Testpassport問題集

更に上のクオリティ　更に上のサービス

一年で無料進級することに提供する

Http://www.testpassport.jp

Exam : **70-503 (C#)**

Title : TS: MS.NET Frmwrk 3.5, Wndws Commun Fndtion App Dev

Version : Demo

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

You write the following code segment.

```
namespace MyServices
{
   [ServiceContract()]
   interface IManageOrders
   {
      ...
   }
}
```

The service metadata must be exposed at the relative address named meta.

You need to add an endpoint element to the app.config file of the service host.

Which code fragment should you add?

A. <endpoint address="meta" binding="wsHttpBinding"

contract="IManageOrders" />

B. <endpoint address="meta" binding="wsHttpBinding"

contract="MyServices.IMetadataExchange" />

C. <endpoint address="meta" binding="mexHttpBinding"

contract="IMetadataExchange" />

D. <endpoint address="meta" binding="mexHttpBinding"

contract="MyServices.IManageOrders" />

Answer: C

2. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. You have successfully created two interfaces: IMyService and IMyServiceClient.

You need to ensure that the service is able to call methods from the client application by using the IMyServiceClient interface.

Which code segment should you use?

A. [ServiceContract(CallbackContract=typeof(IMyServiceClient))]

```
public interface IMyService
{
   ...
}
public interface IMyServiceClient
{
   ...
}
```

B.

```
[ServiceContract(CallbackContract=typeof(IMyService))]
public interface IMyService
{
   ...
}
public interface IMyServiceClient
{
   ...
}
```

C.

```
[ServiceContract(SessionMode=SessionMode.Allowed)]
[ServiceKnownType(typeof(IMyServiceClient))]
public interface IMyService : IMyServiceClient
{
   ...
}
public interface IMyServiceClient
{
   ...
}
```

D.

```
[ServiceContract]
[ServiceKnownType(typeof(IMyServiceClient))]
public interface IMyService
```

```
{
   ...
}
[ServiceContract]
public interface IMyServiceClient : ICommunicationObject
{
   ...
}
```

Answer: A

3. 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]
   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

4. 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
{
   …

}
[DataContract]
```

```
public class Customer : Person
{
   …

}
```

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]
   Person GetPerson();
}
```

B.
```
[ServiceContract]
public interface IMyService
{
   [OperationContract]
   [ServiceKnownType("Customer")]
   Person GetPerson();
}
```

C.
```
[ServiceContract]
[ServiceKnownType(typeof(Customer))]
public interface IMyService
{
   [OperationContract]
   Person GetPerson();
```

```
}
```

D.

```
[ServiceContract]
[ServiceKnownType("GetPerson",typeof(Customer))]
public interface IMyService
{
   [OperationContract]
   Person GetPerson();
}
```

Answer: C

5. 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="*")]
   void ProcessSimpleOrder();
   [OperationContract]
   void ProcessComplexOrder();
}
```

B.

```
[ServiceContract]
public interface IOrderManager
{
   [OperationContract(Name="http: //contoso.com/Simple")]
   void ProcessSimpleOrder();
   [OperationContract(Name="http: //contoso.com/Complex")]
```

```
    void ProcessComplexOrder();
}
C. [ServiceContract]
public interface ISimpleOrderManager
{
    [OperationContract]
    void ProcessSimpleOrder();
}
[ServiceContract]
public interface IComplexOrderManager: ISimpleOrderManager
{
    [OperationContract]
    void ProcessComplexOrder();
}
D. [ServiceContract]
public interface ISimpleOrderManager
{
    [OperationContract(Name="http: //contoso.com/Simple")]
    void ProcessSimpleOrder();
}
public interface IComplexOrderManager: ISimpleOrderManager
{
    [OperationContract(Name="http: //contoso.com/Complex")]
    void ProcessComplexOrder();
}
```

Answer: C

6. 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

7. 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

8. 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

9. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. The service will expose a method named DoProcess to the client applications.

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

01 [ServiceContract]

02 public interface IMyService

03 {

04

05 }

You need to ensure that the client applications are not required to obtain results from the DoProcess method.

Which code segment should you insert at line 04?

A. [OperationContract(IsOneWay=true)]

bool DoProcess();

B. [OperationContract(IsOneWay=true)]

void DoProcess();

C. [OperationContract(AsyncPattern=false)]

bool DoProcess();

D. [OperationContract(AsyncPattern=false)]

void DoProcess();

Answer: B

10. 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

11. You create 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(SessionMode=SessionMode.Required)]

02 public interface IOrderManager

03 {

04

05 void CloseOrder();

06 }

You need to decorate the operation as the method that closes the current session.

Which code segment should you insert at line 04?

A. [OperationContract(IsInitiating=false)]

B. [OperationContract(IsTerminating=true)]

C. [OperationContract]

[OperationBehavior(ReleaseInstanceMode=

ReleaseInstanceMode.AfterCall)]

D. [OperationContract(IsTerminating=false)]

[OperationBehavior(ReleaseInstanceMode=

ReleaseInstanceMode.AfterCall)]

Answer: B

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

You create the following service contract. (Line numbers are included for reference only.)

01 [ServiceContract]

02 public interface IMyService

03 {

04 [OperationContract]

05

06 List<string> GetData(int index);

07 }

You need to ensure that the GetData operation can be accessed by using the URI in the following manner:

` GetData/1 for the index value 1

` GetData/2 for the index value 2

` .

` .

` GetData/n for the index value n

Which code segment should you insert at line 05?

A. [WebGet(UriTemplate="GetData/{index}",

?BodyStyle=WebMessageBodyStyle.Bare)]

B. [WebGet(UriTemplate="GetData index={index}",

?BodyStyle=WebMessageBodyStyle.Bare)]

C. [WebGet(UriTemplate="GetData/[index]",

?BodyStyle=WebMessageBodyStyle.Bare)]

D. [WebGet[UriTemplate="GetData index={index}",

?BodyStyle=WebMessageBodyStyle.Bare]]

Answer: A

13. You create a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. You create the following service contract.

```
[ServiceContract]
public interface IMath
{
   [OperationContract]
   int Add(int num1, int num2);
}
```

You need to add an operation contract to perform the Add operation asynchronously.

Which operation contract should you use?

A. [OperationContract(AsyncPattern=true)]

IAsyncResult BeginAdd(int num1, int num2);

int EndAdd(IAsyncResult res);

B. [OperationContract]

int BeginAdd(int num1, int num2, AsyncCallback cb, object state);

IAsyncResult EndAdd();

C. [OperationContract]

IAsyncResult BeginAdd(int num1, int num2);

[OperationContract]

int EndAdd(IAsyncResult res);

D. [OperationContract(AsyncPattern=true)]

IAsyncResult BeginAdd(int num1, int num2, AsyncCallback cb, object

state);

int EndAdd(IAsyncResult res);

Answer: D

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

You need to ensure that data sent in a SOAP header is in the following XML format.

<Data>

<string>String 1</string>

<string>String 2</string>

<string>String 3</string>

</Data>

Which code segment should you use?

A. [MessageContract]

public class MyMessage

{

[MessageHeader]

```
   public string[] Data;
}
```

B.

```
[MessageContract]
public class MyMessage
{
   [MessageHeaderArray]
   public string[] Data;
}
```

C.

```
[MessageContract]
public class MyMessage
{
   [MessageProperty]
   public string[] Data;
}
```

D.

```
[MessageContract]
public class MyMessage
{
   [MessageBodyMember (Order=0)]
   public string[] Data;
}
```

Answer: A

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

You need to programmatically add the following endpoint definition to the service.

http://localhost:8000/ExamService/service

Which code segment should you use?

A.

```
String baseAddress="http: //localhost:8000/ExamService";
BasicHttpBinding binding1=new BasicHttpBinding();
using(ServiceHost host=new ServiceHost(typeof(ExamService)))
```

```
{
   host.AddServiceEndpoint(typeof(IExam),binding1,baseAddress);
}
```

B. String baseAddress="http: //localhost:8000/ExamService/service";

```
BasicHttpBinding binding1=new BasicHttpBinding();
using(ServiceHost host=new ServiceHost(typeof(ExamService)))
{
   host.AddServiceEndpoint(typeof(IExam),binding1,baseAddress);
}
```

C. String baseAddress="http: //localhost:8000/ExamService";

```
WsHttpBinding binding1=new WsHttpBinding();
using(ServiceHost host=new ServiceHost(typeof(ExamService)))
{
   host.AddServiceEndpoint(typeof(IExam),binding1,baseAddress);
}
```

D. String baseAddress="net.tcp: //localhost:8000/ExamService/service";

```
NetTcpBinding binding1=new NetTcpBinding();
using(ServiceHost host=new ServiceHost(typeof(ExamService)))
{
   host.AddServiceEndpoint(typeof(IExam),binding1,baseAddress);
}
```

Answer: B

16. 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 Uri baseAddress=
02    new Uri (″ http://localhost:8000/ExamService″ );
03 Uri mexAddress=new Uri("");
04 ServiceHost serviceHost=new ServiceHost(typeof(
```

```
05 ?ExamService), baseAddress);
06 ServiceMetadataBehavior smb=
07    serviceHost.Description.Behaviors.
08 Find<ServiceMetadataBehavior>();
09 if (smb != null)
10 {
11
12 }
13 else
14 {
15     smb=new ServiceMetadataBehavior();
16     serviceHost.Description.Behaviors.Add(smb);
17 }
18 smb.HttpGetUrl=mexAddress;
19
```

When you browse to the base address of the service, you receive the following message: "Metadata publishing for this service is currently disabled."

You debug the code and discover that the ServiceMetadataBehavior behavior was previously nonexistent.

You need to enable metadata publishing.

What should you do?

A. Delete lines 15 and 16.

B. Modify the code segment at line 03 in the following manner.

Uri mexAddress=new Uri("/service");

C. Insert the following code segment at line 11.

smb.HttpGetEnabled=true;

D. Insert the following code segment at line 19.

smb.HttpGetEnabled=true;

Answer: D

17. You are creating an endpoint for a Windows Communication Foundation service by using

Microsoft .NET Framework 3.5. You create the endpoint by using a custom binding.

You write the following code segment.

```
BasicHttpBinding binding=new BasicHttpBinding();
binding.Security.Message.ClientCredentialType=
?BasicHttpMessageCredentialType.Certificate;
binding.Security.Mode=BasicHttpSecurityMode.Message;
CustomBinding cb=new CustomBinding(binding);
BindingElementCollection bec=cb.CreateBindingElements();
```

You need to prevent the custom binding from making a persistent connection to the service endpoint.

Which code segment should you use?

A.
```
cb.SendTimeout=TimeSpan.Zero;
```

B.
```
binding.ReceiveTimeout=TimeSpan.Zero;
```

C.
```
foreach (BindingElement bindingElement in bec)
{
   HttpTransportBindingElement element=
    (HttpTransportBindingElement) bindingElement;
   element.KeepAliveEnabled=false;
}
```

D.
```
foreach (BindingElement be in bec)
{
   if (be is HttpTransportBindingElement)
   {
      HttpTransportBindingElement httpElement=
       (HttpTransportBindingElement)be;
      httpElement.KeepAliveEnabled=false;
   }
}
```

Answer: D

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

The service will receive notification messages from client applications by using a number of transports, including HTTP, TCP, MSMQ, and others.

You need to ensure that all client applications can send notification messages to the service.

Which service contract should you use?

A. [ServiceContract]

public interface INotificationService {

[OperationContract(AsyncPattern=true)]

void LogMessage(string message);

}

B. [ServiceContract]

public interface INotificationService {

[OperationContract(ReplyAction="none")]

void LogMessage(string message);

}

C. [ServiceContract]

public interface INotificationService {

[OperationContract(IsOneWay=true)]

void LogMessage(string message);

}

D. [ServiceContract]

public interface INotificationService {

[OperationContract(IsTerminating=true)]

void LogMessage(string message);

}

Answer: C

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

The service will be hosted in a Console application.

You need to configure the service by using a configuration file other than the default app.config file.

Which code segment should you use?

A. class MyServiceHost : ServiceHost

```
{
   public MyServiceHost(Type serviceType,
     params Uri[] baseAddresses) : base(serviceType, baseAddresses)
   {
   }
   protected override void InitializeRuntime()
   {
      //Load configuration here
   }
}
```

B. class MyServiceHost : ServiceHost

```
{
   public MyServiceHost(Type serviceType,
     params Uri[] baseAddresses) : base(serviceType, baseAddresses)
   {
   }
   protected override void ApplyConfiguration()
   {
      //Load configuration here
   }
}
```

C. class MyServiceHost : ServiceHost

```
{
   public MyServiceHost(Type serviceType,
     params Uri[] baseAddresses) : base(serviceType, baseAddresses)
   {
   }
   protected new void InitializeDescription(Type serviceType,
```

```
    UriSchemeKeyedCollection baseAddresses)
  {
     //Load configuration here.
  }
}
```

D.

```
class MyServiceHost : ServiceHost
{
  public MyServiceHost(Type serviceType,
    params Uri[] baseAddresses) : base(serviceType, baseAddresses)
  {
  }
  protected new void AddBaseAddress(Uri baseAddress)
  {
     //Load configuration here.
  }
}
```

Answer: B

20. You are creating a Windows Communication Foundation service by using Microsoft .NET Framework 3.5. The service will be hosted in a managed Console application.

You want to add endpoints to the service.

You need to ensure that all endpoints use the same base address.

Which code fragment should you use?

A.

```
[ServiceContract]
public interface IMortgageService {…-
public class MortgageService : IMortgageService {…-
Uri baseAddress=new Uri("http: //localhost:8888/MortgageService");
ServiceHost serviceHost=
  new ServiceHost(typeof(MortgageService), new Uri[] {baseAddress });
serviceHost.AddServiceEndpoint(typeof(IMortgageService),
```

```
  new BasicHttpBinding(), "");
serviceHost.Open();
```

B. [ServiceContract]

```
public interface IMortgageService {…-
public class MortgageService : IMortgageService {…-
Uri baseAddress=new Uri("http: //localhost:8888/MortgageService");
ServiceHost serviceHost=
  new ServiceHost(typeof(MortgageService), new Uri[] {});
serviceHost.AddServiceEndpoint(typeof(IMortgageService),
  new BasicHttpBinding(), baseAddress);
serviceHost.Open();
```

C. [ServiceContract]

```
public interface IMortgageService {…-
public class MortgageService : IMortgageService {…-
string baseAddress="http: //localhost:8888/MortgageService";
ServiceHost serviceHost=
  new ServiceHost(typeof(MortgageService), new Uri[] { });
serviceHost.AddServiceEndpoint(typeof(IMortgageService),
  new BasicHttpBinding(), baseAddress);
serviceHost.Open();
```

D. [ServiceContract(Namespace="http: //localhost:8888/MortgageService")]

```
public interface IMortgageService {…-
public class MortgageService : IMortgageService {…-
ServiceHost serviceHost=
  new ServiceHost(typeof(MortgageService), new Uri[] { });
serviceHost.AddServiceEndpoint(typeof(IMortgageService),
  new BasicHttpBinding(), "");
serviceHost.Open();
```

Answer: A