WCF (Windows Communication Foundation)

1) What is WCF?

- ✓ WCF stands for Windows Communication Foundation.
- ✓ WCF is an application programming interface (API) in the .NET framework for developing applications based on service-oriented architecture (SOA).
- ✓ WCF is introduced in .NET 3.0. It is derived from **System.ServiceModel** namespace.
- ✓ **Interoperability** is the fundamental characteristics of WCF.
- ✓ WCF is a combined feature of **Web Service, Remoting, MSMQ** and **COM+.**

2) What is endpoint in WCF service?

- ✓ **Endpoint** defines how a client will communicate with the service.
- ✓ **Endpoint** is used to identify the service. All the WCF communications are take place through **end point**.
- ✓ It consists of three main points: **Address, Binding** and **Contract**.
- ✓ It is a relationship between **Address, Binding** and **Contract**.

3) Explain Address, Binding and contract for a WCF Service?

Address (where) => Defines where the service resides (or) hosted. Basically it is an URL. Client will use this URL to connect to the service.

Binding (how) => Defines how to communicate with the service (or) how to access the service.

Contract (what) => Defines what can be done with the service (or) what the service will do.

Usually name of the Interface will be mentioned in the Contract. So, the client application will be aware of the operations which are exposed to the client.

4) What are the various address formats in WCF?

- ✓ **HTTP Address Format =>** http://localhost:
- ✓ **TCP Address Format =>** net.tcp://localhost:
- ✓ **MSMQ Address Format =>** net.msmq://localhost:

5) What are the types of binding available in WCF?

- ✓ BasicHttpBinding
- ✓ NetTcpBinding
- ✓ WSHttpBinding
- ✓ NetMsmqBinding

6) What are the types of contract available in WCF (or) describe which operations the client can perform on the service?

Service Contract	This attribute is used to define the Interface.
Operation Contract	This attribute is used to define the methods inside Interface.
Data Contract	Data Contract is used to describe the data types used by a service. Interoperability is possible through this.
Message Contract	Using Message Contract, a service can interact directly with messages. It is used to control the structure of a message body and serialization process. It is also used to send / access information in SOAP headers.
Fault contracts	Define which errors are raised by the service and how the service handles.

7) What are the various ways of hosting a WCF Service?

- ✓ IIS
- ✓ Self-Hosting
- ✓ WAS (Windows Activation Service)

8) What is the proxy for WCF Service?

A proxy is a class by which a service client can interact with the service. Using proxy we can call the different methods exposed by the service.

9) How can we create Proxy for the WCF Service?

We can create proxy using the tool **svcutil.exe** after creating the service. We can use the following command at command line.

svcutil.exe *.wsdl *.xsd /language:C# /out:SampleProxy.cs /config:app.config.

10) What is the difference between WCF Service and Web Service?

- ✓ WCF Service supports both **HTTP** and **TCP** protocol while web service supports only **HTTP** protocol.
- ✓ WCF Service is more flexible than web service.

11) Which namespace is used to access WCF service?

System.ServiceModel

12) What is a SOA Service?

Service Oriented Architecture (SOA) is architecture for building applications using reusable, interoperable services. **SOA** service is composed of three parts.

- ✓ **Service** => This class implements some services as a set of methods.
- ✓ **Environment =>** It is used to host the service.
- ✓ **Endpoints** => All communications with the WCF service will happen via the endpoints.

13) Which namespace is required to use [DataContract] or [DataMember] attribute for a class or properties?

Using **System.Runtime.Serialization;**

14) In WCF, Which contract is used to document the errors occurred in the service to client?

Fault Contract is used to document the errors occurred in the service to client.

15) What is the default Exchange Pattern (MEP)?

Request/Response.

16) What is the Messaging Pattern? Which Messaging Pattern WCF supports?

Messaging patterns describes how client and server should exchange the message.

There is a protocol between client and server for sending and receiving the message. These are also called **Message Exchange Pattern.**

WCF supports following 3 types of Message Exchange Patterns

- ✓ **request - reply (default message exchange pattern)**
- ✓ **OneWay (Simplex / datagram)**
- ✓ **Duplex(CallBack)**

17) What is XML Infoset?

The XML Infoset defines a data model for XML. It can be used to describe a valid XML document.

18) What is DataContractSerializer in WCF?

- ✓ **DataContractSerializer** is new WCF serializer. This is serialization engine in WCF.
- ✓ **DataContractSerializer** translate the .NET framework objects into XML and vice-versa.
- ✓ By default WCF uses **DataContractSeriazer**.

19) What is .svc file in WCF?

.SVC file is a text file. This file is similar to ours **.ASMX** file in web services. This file contains the details required for WCF service to run it successfully.

This file contains following details,

- ✓ Language (C# / VB)
- ✓ Name of the service
- ✓ Where the service code resides

Example

<%@ ServiceHost Language="C#/VB" Debug="true/false" CodeBehind="Service code files path" Service="ServiceName" %>

20) What is Message Contract in WCF?

- ✓ Using Message Contract, a service can interact directly with messages.
- ✓ Message Contract controls the SOAP messages sent and received by the client and server.
- ✓ It is used to control the structure of a message body and serialization process.
- ✓ It is also used to send/ access information in SOAP headers.

21) What is Fault Contracts in WCF?

Fault Contracts is the way to handle exceptions in WCF.

22) Which protocol is used for platform-independent communication?

SOAP (Simple Object Access Protocol), which is directly supported from WCF.

23) Which bindings in WCF support the message streaming?

- ✓ BasicHttpBinding
- ✓ NetTcpBinding
- ✓ NetNamedPipeBinding.

24) What are the 3 types of transactions manager WCF supports?

Lightweight	Used to manage the transaction within same application domain. Best Performance compare to other.
OLE Transactions	Manage transaction in Internet and windows environment.
WS-Atomic Transactions	It can propagate transactions across firewalls.

25) What is the default mode for Instancing in WCF?

PerCall

26) What is service host factory in WCF?

- ✓ **Service host factory** is the mechanism by which we can create the instances of service host dynamically as the request comes in.
- ✓ This is useful when we need to implement the **event handlers** for opening and closing the service. WCF provides **Service Factory** class for this purpose.

27) Can we overload methods in WCF Service or Web Service?

Yes

- ✓ For a **WCF Service** use the **Name** property of **OperationContract** Attribute class.
- ✓ For a **Web Service** use the **MessageName** property of **WebMethod** Attribute class.

28) Can we have a [Message contract] as input parameter and [Data contract] as returned parameter in a single [operation contract]?

No. if we want to use **[Message contract]** in an **[operation contract]**, input and return parameter should be same as **[Message contract].**

29) If you define additional methods in the WCF service that are not in the service contract, will it be visible to Client applications?

No

30) Which file specifies the name and location of the class that implements the WCF service?

Service.svc

31) What is the former name of WCF?

Indigo

32) How to set the instancing mode in WCF service?

Instancing mode is set in WCF Service on the Service Level using the **InstanceContextMode** Enum It has these values,

- ✓ Single => One service instance is allocated for all client calls.
- ✓ PerCall => One service instance is allocated for each client call.
- ✓ PerSession => One service instance is allocated for each client session.

33) What are the main components of WCF?

The main components of WCF are

- ✓ **Service class**
- ✓ **Hosting environment**
- ✓ **End point**

34) How to set the timeout property for the WCF Service client call?

The **timeout** property can be set for the WCF Service client call using **binding** tag. If no timeout has been specified, the default is considered as **one minute**.

35) What is the main disadvantage of using IIS to host a service?

Using **IIS** to host your services means that you will not be able to support **non-HTTP** protocols such as **TCP, named pipes** and **MSMQ**. You will have access to the many built-in features available with IIS such as **process recycling** and **message based activation**.

36) What is the utility name to test WCF services?

WcfTestClient.exe

It resides in the folder **C:\Program Files\Microsoft Visual Studio 9.0\Common7\IDE**

37) What is the application utility name to view logs in WCF?

eventvwr.exe

38) What is Services?

WCF applications expose functionality through **Services**.

Service is a **Common Language Runtime (CLR)** type that encapsulates business functionality and exposes a set of methods that can be accessed by remote clients.

39) What is the use of Proxy and Channels?

Proxy => Using Proxy Client can communicate with Services in WCF.

Channels => facilitate communication between Clients and Services in WCF.

40) What is ServiceBehaviour and ServiceMetadata node?

ServiceBehaviour => It is used to publish the metadata using **HTTP-GET** and Instance mode (Singleton, Per-call, Per-Session) can be configured using **ServiceBehavior** attribute.

ServiceMetadata node is used to expose metadata with **httpGetEnabled="true".**

41) Difference between WCF and Web service?

Web service	WCF
It can be hosted in IIS	It can be hosted in IIS, windows activation service (WAS), Self-hosting, Windows service.
Web service supports only **HTTP** protocol.	**WCF Service** supports both **HTTP and TCP** protocol.
[WebService] attribute has to be added to the class.	**[ServiceContraact]** attribute has to be added to the class.
[WebMethod] attribute represents the method exposed to client.	**[OperationContract]** attribute represents the method exposed to client.
Webservice use **SOAP** and **Xml** for bindings.	Supports different types of bindings like ✓ BasicHttpBinding ✓ WSHttpBinding ✓ WSDualHttpBinding
One-way and **Request- Response** are the different operations supported in web service.	**One-Way, Request-Response, Duplex** are different type of operations supported in WCF.
System.Xml.serialization namespace is used for serialization.	**System.Runtime.Serialization** namespace is used for serialization.

42) What is HTTP_GET Enabled Metadata?

We will use **ServiceBehaviour** to publish the **metadata** using **HTTP-GET**. This can be configures either administratively (or) programmatically. In which, we have to set **httpGetEnabled="true"** in **ServiceMetadata** node.

Example:

```
<behaviors>
    <serviceBehaviors>
    <behavior name="ServiceBehavior">
        <!-- Setting httpGetEnabled you can publish the metadata -->
        <serviceMetadata httpGetEnabled="true"/>
        </behavior>
        </serviceBehaviors>
   </behaviors>
```

43) Consider a scenario say, I am creating a service that has to be used by two type of client. One of the clients will access SOAP using HTTP and other client will access Binary using TCP. How it can be done?

With **Web service** it is very difficult to achieve, but in **WCF** it's just we need to add extra **endpoint** in the **configuration** file.

Example:

```
<system.serviceModel>
    <services>
      <service name="MathService"
        behaviorConfiguration="MathServiceBehavior">
      <endpoint address="http://localhost:8090/MyService/MathService.svc"
                Contract="IMathService" binding="wsHttpBinding"/>
      <endpoint address="net.tcp://localhost:8080/MyService/MathService.svc"
                Contract="IMathService" binding="netTcpBinding"/>
      </service>
    </services>
    <behaviors>
      <serviceBehaviors>
        <behavior name="MathServiceBehavior">
          <serviceMetadata httpGetEnabled="True"/>
          <serviceDebug includeExceptionDetailInFaults="true" />
        </behavior>
      </serviceBehaviors>
    </behaviors>
  </system.serviceModel>
```

44) What is the use of ServiceBehavior attribute in WCF?

ServiceBehaviour attribute is used to specify the **InstanceContextMode** for the WCF Service class.

45) Using which binding we can build WCF REST?

For **WCF REST** we need to use **WebHttpBinding**.

WebhttpBinding is enabled by as shown in the below code snippet.

Example:

```
<endpointBehaviors>
    <behavior name="NewBehavior0">
      <webHttp />
      </behavior>
</endpointBehaviors>
```

46) How the concurrency mode is specified in WCF service?

The **concurrency mode** is specified using the **ServiceBehavior** attribute on the class that implements the service.

Example:

```
[ServiceBehavior(ConcurrencyMode=ConcurrencyMode.Single)]
Public class ServiceClass : IServiceInterface
{
    //Implementation Code
}
```

There are 3 possible values of **ConcurrencyMode** enumeration

- ✓ Single
- ✓ Reentrant
- ✓ Multiple

47) How to set the instancing mode in WCF service?

In WCF, **instancing mode** is set at service level.

Example:

```
//Setting PerSession instance mode
[ServiceBehavior(InstanceContextMode = InstanceContextMode.PerSession)]
class MyService : IMyService
{
   //Implementation goes there
}
```

48) What is Address Header in WCF?

Address Header contains the information which is sent with every request; it can be used by either **end point** service or any intermediate device for determining any routing logic or processing logic.

WCF provides **AddressHeader** class for this purpose.

Example:

```
AddressHeader addressHeader= AddressHeader.CreateAddressHeader("Name of the header",
                                                "Information included in header ");
```

Once the **AddressHeader** instance is created, it can be associated with end point instance as follows,

```
EndpointAddress endpoint = new EndpointAddress (new Uri("http://myserver/myservice"),
                                                addressHeader);
```

49) What are the advantages of hosting WCF service in WAS?

WAS (Windows Activation Service) is a component of IIS 7.0. Following are few advantages,

- ✓ We are not only limited to **HTTP** protocol. We can also use supported protocols like **TCP**, **named pipes** and **MSMQ**.
- ✓ No need to completely install **IIS.** We can only install **WAS** component and keep away the Webserver.

50) Advantages and Disadvantages of Hosting WCF in IIS?

Advantages of Hosting WCF in IIS are,

- ✓ It is a simplified way of deployment and development of hosted services.
- ✓ The main advantage of hosting service in IIS is that, it will automatically launch the host process when it gets the first client request.
- ✓ It uses the features of IIS such as process recycling, idle shutdown, process health monitoring and message based activation.

Disadvantages of Hosting WCF in IIS are,

- ✓ The main disadvantage of using IIS is that, it will support only **HTTP** protocol.
- ✓ Another disadvantage is all the service will run on the same port.

51) What is Self Hosting in WCF?

In web service, we can host the service only in **IIS**, but WCF provides the user to host the service in any application (e.g. console application, Windows form etc.). Here developer is responsible for providing and managing the life cycle of the host process. Service can also be in-process i.e. client and service in the same process.

Note: Host process must be running before the client calls the service, which typically means you have to prelaunch it.

52) What is Windows Service Hosting and Advantages?

Windows Service Hosting is same as hosting the service in **IIS** without message activated.

Advantages of Hosting WCF in Windows service are,

- ✓ The service will be hosted, when system starts.
- ✓ Process life time of the service can be controlled by Service Control Manager for windows service.
- ✓ All versions of Windows will support hosting WCF service.

53) What is WAS (windows activation server) and Advantage?

Windows Activation service is a system service available with Windows vista and windows server 2008. It is available with IIS 7.0 and it is more powerful compared to IIS 6.0 because it supports **Http, TCP** and **named pipes** were IIS 6.0 supports only **Http**. It can be installed and configured separately.

Hosting WCF in Activation service takes many advantages such as,

- ✓ **Process recycling**
- ✓ **Isolation**
- ✓ **Idle time management**
- ✓ **Common configuration system**

WAS hosted service can be created using following steps,

1. Enable WCF for non-http protocols
2. Create WAS hosted service
3. Enable different binding to the hosted service

54) Can we overload methods in WCF Service or Web Service?

Yes.

1) For a WCF Service use the Name property of **[OperationContract]** Attribute class.

Example:

```
[ServiceContract]
Interface ddd
{
    [OperationContract(Name = "one")]
    int calc (int a, int b);

    [OperationContract(Name = "two")]
    double calc (double a, double b);
}
```

2) For a Web Service use the **MessageName** property of **[WebMethod]** Attribute class.

```
Please comment the following line in the .cs file of the Web Service.
// [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
[WebMethod]
public string HelloWorld (string a)
 {
    return "Hello"+" "+a;
 }

[WebMethod (MessageName="second")]
public string HelloWorld ()
 {
    return "Hello second";
 }
```

55) Which of the following members of the class are serialized?

```
[DataContract]
public class Person
{
  [DataMember]
  internal string Name;

  [DataMember]
  private int Age;

  private string Address;    // There is no [DataMember] for this
}
```

1) Name and Address
2) Name and Age
3) All the above
4) None

The answer is **2) Name and Age**

Because, the class members with **[DataAttributes]** are serializable.

For Address, there is no **[DataAttribute]** and hence it is not serializable.

56) What happens if we apply [DataMember] attributes to Static Fields?

Ignored.

Because, if we apply **[DataMember]**attributes to Static Fields, the system won't give any exception but it simple ignores it. The Information provided by an attribute is also known as **metadata**.

57) Which of the following statements are true?

Select from following answers:

1) **The data contract names are case-sensitive**
2) **The data contract namespaces are case-sensitive**
3) **The data member names are case-sensitive**
4) **All the above**

The answer is 4) All the above

Because, when processing **[DataContracts],** the WCF infrastructure is case-sensitive to both the namespaces and the names of **[DataContracts]** and **[DataMembers].**

58) What is InstanceContextMode in WCF?

This can be used to maintain a state of the service (or) a client too.

59) What is Asynchronous Messaging?

Asynchronous messaging describes a way of communications that takes place between two applications (or) systems, where the system places a message in a message queue and does not need to wait for a reply to continue processing.

60) What is [DataContract] equivalence?

Two data contracts are said to be equivalent, if they satisfy the following conditions,

- ✓ Both the **[DataContracts]** must have same namespace
- ✓ Both the **[DataContracts]** must have same name
- ✓ The data member on **[DataContract]** must have an equivalent **[DataMember]** on the other.
- ✓ The members in the **[DataContract]** must appear in the same order in both **[DataContracts].**

The following two data contracts are equal.

```
[DataContract]
public class Person
{
    [DataMember]
    public string Name;

    [DataMember]
    public string Email_ID;
}

[DataContract(Name = "Person")]
public class Employee
{
    [DataMember(Name = "Name")]
    private string EmpName;

    private string address;

    [DataMember(Name = "Email_ID")]
    private string EmpEmailId;

}
```

61) What is "Automatic activation" in WCF?

Automatic Activation means that the service is not necessary to be running in advance. When any message is received by the service it then launches and fulfills the request.

But in case of **self-hosting** the service should always be running.

62) How do we customize Data Contract Names for Generics types?

We can customize the generic DataContract names by allowing parameters.

Example:

```
[DataContract(Name = "Shape_{1}_brush_and_{0}_shape")]
public class Shape< Square,RedBrush>
{
    // Code not shown.
}
```

Here, the DataContract Name is "Shape_RedBrush_brush_and_Square_shape"

{0} – First Parameter in the generic type

{1} – Second Parameter in the generic type.

63) Why in WCF, the "httpGetEnabled" attribute is essential?

The attribute **httpGetEnabled** is essential for publish service metadata.

Example:

<serviceMetadata httpGetEnabled="true" />

Without the metadata, client applications can't generate the proxy and thus won't be able to use the service.

64) How to define a service as REST based service in WCF?

In WCF, we can define a service as **REST** based service using **WebHttpBinding**.

Example:

```
[ServiceContract]
interface IStock
{
  [OperationContract]
  [WebGet]
  int GetStock(string StockId);
}
```

By adding the **[WebGet]** attribute, we can define a service as **REST** based service that can be accessible using **HTTP GET** operation.

65) What are different elements of WCF Services Client configuration file?

WCF Services client configuration file contains **endpoint**, **address**, **binding** and **contract**.

```
<system.serviceModel>
    <client>
        <endpoint name = "MyEndpoint"
                  address  = "http://localhost:8000/MyService/"
                  binding  = "wsHttpBinding"
                  contract = "IMyContract"
        />
    </client>
</system.serviceModel>
```

66) The following Two data contracts are equal.

```
[DataContract(Name = "Maruthi")]
public class Car
{
  [DataMember]
  public int Cost;
  [DataMember]
  public int Bhp;
}

[DataContract(Name = "Maruthi")]
public class Car2
{
  [DataMember(Order = 2)]
  public int Bhp;
  [DataMember(Order = 1)]
  public int Cost;
}
```

Select from following answers:

1) True

2) False

3) None

The answer is 3) None

The **[Data contracts]** are said to be equal if they have

- ✓ Same Namespaces
- ✓ Same Names
- ✓ Same Data Member Names
- ✓ Same Order of the Member Names

Snippet-1:

As order not specified, the default Order is alphabetical (Bhp, Cost).

Snippet-2:

Here the Order is based on the order attribute. Which is (Cost, Bhp)?

As the order of the data member names differs, the above statement is false.

67) How to generate proxy for WCF Services?

The **proxy** can be generated using **Visual Studio** by right clicking **Reference** and clicking on **Add Service Reference**. This brings up the **Add Service Reference dialog box**, where you need to supply the base address of the service (or a base address and a MEX URI) and the namespace to contain the proxy.

Proxy can also be generated by using **SvcUtil.exe** command-line utility. We need to provide **SvcUtil** with the **HTTP-GET** address (or) the metadata exchange endpoint address and, optionally, with a proxy filename. The default proxy filename is **output.cs** but you can also use the /out switch to indicate a different name.

SvcUtil http://localhost/MyService/MyService.svc /out:Proxy.cs

When we are hosting in **IIS** and selecting a port other than port 80 (such as port 88), we must provide that port number as part of the base address:

SvcUtil http://localhost:88/MyService/MyService.svc /out:Proxy.cs

68) How to deal with operation overloading while exposing the WCF services?

By default overload operations (methods) are not supported in **WSDL** based operation. However by using **Name** property of **[OperationContract]** attribute, we can deal with operation overloading scenario.

Example:

```
[ServiceContract]
interface ICalculator
{
    [OperationContract(Name = "AddInt")]
    int Add(int arg1,int arg2);

    [OperationContract(Name = "AddDouble")]
    double Add(double arg1,double arg2);
}
```

Notice that both method names in the above interface is same (Add), however the **Name** property of the **[OperationContract]** is different. In this case client proxy will have two methods with different name **AddInt** and **AddDouble**.

69) What are operations Request-Reply, One-Way, Callback in WCF?

Request-Replay:

By default all WCF will operated in the **Request-Replay** mode. It means that, when client make a request to the WCF service and client will wait to get response from service (till receive Timeout). After getting the response it will start executing the rest of the statement. If service doesn't respond within receive Timeout, client will receive Timeout Exception.

Apart from **NetPeerTcpBinding** and the **NetMsmqBinding** all other bindings will support **request-reply** operations.

One-Way:

In **One-Way** operation mode, client will send a request to the server and does not care whether it is **success** (or) **failure** of service execution. There is no return from the server side, it is **one-way** communication.

Client will be blocked only for a moment till it dispatches its call to service. If any **exception** thrown by service will not reach the server.

One-way operation can be enabled by setting **IsOneWay** property to true in **[Operation contract]** attribute.

Example:

```
[ServiceContract]
public interface IMyService
{
    [OperationContract(IsOneWay=true)]
    void MyMethod(EmployeeDetails emp);
}
```

Callback:

Till now we have seen that the all clients will call the service to get the things done. But WCF also provides the service to call the client. In which, service will act as client and client will act as service.

- ✓ HTTP protocols are connectionless nature, so it is not supported for callback operation. So **BasicHttpBinding** and **WSHttpBinding** can't be used for this operation.
- ✓ WCF support **WSDualHttpBinding** for call back operation.
- ✓ All **TCP** and **IPC** protocols support **Duplex** communication. So all these binding will be used for callback operation.