stored procedures, Views and cursors.
* UNION and UNION ALL.

The fifth section is ADO.NET. ADO.NET is a component which helps us to connect to SQL Server and fetch data in to c# applications. Some of the question which is making rounds in ADO.NET are datareader versus dataset , use of dataview, connection pooling etc. Below are some sections which will help you prepare around ADO.NET:-

* ADO.NET components
* Dataset VS datareader
* Using SP with ADO.NET
* Locking and connection pooling.

**Second level WCF / MVC and EF**

If you are more than 3 to 4 years of experience interviewers expects you to answer questions around these 3 sections WCF, MVC and EF.

WCF is one of the most discussed topics when it comes to c# interviewers. If you are a senior this topic is a must in your portfolio. Following are some most frequently asked WCF questions:-

* Why WCF ?
* WCF vsWebservicesvsremoting
* Operation contract, data contract and service contract.
* One way and two way contracts.
* Security, Transactions,REST , Instancing etc.

MVC is again one of the must topics when it goes above 3 years of experience. On the contrary after some years I am confident this will be the sixth compulsory pillar for .NET interviews. Again EF can be one of the asked topics. Now EF is not compulsory but depending on job description , it can be a make or break thing.

**Optional but can make a difference**

C# is vast, so preparing everything is just not possible. There are some optional sections. Again depending on JD (job description) these optional sections can be do or die. In case are dropping these optional sections make sure it’s not a part of the JD.  There are 3 things which can be kept optional:-

* Business intelligence (SSIS, SSAS and SSRS). Just a thin line here SSRS is not exactly optional. It’s the prime reporting tool of MS. Ensure you prepare this section and not drop it completely out.
* Sharepoint is a server product. Though it’s hot you can still exclude the same. But if your job description is saying sharepoint you need to prepare this section as well.
* WPF. This section can be omitted, if you are pure web developer, you would be hardly concerned with WPF. But if you are applying for job description which is doing windows development then WPF is a must.

**Senior level categories**

For senior’s, questions around design patterns, UML and project management will be asked more. They will probed around management capabilities, writing technical documents, architecture etc.

**Others**

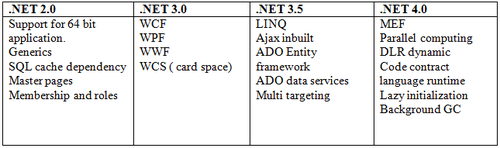
Other than technical capabilities, people also question around:-

* What’s your role?
* What’s your architecture of your current project?
* Why you are leaving your current organization?
* What’s your salary expectation?

**Exhaustive list of c# interview question covering 16 important c# section**

Can you explain architecture of your current project?  
What role did you play in your project and company?  
What’s your salary expectation?  
Why do you want to leave your previous organization?  
How much do you rate yourself between 1 to 10?  
Can you speak about yourself?  
**How can we improve performance of .NET?**

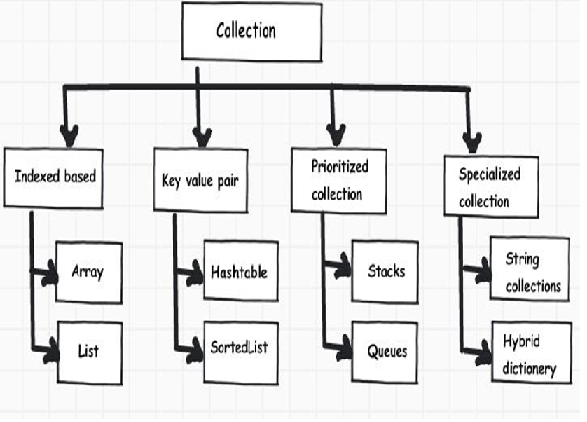
**What is the difference between .NET 1.X,2.0,3.0, 3.5 and 4.0?**



What is ILcode, JIT, CLR, CTS, CLS and CAS?  
What is a garbage collector?  
What is GAC?  
**What are stack heap value, reference types, boxing and unboxing?**

<https://www.codeproject.com/Articles/76153/Six-important-NET-concepts-Stack-heap-value-types>

**How are exceptions handled in .NET?**  
Using try catch finally

**What are different types of collections in .NET?**

**What are generics?**

Generics refers to features of C# that allow to declare methods, class, interfaces etc. independent of the data type that the particular method, class, interface etc. stores or operates on it.

Advantages of Generics In C#

* Type safety

- This ensures that the method, class, or interface etc. works only on the correct data type. For example, if you use string when calling a generic method, then all the values passed in as expected to be strings. If you pass in an int value parameter, the compiler will detect that as an error and the program will not be able to compile.

* Errors/Exceptions are caught at compile time as opposed to runtime errors/exceptions

- If given a choice between compile time and runtime errors, 100% of developers choose compile time errors so should you. The last thing we want is errors crashing the program when the users are working with it.

* Re-usability & Flexibility

- In the above example on the problem statement, we declared more than one method and the only difference was the data type of the parameters that we passed. With generics, you only have to create one method and re-use it with any data type that you want

* Write clean code

EX:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace CSharpGenerics

{

class Program

{

static void Main(string[] args)

{

//problem statement

string value = IsEqual(1, "1") ? "Equal" : "Not Equal";

Console.WriteLine("Non Generic Method string " + value);

//generic method solution

string genericValue = IsEqual<string>("1", "1") ? "Equal" : "Not Equal";

Console.WriteLine("Generic Method string " + genericValue);

string genericValue2 = IsEqual<int>(1, 2) ? "Equal" : "Not Equal";

Console.WriteLine("Generic Method int " + genericValue2);

Console.ReadLine();

}

static bool IsEqual(string x, string y)

{

return x.Equals(y);

}

static bool IsEqual(int x, int y)

{

return x.Equals(y);

}

static bool IsEqual(object x, object y)

{

return x.Equals(y);

}

static bool IsEqual<T>(T x, T y)

{

return x.Equals(y);

}

}

}

Explain Abstraction, encapsulation, inheritance and polymorphism?

**How is abstract class different from an interface?**

Abstract will have common feature, interface is kind of contract

What are the different types of polymorphism?

How does delegate differ from an event?

**What are different access modifiers?**

Public, internal, protected internal, protected, private.

**Can you explain connection, command, data reader and dataset in ADO.NET?**

**Connection**: - This object creates a connection to the database. If you want to do any operation on the database you have to first create a connection object.

**Command**: - This object helps us to execute SQL queries against database. Using command object we can execute select, insert, update and delete SQL command.

**Data reader**: - This provides a record set which can be browsed only in forward direction. It can only be read but not updated. Data reader is good for large number of records where you want to just browse quickly and display it.

**Dataset object**: - This provides a record set which can be read back and in forward direction. The record set can also be updated. Dataset is like an in memory database with tables, rows and fields.

**Data Adapter**: - This object acts as a bridge between database and dataset; it helps to load the dataset object.

**How does “Dataset” differ from a “Data Reader”?**

**DataReader** is used to read the data from database and it is a read and forward only connection oriented architecture during fetch the data from database. DataReader will fetch the data very fast when compared with dataset. Generally we will use ExecuteReader object to bind data to datareader.

* Holds the connection open until you are finished (don't forget to close it!).
* Can typically only be iterated over once
* Is not as useful for updating back to the database

To bind DataReader data to GridView we need to write the code like as shown below:

Protected void BindGridview() {

using(SqlConnection conn = new SqlConnection("Data Source=abc;Integrated Security=true;Initial Catalog=Test")) {

con.Open();

SqlCommand cmd = new SqlCommand("Select UserName, First Name,LastName,Location FROM Users", conn);

SqlDataReader sdr = cmd.ExecuteReader();

gvUserInfo.DataSource = sdr;

gvUserInfo.DataBind();

conn.Close();

}

}

**DataSet** is a disconnected orient architecture that means there is no need of active connections during work with datasets and it is a collection of DataTables and relations between tables. It is used to hold multiple tables with data. You can select data form tables, create views based on table and ask child rows over relations. Also DataSet provides you with rich features like saving data as XML and loading XML data.

protected void BindGridview() {

SqlConnection con = new SqlConnection("Data Source=abc;Integrated Security=true;Initial Catalog=Test");

conn.Open();

SqlCommand cmd = new SqlCommand("Select UserName, First Name,LastName,Location FROM Users", conn);

SqlDataAdapter sda = new SqlDataAdapter(cmd);

DataSet ds = new DataSet();

da.Fill(ds);

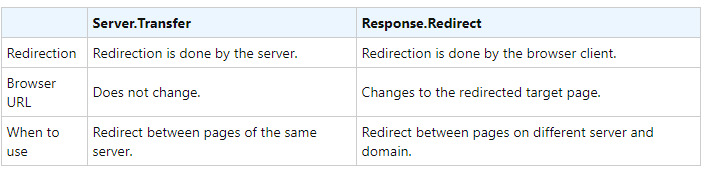
gvUserInfo.DataSource = ds;

gvUserInfo.DataBind();

}

How is ASP.NET page life cycle executed?  
What are Httphandlers and HttpModules and difference between them?  
What are different kind of validator controls in ASP.NET ?

**How is ‘Server.Transfer’ different from ‘response.Redirect’ ?**



**Can you explain windows, forms and passport authentication?**

**Windows authentication** enables you to identify users without creating a custom page. Credentials are stored in the Web server s local user database or an Active Directory domain. Once identified you can use the user s credentials to gain access to resources that are protected by Windows authorization.

**Forms authentication** enables you to identify users with a custom database such as an ASP.NET membership database. Alternatively you can implement your own custom database. Once authenticated you can reference the roles the user is in to restrict access to portions of your Web site.

**Passport authentication** relies on a centralized service provided by Microsoft. Passport authentication identifies a user with using his or her e-mail address and a password and a single Passport account can be used with many different Web sites. Passport authentication is primarily used for public Web sites with thousands of users.

**What is difference between Grid view, Data list, and repeater?**

* Gridview - Limited in design, works like an html table. More in built functionality like edit/update, page, sort. Lots of overhead.
* DataGrid - Old version of the Gridview. A gridview is a super datagrid.
* Datalist - more customisable version of the Gridview. Also has some overhead. More manual work as you have to design it yourself.
* ListView - the new Datalist :). Almost a hybrid of the datalist and gridview where you can use paging and build in Gridview like functionality, but have the freedom of design. One of the new controls in this family
* Repeater - Very light weight. No built in functionality like Headers, Footers. Has the least overhead.

**Which are the various modes of storing ASP.NET session?**

* InProc: - In this mode Session, state is stored in the memory space of the Aspnet\_wp.exe process. This is the default setting. If the IIS reboots or web application restarts then session state is lost.
* StateServer:-In this mode Session state is serialized and stored in a separate process (Aspnet\_state.exe); therefore, the state can be stored on a separate computer (a state server).
* SQL SERVER: - In this mode Session, state is serialized and stored in a SQL Server database.

**Interface vs abstract class?**

[**http://csharp-video-tutorials.blogspot.com/2013/09/part-3-why-and-when-should-we-use.html**](http://csharp-video-tutorials.blogspot.com/2013/09/part-3-why-and-when-should-we-use.html)

How can we do caching in ASP.NET?

Caching is the technique of storing frequently used data/pages in memory.

To cache a webform, use the **@OutputCache** page directive. The @OutputCache directive has 2 mandatory attributes  
**Duration** - Specifies the time in seconds for which the webform should be cached  
**VaryByParam** - Cache multiple responses of a single webform. For now set the value to "None". We will discuss about "VaryByParam" in a later video.  
  
**Webform with the following "OutputCache" directive is cached for 30 seconds.**  
<%@ OutputCache Duration="30" VaryByParam="None" %>  
  
When any user requests this Web form for the first time, the web server will process the web form events, execute the stored procedure, create objects, generate HTML and send that HTML to the client browser, and retains a copy of the response in memory for the next 30 seconds. Any subsequent requests during that time receive the cached response.   
  
After the cache duration has expired, the next request for the Web form, has to process the web form events, execute the stored procedure, create objects, generate HTML, which is then cached for another 30 seconds. So this web form is processed by the server, once every 30 second, at the most.

**What is ViewState?**

<https://www.codeproject.com/Articles/492397/State-Management-in-ASP-NET-Introduction>

What are indexes and what is the difference between clustered and non-clustered?

**How is stored procedure different from functions?**

* Function must return a value but in Stored Procedure it is optional (Procedure can return zero or n values).
* Functions can have only input parameters for it whereas Procedures can have input/output parameters.
* Functions can be called from Procedure whereas Procedures cannot be called from Function.
* Procedure allows SELECT as well as DML(INSERT/UPDATE/DELETE) statement in it whereas Function allows only SELECT statement in it.
* Procedures cannot be utilized in a SELECT statement whereas Function can be embedded in a SELECT statement.
* Stored Procedures cannot be used in the SQL statements anywhere in the WHERE/HAVING/SELECT section whereas Function can be.

**What’s the difference between web services and remoting?**

Web services are cross platform, using common standards and work through firewalls. They also think in terms of messages, not objects - you send a message to a service, and you get a reply.

Remoting is an MS only technology which is not cross platform and talks in a binary format. It thinks in terms of objects, you create an object on the remote server and work with it. It doesn't work well with firewalls. Remoting is also dead these days, MS favour WCF (which includes web services)

**What’s the difference between WCF and Web services?**

What are end point, contract, address, and bindings?  
What is WPF and silverlight?

What is LINQ and Entity framework?

**What’s the difference between LINQ to SQL and Entity framework?  
Entity Framework**

Entity Framework used various Database(such as oracle,db2,sql server, mysql etc).

Entity Framework generate .edmx extension file.

It is support complex type.

Relationship tables/view one-to-one, one-to-many, many-to-many.

Three Difference way to using data Query.

* Entity SQL
* ObjectContext
* DbContext

Entity Framework provide loosely coupled approach.

**Linq To SQL**

Linq To SQL Used only one Sql Server Database.

It is generate .dbml extension file.

It is not support complex type.

Relationship Tables/view one-to-one.

It is use data query DataContext

It is provide tightly coupled approach.

What are design patterns?  
Which design patterns are you familiar with?  
Can you explain singleton pattern?  
What is MVC, MVP and MVVM pattern?

**What is UML and which are the important diagrams?**

<https://www.smartdraw.com/uml-diagram/>

What are different phases in a software life cycle?

What is Ajax?

How did you do unit testing in your project?

What is Agile?

How did you do code reviews?

**Sonar Cube**

How did you convert requirements to technical document?

**What is an IL code?**

Why IL code is not fully compiled?

Who compiles the IL code and how does it work?

How does JIT compilationwork?

What are different types of JIT?

What is Native Image Generator (Ngen.exe)?

So does it mean that NGEN.EXE will always improve performance?

What is a CLR?

What is the difference betweenmanaged and unmanaged code?

What is a garbage collector?

What are generations in Garbage collector (Gen 0, 1 and 2)?

Garbage collector cleans managed code,how do we clean unmanaged code?

But when we create a destructor the performance falls down?

So how can we clean unmanaged objects and also maintain performance?

Can we force garbage collector to run?

What is difference between finalize and dispose?

What is CTS?

What is a CLS (Common Language Specification)?

What is an Assembly?  
What are the different types of Assembly?  
What is Namespace?  
What is Difference between NameSpace and Assembly?  
What is ILDASM?  
What is Manifest?  
Where is the version information stored of an assembly?  
Is versioning applicable to private assemblies?  
What is the use of strong names?  
What is Delay signing?  
What is GAC?  
How to add and remove an assembly from GAC?  
If we have two versions of the same assembly in GAC how to we make a choice?  
What is reflection?  
What are stack and heap?  
What are Value types and Reference types?  
What is concept of Boxing and Unboxing?  
How performance is affected due to boxing and unboxing?  
How can we avoid boxing and unboxing?  
How to prevent my .NET DLL to be decompiled?

**What is the difference between Convert.toString and .toString () method?**

Convert.ToString() handles null, while ToString() doesn't.

How can we handle exceptions in .NET?

**How can I know from which source the exception occurred?**

What if we do not catch the exception?

**What are system level exceptions and application level exceptions?**

* Exceptions that are thrown by .NET framework are called as system exceptions. These errors are non-recoverable or fatal errors like ArgumentOutOfRangeException, IndexOutOfRangeException, StackOverflowException etc.
* Application exceptions are custom exceptions created for the application. Application exceptions are created by deriving from "ApplicationException" class as shown below. You can then create the object of the below exception and throw the same from the code and catch the same on the client side.

**Can two catch blocks be executed?**No

What are different types of collections in .NET?

**What is the difference between array, array list and list?**

* An Array (System.Array) is fixed in size once it is allocated. You can't add items to it or remove items from it. Also, all the elements must be the same type. As a result, it is type safe, and is also the most efficient of the three, both in terms of memory and performance. Also, System.Array supports multiple dimensions (i.e. it has a [Rank](https://msdn.microsoft.com/en-us/library/system.array.rank(v=vs.110).aspx) property) while List and ArrayList do not (although you can create a List of Lists or an ArrayList of ArrayLists, if you want to).

Eg: int[] Array = new Int[5];

* An ArrayList is a flexible array which contains a list of objects. You can add and remove items from it and it automatically deals with allocating space. If you store value types in it, they are boxed and unboxed, which can be a bit inefficient. Also, it is not type-safe.

Eg: ArrayList arrayList = new ArrayList();

* A List<> leverages generics; it is essentially a type-safe version of ArrayList. This means there is no boxing or unboxing (which improves performance) and if you attempt to add an item of the wrong type it'll generate a compile-time error.

Eg: List<int> list = new List<int>();

What are hashtable collections?

**What are Queues and stack collection?**

* Queue 🡪 FIFO
* Stack 🡪 LIFO

Can you explain generics in .NET?

Can you explain the concept of generic collection?

**What is the difference between dictionary and hashtable?**

**Dictionary:**

– Dictionary is a generic type which means we can use it with any data type.

– Only public static members are thread safe.

– It returns error if we try to find a key which does not exist.

– It is faster than a Hashtable because there is no boxing and unboxing.

**Hashtable:**

– Hashtable is not a generic type.

– All the members in a Hashtable are thread safe.

– It returns null if we try to find a key which does not exist.

– It is slower than dictionary because it requires boxing and unboxing.

**What are the generic equivalent for array list,stack, queues and hashtable?**

Lists allow duplicate items, can be accessed by index, and support linear traversal.

* **ArrayList** - An array-based list that doesn't support generic types. It does not enforce type safety and should generally be avoided.
* **List** - An array list that supports generic types and enforces type-safety. Since it is non-contiguous, it can grow in size without re-allocating memory for the entire list. This is the more commonly used list collection.
* **Hashes**

Hashes are look-ups in which you give each item in a list a "key" which will be used to retrieve it later. Think of a hash like a table index where you can ask questions like "I'm going to find this object by this string value. Duplicate keys are not allowed.

* **HashTable** - A basic key-value-pair map that functions like an indexed list.
* Dictionary - A hashtable that supports generic types and enforces type-safety.
* **Queues**

Queues control how items in a list are accessed. You typically push/pop records from a queue in a particular direction (from either the front or back). Not used for random access in the middle.

* **Stack** - A LIFO (last in, first out) list where you push/pop records on top of each other.
* **Queue** - A FIFO (first in, first out) list where you push records on top and pop them off the bottom.

What is the use of IEnumerable, ICollection, Ilist and IDictionary?

What is code access security (CAS)?

So how does CAS actually work?

Is CAS supported in .NET 4.0?

**What is sandboxing?**

software sandbox, we let any software(child) to execute(play) but with some restrictions over what it (he) can do. We can feel safe & secure about what the executing software can do.

You've seen & used Antivirus software. Right? It is also a kind of sandbox. It puts restrictions on what any program can do. When a malicious activity is detected, it stops and informs user that "this application is trying to access so & so resources. Do want to allow?"

**How can we create a windows service using .NET?**

**What is serialization and deserialization in .NET?**

When you create an object in a .Net framework application, you don't need to think about how the data is stored in memory. Because the .Net Framework takes care of that for you. However, if you want to store the contents of an object to a file, send an object to another process or transmit it across the network, you do have to think about how the object is represented because you will need to convert to a different format. This conversion is called SERIALIZATION.

Serialization allows the developer to save the state of an object and recreate it as needed, providing storage of objects as well as data exchange. Through serialization, a developer can perform actions like sending the object to a remote application by means of a Web Service, passing an object from one domain to another, passing an object through a firewall as an XML string, or maintaining security or user-specific information across applications.

Can you mention some scenarios where we can use serialization?

**When should we use binary serialization as compared to XML serialization?**

Specific to .NET, If you have two applications that are using the same type system, then you can use binary serialization. On the other hand if you have applications that are in different platforms then it is recommended to use XML Serialization. So if i am writing a chat application (client and server), I might use binary serialization, but if I later decide that I should use Python to write a client, then I may not.

Can you explain the concept of “Short Circuiting”?

**What is the difference between “Typeof” and “GetType” ?**

GetType: when you want to obtain the type from an instance of your class, you use GetType, gets resolved at runtime

TypeOf: If you don't have an instance, but you know the type name you would use typeof. resolved at compile time.

What is Object Oriented Programming?

What is a Class and object?

**What are different properties provided by Object-oriented systems?**Inheritance, abstraction, encapsulation, polymorphism

How can we implement encapsulation in .NET?

What’s the difference between abstraction and encapsulation?

How is inheritance implemented in .NET?

What are the two different types of polymorphism?

**How can we implement static polymorphism?**

Methodoverloading

**How can we implement dynamic polymorphism?**Methodoverriding

What is the difference overriding and overloading?

What is operator overloading?

What are abstract classes?

What are abstract methods?

What is an Interface?

**Do interface have accessibility modifier.**

No

**Can we create an object of abstract class or an interface?**No

What is difference between abstract classes and interfaces?

**An abstract with only abstract method, how is it different from interfaces?**

* interfaces can have no state or implementation
* a class that implements an interface must provide an implementation of all the methods of that interface
* abstract classes may contain state (data members) and/or implementation (methods)
* abstract classes can be inherited without implementing the abstract methods (though such a derived class is abstract itslef)
* interfaces may be multiple-inherited, abstract classes may not (this is probably the key concrete reason for interfaces to exist separately from abtract classes - they permit an implementation of multiple inheritance that removes many of the problems of general MI).

**If we want to update interface with new methods, what is the best practice?**

Create new interface for the new functionality without effect old one.

What is a delegate?  
How can we create a delegate?  
What is a multicast delegate?  
What are Events?  
What is the difference between delegate and events?  
Do events have return type?  
Can events have access modifiers?  
Can we have shared events?  
What is shadowing?  
What is the difference between Shadowing and Overriding?  
If we inherit a class do the private variables also get inherited?  
How can we stop the class from further inheriting?  
What is the use of “Must inherit” keyword in VB.NET?  
What are similarities between Class and structure?  
What is the difference between Class and structure’s?  
What does virtual keyword mean?  
What are shared (VB.NET)/Static(C#) variables?  
What is ENUM?  
What is nested Classes?  
If you create the child class object which constructor will fire first?  
In what instances you will declare a constructor to be private?  
Can we have different access modifiers on get/set methods of a property?  
If we put goto / return statement in try / catch block will the finally code execute?  
What is Indexer?  
Can we have static indexer in C#?  
Section 4:    ADO.NET  
What are the different components in ADO.NET?  
What is the namespace in which .NET has the data functionality class?  
When should we use System.Data.SqlClient and System.Data.OleDB ?  
What is difference between dataset and data reader?  
What is the use of command objects?  
What are Dataset objects?  
What is the use of data adapter?  
What are basic methods of Data adapter?  
How can we fire a simple SQL Statement using ADO ?  
How do we use stored procedure in ADO.NET and how do we provide parameters to the stored procedures?  
How can we force the connection object to close after my data reader is closed?  
I want to force the data reader to return only schema of the data store rather than data.  
How can we fine-tune the command object when we are expecting a single row?  
Which is the best place to store connection string in .NET projects?  
How do you fill the dataset?  
What are the various methods provided by the dataset object to generate XML?  
How can we save all data from dataset?  
How can we check that some changes have been made to dataset since it was loaded?  
How can we add/remove row is in “Data Table” object of “Dataset”?  
What is basic use of “Data View”?  
What is the difference between “Dataset” and “Data Reader”?  
How can we load multiple tables in a Dataset?  
How can we add relation between tables in a Dataset?  
What is the use of Command Builder?  
What’s difference between “Optimistic” and “Pessimistic” locking?  
How many ways are there to implement optimistic locking in ADO.NET?  
How can do pessimistic locking?  
How can we perform transactions in .NET?  
What is difference between Dataset.Clone and Dataset.Copy?  
Can you explain the difference between an ADO.NET Dataset and an ADO Record set?  
Explain in detail the fundamental of connection pooling?  
What is Maximum Pool Size in ADO.NET Connection String?  
How to enable and disable connection pooling?  
What are  the major differences between classic ADO and ADO.NET?  
Section 5:    ASP.NET  
Can you explain ASP.NET page life cycle?  
What are Httphandlers and HttpModules?  
What is the difference between Httphandlers and HttpModules?  
How do we write a Httphandler ?  
How do we write anHttpModule?  
Can you explain how ASP.NET application life cycle works?  
In which event are the controls fully loaded?  
How can we identify that the Page is Post Back?  
What is the use of @ Register directives?  
What is the use of Smart Navigation property?  
What is AppSetting Section in “Web.Config” file?  
Where is View State information stored?  
How can we create custom controls in ASP.NET?  
How many types of validation controls are provided by ASP.NET?  
How can we force all the validation control to run?  
How can we check if all the validation control are valid and proper?  
If client side validation is enabled, will server side code still run ?  
Which JavaScript file is referenced for validating the validators at the client side?  
How to disable client side script in validators?  
How can I show the entire validation error message in a message box on the client side?  
If a validation is  very complex what will you do ?  
Can you explain “AutoPostBack”?

How can you enable automatic paging in Data Grid?  
What is the use of “GLOBAL.ASAX” file?  
What is the difference between “Web.config” and “Machine.Config”?  
What is a SESSION and APPLICATION object?  
What is the difference between ‘Server.Transfer’ and ‘response.Redirect’ ?  
What is the difference between Authentication and authorization?  
What is impersonation in ASP.NET?  
What are the various ways of authentication techniques in ASP.NET?  
Can you explain Forms authentication in detail?  
How do I sign out in forms authentication?  
If cookies are disabled how will forms authentication work?  
How do we implement windows authentication?  
How can we do single sign on in ASP.NET?  
Can you explain membership and role providers in ASP.Net 2.0?  
Can you explain master pages concept in ASP.NET?  
So how do you create master pages?  
What is the concept of Web parts?  
What are partial classes in ASP.NET ?  
What is the difference between data grid and grid view?  
What is difference between Grid view, Data list, and repeater?  
From performance point of view, how do they rate?  
What is the method to customize columns in Data Grid?  
How can we format data inside Data Grid?  
How to decide on the design consideration to take a Data grid, data list, or repeater?  
What are major events in GLOBAL.ASAX file?  
How can we kill a user session?  
How do you upload a file in ASP.NET?  
How do I send email message from ASP.NET?  
What are different IIS isolation levels?  
ASP used STA threading model, what is the threading model used for ASP.NET.  
What is the use of <%@ page aspcompat=true %> attribute?  
Explain the differences between Server-side and Client-side code?  
How to use a checkbox in a data grid?  
What is the difference between “Web farms” and “Web garden”?  
How do we configure “Web Garden”?  
What’s the difference between trace and debug in ASP.NET?  
How do you enable tracing in on an ASP.NET page?  
Which namespace is needed to implement debug and trace ?  
Can you explain the concept of trace listener?  
What aretrace switches?  
What is an application object?  
What is the use of cache object?  
What is the difference between Cache object and application object?  
How can get access to cache object?  
What are dependencies in cache and types of dependencies?  
Can you show a simple code showing file dependency in cache?  
What is Cache Callback in Cache?  
What is scavenging?  
What are different types of caching using cache object of ASP.NET?  
How can you cache different version of same page using ASP.NET cache object?  
How will implement Page Fragment Caching?  
Can you compare ASP.NET sessions with classic ASP?  
Which are the various modes of storing ASP.NET session?  
Do session use cookies?  
Is Session\_End event supported in all session modes?  
Where do you specify session state mode in ASP.NET?  
What are the other ways you can maintain state?  
What are benefits and Limitation of using Hidden fields?  
What is ViewState?  
How do we ensure viewstate is not tampered?  
Does the performance for viewstate vary according to User controls?  
What are benefits and Limitation of using Viewstate for state management?  
How can you use Hidden frames to cache client data ?  
What are benefits and limitations of using Hidden frames?  
What are benefits and limitations of using Cookies?  
What is Query String and what are benefits and limitations of using Query Strings?  
What is Absolute and Sliding expiration?  
What is cross page posting?  
How do we access viewstate value of the current page in the next page ?  
Can we post and access view state in another ASP.NET page?  
What is SQL Cache Dependency in ASP.NET 2.0?  
How do we enable SQL Cache Dependency in ASP.NET 2.0?  
What is Post Cache substitution?  
Why do we need methods to be static for Post Cache substitution?  
How do we encrypt web.config files in ASP.NET 2.0 ?  
In .NET 1.X how was the encryption implemented for config files?  
How do you send a email using ASP.NET ?  
How did you deployment and setup in ASP.NET ?  
Section 6:   MVC (Model view controller)  
What is MVC?  
Can you explain the complete flow of MVC?  
Is MVC suitable for both windows and web application?  
What are the benefits of using MVC?  
Is MVC different from a 3 layered architecture?  
What is the latest version of MVC?  
What is the difference between each version of MVC?  
What are routing in MVC?  
Where is the route mapping code written?  
Can we map multiple URL’s to the same action?  
How can we navigate from one view to other view using hyperlink?  
How can we restrict MVC actions to be invoked only by GET or POST?  
How can we maintain session in MVC?  
What is the difference between tempdata ,viewdata and viewbag?  
What are partial views in MVC?  
How did you create partial view and consume the same?  
How can we do validations in MVC?  
Can we display all errors in one go?  
What are the other data annotation attributes for validation in MVC?  
How can we enable data annotation validation on client side?  
What is razor in MVC?  
Why razor when we already had ASPX?  
So which is a better fit Razor or ASPX?  
How can you do authentication and authorization in MVC?  
How to implement windows authentication for MVC?  
How do you implement forms authentication in MVC?  
How to implement Ajax in MVC?  
What kind of events can be tracked in AJAX ?  
Figure 6.9:- tracked in AJAX  
What is the difference between “ActionResult” and “ViewResult”?  
What are the different types of results in MVC?  
What are “ActionFilters”in MVC?  
Can we create our custom view engine using MVC?  
Figure 6.12:- output  How to send result back in JSON format in MVC?  
Figure 6.13:- JSON format in MVC   What is “WebAPI”?  
But WCF SOAP also does the same thing, so how does “WebAPI” differ?  
With WCF also you can implement REST,So why “WebAPI”?  
How to implement “WebAPI” in MVC?  
Section 7:   SQL SERVER  
What is normalization? What are different types of normalization?  
What is denormalization?  
What are the different types of joins? What is the difference between them?  
What is a candidate key?  
What are indexes and what is the difference between clustered and nonclustered?  
How can you increase SQL performance?  
What is DTS?  
What is fill factor ?  
What is RAID and how does it work?  
What is the difference between DELETE and TRUNCATE TABLE?  
If locking is not implemented, what issues can occur?  
What are different transaction levels in SQL SERVER?  
What are the different locks in SQL SERVER?  
Can we suggest locking hints to SQL SERVER?  
What is LOCK escalation?  
What are the different ways of moving data between databases in SQL Server?  
What is the difference between a HAVING CLAUSE and a WHERE CLAUSE?  
What is the difference between UNION and UNION ALL SQL syntax?  
How can you raise custom errors from stored procedure?  
What is ACID fundamental?  
What is DBCC?  
What is the purpose of Replication?  
What are the different types of replication supported by SQL SERVER?  
What is BCP utility in SQL SERVER?  
What are the different types of triggers in SQl SERVER?  
If we have multiple AFTER Triggers , can we specify sequence ?  
What is SQL injection?  
What is the difference between Stored Procedure and User Defined Function?  
Section 7: Remoting, Web services and WCF  
What is the Web services ,remoting and WCF ?  
What is an application domain?  
What is .NET Remoting?  
Which class does the remote object has to inherit?  
What are two different types of remote object creation mode in .NET remoting?  
What are the basic steps to implement remoting?  
What are drawbacks of remoting and how can we overcome the same?  
What is a Web Service?  
What’s the difference between web services and remoting?  
What is UDDI?  
What is DISCO?  
What is WSDL?  
What are the steps to create a web service and consume it?  
How do we secure a web service?  
Does web service have state?  
What is SOA?  
What’s the difference between WCF and Web services?  
What are end point, contract, address, and bindings?  
What are the main components of WCF?  
What is a service contract, operation contract and Data Contract?  
What are the various ways of hosting a WCF service?  
How do we host a WCF service in IIS?  
What are the advantages of hosting WCF Services in IIS as compared to self-hosting?  
What are different bindings supported by WCF?  
What is the difference between BasicHttpBinding and WsHttpBinding ?  
Can we overload WCF service methods and functions?  
What is a one-way operation?  
In one way contract we do not get call back, how can we solve the same?  
How can we host a service on two different protocols on a single server?  
How can we integrate with WCF services with MSMQ?  
How can we do security in WCF services?  
In what scenarios will you use message security and transport security?  
Where do we specify security option in WCF services?  
What are the different ways of doing WCF concurrency?  
What are different ways of doing WCF instancing?  
What is REST?  
How can we make WCF rest enabled?  
Can we call two WCF services in one transaction?  
How can we enable debugging and tracing on WCF services?  
How are exceptions thrown in WCF?  
What is the difference between WCF fault exceptions and .NET exceptions?  
Section 8: WPF andSilverligh  
What is WPF?  
What is the need of WPF when we had windowsforms?  
How does hardware acceleration work with WPF?  
Does that mean WPF has replaced DirectX?  
So can we define WPF in a precise way?  
What is XAML?  
So is XAML meant only for WPF ?  
Can you explain the overall architecture of WPF?  
Which are the different namespaces and classes in WPF ?  
What is App.xaml in WPF project?  
What are dependency properties?  
Are XAML file compiled or built on runtime?  
Can you explain how we can separate code and XAML?  
How can we access XAML objects in behind code?  
What is SilverLight?  
Can SilverLight run in other platforms other than window?  
Come on, even WPF runs under browser why SilverLight ?  
Can SilverLight run in other platforms other than window?  
What is the relationship between Silver Light, WPF and XAML?  
What is XAP file in Silverlight?  
Can you explain Sliver Light architecture?  
What are the various basic steps to make a simple Silver Light application?  
What are the different kinds of bindings in Silverlight?  
How does Silverlight connect with databases?  
What are the 2 important points we noted when we call WCF service from Silverlight?  
What are the different ways of doing alignment in Silverlight and WPF?  
What is expression blend?  
Section 9: LINQand Entity framework  
Define LINQ?  
How does LINQ help us from the perspective of business objects?  
Can you explain how a basic LINQ Query looks like?  
How do we write a LINQ query to search with criteria?  
How can do a join using LINQ query?  
How can we do a group by using LINQ query?  
What are entity classes in LINQ?  
How can we load the LINQ entity  class ?  
How do we define 1 to many and many to 1 relationship in LINQ?  
How can we call a stored procedure using LINQ ?  
How can we insert, update and delete using LINQ?  
What are DBML files in LINQ?  
What is Entity framework?  
What’s the difference between LINQ to SQL and Entity framework?  
What are CSDL, SSDL and MSL?  
What is the work of EDMX file?  
How can we browse using entity framework classes?  
How can we add using EF?  
How can we use stored procedures in entity frame work?  
What are POCO classes in Entity framework?  
Section 10:    Design patterns,UML, Estimation and Project management  
What are design patterns?  
Which design patterns have you used in your project?  
Can you explain singleton pattern?  
Can you explain Façade pattern?  
What is MVC, MVP and MVVM pattern?  
What is MVC pattern?  
How can we implement MVC in ASP.NET?  
What is MVP?  
What is MVVM?  
What is the difference between MVC, MVP and MVVM and when to use what?  
What is three-tier architecture?  
Have you ever worked with Microsoft Application Blocks, if yes then which?  
What is Service Oriented architecture?  
What are different ways you can pass data between tiers?  
What is UML?  
How many types of diagrams are there in UML?  
What are advantages of using UML?  
How did you implement UML in your project?  
What are different phases in a software life cycle?  
Can you explain different software development life cycles?  
What does Agile mean?  
What is SCRUM?  
What does product owner, product back log and sprint mean in SCRUM?  
Can you explain how SCRUM flows?  
Can you explain different roles in SCRUM?  
When should we choose Agile and when should we choose waterfall?  
What are some of the important metrics in project?  
What is effort variance?  
What is CAR (Causal Analysis and Resolution)?  
What is DAR (Decision Analysis and Resolution)?  
What is a fish bone diagram?  
What is Pareto principle?  
How do you handle change request?  
What is internal change request?  
What is difference between SITP and UTP in testing?  
Which software have you used for project management?  
People in your project do not perform, what will you do?  
What is black box testing and White box testing?  
What is the difference between Unit testing, Assembly testing and Regression testing?  
What is V model in testing?  
How do you start a project?  
How did you do resource allocations?  
How will you do code reviews?  
What is CMMI?  
What are the five levels in CMMI?  
What is SIX sigma?  
What are DMAIC and DMADV?  
What are the various ways of doing software estimation ?  
What is function point estimation?  
How did you estimate by using function points ?  
What is the FP per day in your current company?  
What is SMC approach of estimation?  
How do you estimate maintenance project and change requests?  
Section 11:     Ajax  
What problem does Ajax solve?  
What is Ajax?  
What is the fundamental behind Ajax?  
How do we use XMLHttpRequest object in JavaScript?  
Can you explain Scriptmanager control in Ajax?  
What is the use of update panel inAjax?  
How do we consume web service in Ajax?  
Can you explain the concept of triggers in ‘UpdatePanel’ control?  
Can you explain the ‘UpdateProgress’ component?  
How can you do validations in Ajax?  
How do we do exception handling in Ajax?  
What is JSON?  
Do all technologies support JSON?  
Section 12:  Reports  
How do we access crystal reports in .NET?  
What are the various components in crystal reports?  
What basic steps are needed to display a simple report in crystal?  
Can crystal reports be published as a web service?  
How do we invoke the crystal report web service?  
How do we add formulas using crystal reports?  
How do we pass parameters to crystal reports?  
How do we export from crystal reports?  
How do we print to printer using crystal?  
How do we generate cross tab reports?  
How can we do grouping in crystal?  
Can you explain three-pass reporting which crystal report uses?  
Can you explain reporting services architecture?  
We have two IIS application ‘Reports’ and ‘Reportserver’ what do they do ?  
Can you explain Report definition language (RDL) file in reporting services?  
What is the basic process of making a report in reporting services?  
How can we consume reports in ASP.NET?  
Can you explain the difference between private and shared data sources?  
How does reports caching in reporting services work ?  
What are the major differences between Crystal and SQL reporting services?  
Section 13:     Threading  
What is Multi-tasking?  
What is Multi-threading?  
What is a Thread?  
Did VB6 support multi-threading?  
Can we have multiple threads in one App domain?  
Which namespace has threading?  
Can you explain in brief how can we implement threading?  
How can we change priority and what the levels of priority are provided by .NET?  
What does Address Of operator do in background?  
How can you reference current thread of the method?  
what is Thread.Sleep () in threading?  
How can we make a thread sleep for infinite period?  
What is Suspend and Resume in Threading?  
What the way to stop a long running thread?  
How do I debug thread?  
What is Thread.Join () in threading?  
What are Daemon threads and how can a thread be created as Daemon?  
How is shared data managed in threading?  
Can we use events with threading?  
How can we know a state of a thread?  
What is use of Interlocked class ?  
What is a monitor object?  
What are wait handles?  
What isManualResetEvent and AutoResetEvent?  
What is Reader Writer Locks?  
How can you avoid deadlock in threading?  
What is the difference between thread and process?  
Section 14:    XML  
What is XML?  
What is the version information in XML?  
What is ROOT element in XML?  
If XML does not have closing tag will it work?  
Is XML case sensitive?  
What is the difference between XML and HTML?  
Is XML meant to replace HTML?  
Can you explain why your project needed XML?  
What is DTD (Document Type Definition)?  
What is well formed XML?  
What is a valid XML?  
What is CDATA section in XML?  
What is XSL?  
What is element and attributes in XML?  
Which are the namespaces in .NET used for XML?  
What are the standard ways of parsing XML document?  
In What scenarios will you use a DOM parser and SAX parser?  
How was XML handled during COM times?  
What is the main difference between MSML and .NET Framework XML classes?  
What are the core functionalities in XML .NET framework? Can you explain in detail those functionalities?  
What is XSLT?  
Define XPATH?  
What is the concept of XPOINTER?  
What is an XMLReader Class?  
What is XMLTextReader?  
How do we access attributes using “XmlReader”?  
Explain simple Walk through of XmlReader?  
What does XmlValidatingReader class do?  
Section 15:    NET Interoperability  
How can we use COM Components in .NET?  
We have developed the COM wrapper do we have to still register the COM?  
How can we use .NET components in COM?  
How can we make Windows API calls in .NET?  
When we use windows API in .NET is it managed or unmanaged code?  
What is COM?  
What is Reference counting in COM?  
Can you describe IUKNOWN interface in short?  
Can you explain what DCOM is?  
How do we create DCOM object in VB6?  
How to implement DTC in .NET?  
How many types of Transactions are there in COM + .NET?  
How do you do object pooling in .NET?  
What are types of compatibility in VB6?  
What is equivalent for regsvr32 exe in .NET?  
Section 16:      Windows workflow Foundation  
What is Windows Workflow Foundation?  
What is a Workflow?  
What are different types of Workflow in Windows Workflow foundation?  
when should we use a sequential workflow and when should we use state machines?  
How do we create workflows using designer?  
How do we specify conditions in Work flow?  
How do you handle exceptions in workflow?  
What is the use of XOML files?  
How can we pass parameters to workflow?

<https://dotnetinterviewquestion.wordpress.com/2013/02/11/how-to-prepare-for-c-and-netinterviews/>