

70-503 exam questions accompanied by exhibits

Verified Answers Researched by Industry Experts and almost 100% correct

70-503 exam questions updated on regular basis

Same type as the certification exams, 70-503 exam preparation is in multiple-choice questions (MCQs).

Tested by multiple times before publishing

Try free 70-503 exam demo before you decide to buy it in Test-Inside.com.

Note: This pdf demo do not include the question's picture.

Exam : Microsoft 70-503

Title : TS: Microsoft .NET Framework 3.5 C Windows Communication Foundation

1. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. You have successfully defined a service contract named `IManageOrders`.

You write the following code segment.

```
Public Class OrderImpl
Implements IManageOrders
Public Sub MarkOrderClosed(ByVal orderId As Integer) _
Implements IManageOrders.MarkOrderClosed
Try
...
Catch ex As SqlException
Throw New FaultException(Of DataFault)( _
New DataFault())
End Try
End Sub
End Class
<DataContract()> _
Public Class DataFault
End Class
```

You need to create a fault contract for the `MarkOrderClosed` method on the `IManageOrders` service contract.

Which code segment should you add?

- A. `<FaultContract(GetType(DataFault))>`
- B. `<FaultContract(GetType(Exception))>`
- C. `<FaultContract(GetType(SqlException))>`
- D. `<FaultContract(GetType(FaultException))>`

Answer: A

2. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

You need to ensure that the service can send data in the following format to the client applications.

```
<Account Id="">
<Name> </Name>
<Balance Currency=""> </Balance>
</Account>
```

Which code segment should you use?

- A. `[Serializable]`
`public class Account`
`{`
`[XmlAttribute]`
`public string Id;`
`[XmlElement]`

```
public string Name;
[XmlAttribute]
public string Currency;
[XmlElement]
public double Balance;
}
```

```
B. [DataContract]
public class Account
{
[DataMember(Order=0)]
public string Id;
[DataMember(Order=1)]
public string Name;
[DataMember(Order=0)]
public double Balance;
[DataMember(Order=1)]
public string Currency;
}
```

```
C. [Serializable]
public class Account
{
[XmlAttribute]
public string Id;
public string Name;
[XmlElement("Balance")]
public BalanceVal Balance;
}
```

```
[Serializable]
public class BalanceVal
{
[XmlText]
public double Amount;
[XmlAttribute]
public string Currency;
}
```

```
D. [DataContract]
public class Account
{
[DataMember(Order=0)]
public string Id;
[DataMember(Order=1)]
public string Name;
[DataMember(Name="Balance", Order=2)]
public BalanceVal Balance;
}
```

```
[DataContract]
public struct BalanceVal
{
[DataMember(Order=0)]
public double Balance;
[DataMember(Order=1)]
public string Currency;
}
```

Answer: C

3. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. The service will contain an enumeration named OrderState.

The OrderState enumeration will contain the following four values:

Processing
Cancelled
Confirmed
Closed

The client application must be able to set the state of an Order entity to only the following two values:

Cancelled
Closed

You need to create the data contract for OrderState.

Which code segment should you use?

A. `<DataContract()> _`

`Public Enum OrderState`

`Processing = 1`

`<DataMember()> _`

`Cancelled = 2`

`<DataMember()> _`

`Confirmed = 3`

`Closed = 4`

`End Enum`

B. `<DataContract()> _`

`Public Enum OrderState`

`Processing = 1`

`<EnumMember()> _`

`Cancelled = 2`

`Confirmed = 3`

`<EnumMember()> _`

`Closed = 4`

`End Enum`

C. `<DataContract()> _`

`Public Enum OrderState`

`<EnumMember(Value:="False")> _`

`Processing = 1`

`<EnumMember(Value:="True")> _`

`Cancelled = 2`

`<EnumMember(Value:="True")> _`

`Confirmed = 3`

`<EnumMember(Value:="False")> _`

`Closed = 4`

`End Enum`

D. `<DataContract()> _`

`Public Enum OrderState`

`<DataMember()> _`

`Processing = 1`

`<DataMember(IsRequired:=True)> _`

`Cancelled = 2`

`<DataMember()> _`

`Confirmed = 3`

`<DataMember(IsRequired:=True)> _`

`Closed = 4`


```

<OperationContract(> _
Function AddNumbers( _
ByVal a As Integer, ByVal b As Integer) As Integer
Function AddNumbers( _
ByVal a As Double, ByVal b As Double) As Double
End Interface

```

You have not deployed the IMathService service.

You need to expose the AddNumbers (a As Double, b As Double) As Double operation to the IMathService service contract.

Which code segment should you use?

- A. <OperationContract(> _
Function AddNumbers(_
ByVal a As Integer, ByVal b As Integer) As Integer
<OperationContract(> _
Function AddNumbers(_
ByVal a As Double, ByVal b As Double) As Double
- B. <OperationContract(Name:="AddInt")> _
Function AddNumbers(_
ByVal a As Integer, ByVal b As Integer) As Integer
<OperationContract(Name:="AddDouble")> _
Function AddNumbers(_
ByVal a As Double, ByVal b As Double) As Double
- C. <OperationContract(Action:="IMathService/AddInt")> _
Function AddNumbers(_
ByVal a As Integer, ByVal b As Integer) As Integer
<OperationContract(Action:="IMathService/AddDouble")> _
Function AddNumbers(_
ByVal a As Double, ByVal b As Double) As Double
- D. <OperationContract(Action:="AddInt/*")> _
Function AddNumbers(_
ByVal a As Integer, ByVal b As Integer) As Integer
<OperationContract(Action:="AddDouble/*")> _
Function AddNumbers(_
ByVal a As Double, ByVal b As Double) As Double

Answer: B

6. You have created a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

The existing service interface is named IMyService, and contains the following code segment.

```

<ServiceContract(Name:="SvcOrder", _
Namespace:="http://contoso.com/services")> _
Public Interface IMyService
<OperationContract(> _
Sub DoSomething()
End Interface

```

You create a new service named IMyServiceV1 that contains an operation named DoSomethingElse.

You need to ensure that existing client applications are still able to access the IMyService.DoSomething method without modifying client code.

Which code segment should you use?

- A. <ServiceContract(Namespace:="http://contoso.com/services/V1")> _
Public Interface IMyServiceV1
Inherits IMyService
<OperationContract(> _
Sub DoSomethingElse()

End Interface

B. <ServiceContract(Name:="SvcOrder")> _

Public Interface IMyServiceV1

Inherits IMyService

<OperationContract()> _

Sub DoSomethingElse()

End Interface

C. <ServiceContract(Name:="SvcOrderV1", _
Namespace:="http: //contoso.com/services")> _

Public Interface IMyServiceV1

Inherits IMyService

<OperationContract()> _

Sub DoSomethingElse()

End Interface

D. <ServiceContract(Name:="SvcOrder", _
Namespace:="http: //contoso.com/services")> _

Public Interface IMyServiceV1

Inherits IMyService

<OperationContract()> _

Sub DoSomethingElse()

End Interface

Answer: D

7. You create a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

You write the following code segment.

```
[ServiceContract]
```

```
public interface IMathService
```

```
{
```

```
[OperationContract]
```

```
int AddNumbers(int a, int b);
```

```
double AddNumbers(double a, double b);
```

```
}
```

You have not deployed the IMathService service.

You need to expose the AddNumbers (double a, double b) operation to the IMathService service contract.

Which code segment should you use?

A. [OperationContract]

```
int AddNumbers(int a, int b);
```

```
[OperationContract]
```

```
double AddNumbers(double a, double b);
```

B. [OperationContract(Name="AddInt")]

```
int AddNumbers(int a, int b);
```

```
[OperationContract(Name="AddDouble")]
```

```
double AddNumbers(double a, double b);
```

C. [OperationContract(Action="IMathService/AddInt")]

```
int AddNumbers(int a, int b);
```

```
[OperationContract(Action="IMathService/AddDouble")]
```

```
double AddNumbers(double a, double b);
```

D. [OperationContract(Action="AddInt/*")]

```
int AddNumbers(int a, int b);
```

```
[OperationContract(Action="AddDouble/*")]
```

```
double AddNumbers(double a, double b);
```

Answer: B

8. You create a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

The service contains the following code segment.

```
<DataContract> _  
Public Class Person  
...  
End Class  
<DataContract> _  
Public Class Customer  
Inherits Person  
...  
End Class
```

You need to create a service contract that meets the following requirements:

The service contract must have an operation contract named GetPerson that returns an object of type Person.

The GetPerson operation must be able to return an object of type Customer.

Which code segment should you use?

- A. <ServiceContract> _
<ServiceKnownType("GetPerson")> _
Public Interface IMyService
<OperationContract> _
Function GetPerson() As Person
End Interface
- B. <ServiceContract> _
Public Interface IMyService
<OperationContract> _
<ServiceKnownType("Customer")> _
Function GetPerson() As Person
End Interface
- C. <ServiceContract> _
<ServiceKnownType(GetType(Customer))> _
Public Interface IMyService
<OperationContract> _
Function GetPerson() As Person
End Interface
- D. <ServiceContract> _
<ServiceKnownType("GetPerson", GetType(Customer))> _
Public Interface IMyService
<OperationContract> _
Function GetPerson() As Person
End Interface

Answer: C

9. You create a Windows Communication Foundation (WCF) service by using Microsoft .NET Framework 3.5.

You write the following code segment. (Line numbers are included for reference only.)

```
01 Public Interface IMyService  
02  
03 Function ProcessString(ByVal name As String) As String  
04 End Interface
```

You create a host for the WCF service. You also create a service endpoint at `http://localhost:8080/service`. You add an instance of the `HttpTransferEndPointBehavior` class to the host.

You need to ensure that the `ProcessString` method can be invoked from a Web browser by using the URL

`http://localhost:8080/service/process?name=value`

Which code segment should you insert at line 02?

- A. <OperationContract(Name:="process", Action:="Get")> _

B. <OperationContract(Name:="process", Action:="Post")> _

C. <OperationContract()> _

<HttpTransferContract(Path:="process", Method:="Get")> _

D. <OperationContract()> _

<HttpTransferContract(Path:="process", Method:="Post")> _

Answer: C

10. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. The service will contain an enumeration named OrderState.

The OrderState enumeration will contain the following four values:

Processing

Cancelled

Confirmed

Closed

The client application must be able to set the state of an Order entity to only the following two values:

Cancelled

Closed

You need to create the data contract for OrderState.

Which code segment should you use?

A. [DataContract]

```
public enum OrderState
```

```
{
```

```
    Processing=1,
```

```
    [DataMember]
```

```
    Cancelled=2,
```

```
    [DataMember]
```

```
    Confirmed=3,
```

```
    Closed=4
```

```
}
```

B. [DataContract]

```
public enum OrderState
```

```
{
```

```
    Processing=1,
```

```
    [EnumMember]
```

```
    Cancelled=2,
```

```
    Confirmed=3,
```

```
    [EnumMember]
```

```
    Closed=4
```

```
}
```

C. [DataContract]

```
public enum OrderState
```

```
{
```

```
    [EnumMember(Value="False")]
```

```
    Processing=1,
```

```
    [EnumMember(Value="True")]
```

```
    Cancelled=2,
```

```
    [EnumMember(Value="True")]
```

```
    Confirmed=3,
```

```
    [EnumMember(Value="False")]
```

```
    Closed=4
```

```
}
```

D. [DataContract]

```
public enum OrderState
```


}

Answer: C

12. You are creating a Windows Communication Foundation (WCF) service by using Microsoft .NET Framework 3.5. You configure a binding to enable streaming.

You need to ensure that the client application is able to stream large XML files to the WCF service.

Which operation contract should you create?

A. [OperationContract]

```
void UploadFile(Stream xmlData);
```

B. [OperationContract]

```
void UploadFile(XmlWriter xmlData);
```

C. [OperationContract]

```
void UploadFile(StreamWriter xmlData);
```

D. [OperationContract]

```
void UploadFile(byte[] xmlData);
```

Answer: A

13. You are creating a Windows Communication Foundation (WCF) service by using Microsoft .NET Framework 3.5.

You configure a binding to enable streaming.

You need to ensure that the client application is able to stream large XML files to the WCF service.

Which operation contract should you create?

A. <OperationContract> _

```
Sub UploadFile(ByVal xmlData As Stream)
```

B. <OperationContract> _

```
Sub UploadFile(ByVal xmlData As XmlWriter)
```

C. <OperationContract> _

```
Sub UploadFile(ByVal xmlData As StreamWriter)
```

D. <OperationContract> _

```
Sub UploadFile(ByVal xmlData As Byte())
```

Answer: A

14. You create a Windows Communication Foundation (WCF) service by using Microsoft .NET Framework 3.5.

The WCF service contains two operations named ProcessSimpleOrder and ProcessComplexOrder.

You need to expose the ProcessSimpleOrder operation to all the client applications. You also need to expose the ProcessComplexOrder operation only to specific client applications.

Which code segment should you use?

A. <ServiceContract> _

```
Public Interface IOrderManager
```

```
<OperationContract(Action:= "")> _
```

```
Sub ProcessSimpleOrder()
```

```
<OperationContract> _
```

```
Sub ProcessComplexOrder()
```

```
End Interface
```

B. <ServiceContract> _

```
Public Interface IOrderManager
```

```
<OperationContract(Name:= "http://contoso.com/Simple")> _
```

```
Sub ProcessSimpleOrder()
```

```
<OperationContract(Name:= "http://contoso.com/Complex")> _
```

```
Sub ProcessComplexOrder()
```

```
End Interface
```

C. <ServiceContract> _

```
Public Interface ISimpleOrderManager
```

```
<OperationContract> _
```

```

Sub ProcessSimpleOrder()
End Interface
<ServiceContract(> _
Public Interface IComplexOrderManager
Inherits ISimpleOrderManager
<OperationContract(> _
Sub ProcessComplexOrder()
End Interface
D. <ServiceContract(> _
Public Interface ISimpleOrderManager
<OperationContract(Name:="http://contoso.com/Simple")> _
Sub ProcessSimpleOrder()
End Interface
Public Interface IComplexOrderManager
Inherits ISimpleOrderManager
<OperationContract(Name:="http://contoso.com/Complex")> _
Sub ProcessComplexOrder()
End Interface
Answer: C

```

15. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. You need to ensure that the service can send data in the following format to the client applications.

```

<Account Id="">
<Name> </Name>
<Balance Currency=""> </Balance>
</Account>

```

Which code segment should you use?

```

A. <Serializable(> _
Public Class Account
<XmlAttribute(> _
Public Id As String
<XmlElement(> _
Public Name As String
<XmlAttribute(> _
Public Currency As String
<XmlElement(> _
Public Balance As Double
End Class
B. <DataContract(> _
Public Class Account
<DataMember(Order:=0)> _
Public Id As String
<DataMember(Order:=1)> _
Public Name As String
<DataMember(Order:=0)> _
Public Balance As Double
<DataMember(Order:=1)> _
Public Currency As String
End Class
C. <Serializable(> _
Public Class Account
<XmlAttribute(> _
Public Id As String

```

```

Public Name As String
<XmlElement("Balance")> _
Public Balance As BalanceVal
End Class
<Serializable()> _
Public Class BalanceVal
<XmlText()> _
Public Amount As Double
<XmlAttribute()> _
Public Currency As String
End Class
D. <DataContract()> _
Public Class Account
<DataMember(Order:=0)> _
Public Id As String
<DataMember(Order:=1)> _
Public Name As String
<DataMember(Name:="Balance", Order:=2)> _
Public Balance As BalanceVal
End Class
<DataContract()> _
Public Structure BalanceVal
<DataMember(Order:=0)> _
Public Amount As Double
<DataMember(Order:=1)> _
Public Currency As String
End Structure
Answer: C

```

16. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. You have successfully defined a service contract named IManageOrders.

You write the following code segment.

```

public class OrderImpl : IManageOrders {
public void MarkOrderClosed(int orderId){
try {
...
}
catch (SqlException exc){
throw new FaultException<DataFault>(new DataFault());
}
}
}
[DataContract]
public class DataFault {
}

```

You need to create a fault contract for the MarkOrderClosed method on the IManageOrders service contract.

Which code segment should you add?

- A. [FaultContract(typeof(DataFault))]
- B. [FaultContract(typeof(Exception))]
- C. [FaultContract(typeof(SqlException))]
- D. [FaultContract(typeof(FaultException))]

Answer: A

17. You have created a Windows Communication Foundation service by using Microsoft .NET Framework 5.

The existing service interface is named `IMyService`, and contains the following code segment.

```
[ServiceContract(Name="SvcOrder",  
?Namespace="http://contoso.com/services")]  
public interface IMyService  
{  
[OperationContract]  
void DoSomething();  
}
```

You create a new service named `IMyServiceV1` that contains an operation named `DoSomethingElse`.

You need to ensure that existing client applications are still able to access the `IMyService.DoSomething` method without modifying client code.

Which code segment should you use?

A. `[ServiceContract(Namespace="http://contoso.com/services/V1")]`

```
public interface IMyServiceV1 : IMyService  
{  
[OperationContract]  
void DoSomethingElse();  
}
```

B. `[ServiceContract(Name="SvcOrder")]`

```
public interface IMyServiceV1 : IMyService  
{  
[OperationContract]  
void DoSomethingElse();  
}
```

C. `[ServiceContract(Name="SvcOrderV1",
Namespace="http://contoso.com/services")]`

```
public interface IMyServiceV1 : IMyService  
{  
[OperationContract]  
void DoSomethingElse();  
}
```

D. `[ServiceContract(Name="SvcOrder",
Namespace="http://contoso.com/services")]`

```
public interface IMyServiceV1 : IMyService  
{  
[OperationContract]  
void DoSomethingElse();  
}
```

Answer: D

18. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

You write the following code segment. (Line numbers are included for reference only.)

```
01 [ServiceContract(Namespace="http://uri.contoso.com")]  
02 public interface IMyService  
03 {  
04 [OperationContract]  
05 string ProcessDetails(string s);  
06 [OperationContract(Action="UpdateStatus")]  
07 void UpdateStatus();  
08  
09 }
```

If the existing operation contract is unable to process a request made to the service, a generic operation contract

must attempt to process the request.

You need to create the generic operation contract.

Which code segment should you insert at line 08?

A. [OperationContract(Action="")]

void ProcessOthers(Message msg);

B. [OperationContract(Action="")]

void ProcessOthers();

C. [OperationContract(Action="Default")]

void ProcessOthers(Message msg);

D. [OperationContract(Action="Default")]

void ProcessOthers();

Answer: A

19. You create a Windows Communication Foundation (WCF) service by using Microsoft .NET Framework 3.5.

You write the following code segment. (Line numbers are included for reference only.)

```
01 public interface IMyService
```

```
02 {
```

```
03
```

```
04 string ProcessString(string name);
```

```
05 }
```

You create a host for the WCF service. You also create a service endpoint at <http://localhost:8080/service>. You add an instance of the `HttpTransferEndPointBehavior` class to the host.

You need to ensure that the `ProcessString` method can be invoked from a Web browser by using the URL

<http://localhost:8080/service/process?name=value>

Which code segment should you insert at line 03?

A. [OperationContract(Name="process", Action="Get")]

B. [OperationContract(Name="process", Action="Post")]

C. [OperationContract]

[WebGet(UriTemplate = "process?name={name}")]

D. [OperationContract]

[WebInvoke(UriTemplate = "process?name={name}")]

Answer: C

20. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5.

You write the following code segment. (Line numbers are included for reference only.)

```
01 <ServiceContract(Namespace="http://uri.contoso.com")> _
```

```
02 Public Interface IMyService
```

```
03 <OperationContract> _
```

```
04 Function ProcessDetails(ByVal s As String) As String
```

```
05 <OperationContract(Action="UpdateStatus")> _
```

```
06 Sub UpdateStatus()
```

```
07
```

```
08 End Interface
```

If the existing operation contract is unable to process a request made to the service, a generic operation contract must attempt to process the request.

You need to create the generic operation contract.

Which code segment should you insert at line 07?

A. <OperationContract(Action="")> _

Sub ProcessOthers(ByVal msg As Message)

B. <OperationContract(Action="")> _

Sub ProcessOthers()

C. <OperationContract(Action="Default")> _

Sub ProcessOthers(ByVal msg As Message)

