Interview Questions & Answers on basics of .NET

1) What is Namespace?

Namespace does two basic functionalities:-

- **It logically groupsclasses, for instance System.Web.UI logically groups UI related features like textboxes, list control etc.**
- **In Object Oriented world, many times it is possible that programmers will use the same class name. Qualifying NameSpace with class names avoids this collision.**

2) What is Difference between NameSpace and Assembly?

Following are the differences between namespace and assembly:

- **Assembly is physical grouping of logical units, Namespace, logically groupsclasses.**
- **Namespace can span multiple assemblies while assembly is a physical unit like EXE , DLL etc.**

3) What is ILDASM?

ILDASM is a simple tool which helps you to view IL code of a DLL or EXE. In order to view IL code using ILDASM , go to visual studio command prompt and run "ILDASM.EXE". Once ILDASM is running you view the IL code.

4) What is Manifest?

Assembly metadata is stored in Manifest. Manifest contains metadata which describes the following things :-

- **Version of assembly.**
- **Security identity.**
- **Scope of the assembly.**
- **Resolve references to resources and classes.**

The assembly manifest is stored in the DLL itself.

5) Where is the version information stored of an assembly?

Version information is stored in assembly inside the manifest.

6) Is versioning applicable to private assemblies?

Yes, versioning is applicable to private assemblies also.

Interview Questions & Answers on WCF (Windows Communication Foundation)

1) **Tell me something about PerSession instance context mode in WCF?**

   When we set instance context mode as PerSession in WCF, for every channel object or proxy object client creates a dedicated service instance will be created.

2) **What happens by that?**

   Our services become state full. Values of the variables updated during subsequent service call will be maintained.

3) **Is it the default instance context mode?**

   Yes it is, considering binding supports session full endpoints. For example when your binding is netTcpEndpoint or wsHttpBindingPerInstance is default but when its basicHttpBinding default one will be percall because PerInstance is not supported over there?

4) **Does WsHttpBinding supports sessions all the time?**

   No, it supports only if security is enabled. Best point is in case of WSHttpBinding security is be default turned on.

5) **What if I don't want security but wants session?**

   You have to enable reliable messaging at least. Reliable messaging is not enabled by default.

6) **How the server differentiates between different sessions in case of per instance?**

   It will be with the help of sessionId. SessionId will be created for a client when he create the proxy or channel instance for first time.

7) Is it possible to get current session Id in service?

Yes, using OperationContext.Current.SessionId

8) How security works in WCF?

In WCF security is by default enabled in wsHttpBinding and NetHttpBinding. Enabled means all three aspects of security (CIA) is automatically provided by bindings.
Let's understand how.

- **C – Confidentiality – Data can be viewed by only intended recipient.**
  It is done by means of encryption. Data (Messages) to be transferred will be encrypted.
  Even if someone get the message he will not be able to understand it.
- **I – Integrity – Data should not be tempered.**
  It is done by means of signing. Data (messages) to be transferred will be signed.
  If someone modifies the data in between receiver easily come to know about that.
- **A – Authentication – User should validate his/her identity.**
  By default all the binding except basic provides windows authentication. It means every service call will be authenticated automatically.
  While making call automatically clients windows credentials will be passed along which will be automatically get validated in service end.
  In case he is not a valid AD – Active directory user, his/her request won't get processed.

9) How to invoke WCF service asynchronously?

As a service developer we just simply create the service, host it and expose the endpoints. Now whether those functions should be called synchronous or asynchronously is client's lookout.

When the proxy class of our service get generated in the client side, async version of all our service methods also get generated automatically.
For example : If our service contain methods such as GetString, GetCustomer then our proxy class will contain four methods,
GetString,GetCustomer,GetStringAsync,GetCustomerAsync.

There are two approaches normally a .net developer can follow,

- **Event based approach**
- **Task based approach**

Event based approach

Both the asynchronous methods will never return anything. Return type will be void. Instead on completion, some events will be raised.

Complete code:

```
privatevoid button1_Click(object sender, EventArgs e)
{
  FirstClient c = newFirstClient("WSHttpBinding_IFirst");
  c.GetStringCompleted += c_GetStringCompleted;
  c.GetStringAsync();
  .
  .
}
.
.
.
.
voidc_GetStringCompleted(object sender, GetStringCompletedEventArgs e)
{
label2.Text = e.Result;
}
```

Task based approach

In this approach C# 5.0 asycn await pattern will be used. Here asynchronous methods will return “Task” object instead of direct return value which can be awaited later

```
privateasyncvoid button1_Click(object sender, EventArgs e)

{

 FirstClient c = newFirstClient("WSHttpBinding_IFirst");
Task<string> r=c.GetStringAsync();
.
.
.
label2.Text += "\n" + await r;
}
```

Describe the collection framework.

Ans. The .NET supports several types of collections. All the collection classes are available in System. Collections and System.Collections.Specialized namespaces.

Sr.No	Types of Collections	Description
1.	ArrayList	The ArrayList collection is a simplest collection of unordered items.The Add and AddRangemethods of the ArrayListare used to add items.Add method takes object type as a parameter.
2.	SortedList	It is an dictionary object that gives you the sorted list of items.
3.	Queue	It uses the Enqueue method to add items to the queue and the Dequeuemethod to remove items from the list.It works on FIFO principle.
4.	Stack	It uses pop method to retrieves an item from the top of the stack and push method to add an item to the top of the stack. It works on LIFO principle.
5.	Hashtable	It is an dictionary object and works on key,valuepair.The add method takes two parameter the first one is key and the second one is value object. DictionaryEntry class is used to iterate all the items.HashTable is very efficient for large collection.
6.	BitArray	It is a resizeable collection that can store Boolean values. it supports bit-level operations such as and, not, or, and exclusive-or .
7.	StringCollection	It is same as ArrayList but can store only string values.
8.	StringDictionary	It is same as HashTable but can store only string key value pair.
9.	ListDictionary	It is very efficient for small collections of items
10.	HybridDictionary	It is the combination of HashTable and ListDictionary.If the no of items are less then it work as ListDictionary otherwise it works as HashTable.
11.	NameValueCollection	In this class you can store multiple values per key by using add method. For retrieving all the values for a particular key, you can use theGetValuesmethod.