

Microsoft

EXAM 70-513

TS:Windows Communication Found Dev w/MS .NET Frmwk (C# and VB) 4

70-513CSHARP Questions: 81

70-513VB Questions: 81

Tootle Questions: 162

Total Questions:

162

70-513CSHARP

Question: 1.

You are creating a Windows Communication Foundation (WCF) service that is implemented as follows. (Line numbers are included for reference only.)

```
01[ServiceContract]
02[ServiceBehavior(IncludeExceptionDetailsInFaults = true)]
03public class OrderService
04{
05[OperationContract]
06public void SubmitOrder(Order anOrder)
07{
08try
09{
10&
11}
12catch(DivideByZeroException ex)
13{
14
15}
16}
17}
```

You need to ensure that the stack trace details of the exception are not included in the error information sent to the client. What should you do?

A. Replace line 14 with the following line.

```
throw;
```

B. Replace line 14 with the following line.

```
throw new FaultException<Order>(anOrder, ex.ToString());
```

C. After line 05, add the following line.

```
[FaultContract(typeof(FaultException<Order>))]
```

Replace line 14 with the following line.

```
throw ex;
```

D. After line 05, add the following line.

```
[FaultContract(typeof(FaultException<Order>))]
```

Replace line 14 with the following line.

```
throw new FaultException<Order>(anOrder, "Divide by zero exception");
```

Answer: D

Question: 2.

You are creating a Windows Communication Foundation (WCF) service. You do not want to expose the internal implementation at the service layer. You need to expose the following class as a service named Arithmetic with an operation named Sum.

```
public class Calculator
{
    public int Add(int x, int y)
    {
    }
}
```

Which code segment should you use

A. [ServiceContract(Namespace="Arithmetic")]

```
public class Calculator
{
    [OperationContract(Action="Sum")]
    public int Add(int x, int y)
    {
    }
}
```

B. [ServiceContract(ConfigurationName="Arithmetic")]

```
public class Calculator
{
    [OperationContract(Action="Sum")]
    public int Add(int x, int y)
    {
    }
}
```

C. [ServiceContract(Name="Arithmetic")]

```
public class Calculator
{
    [OperationContract(Name="Sum")]
    public int Add(int x, int y)
    {
    }
}
```

D. [ServiceContract(Name="Arithmetic")]

```
public class Calculator
{
    [OperationContract(ReplyAction="Sum")]
    public int Add(int x, int y)
    {
    }
}
```

```
&
}
}
```

Answer: C

Question: 3.

You are developing a data contract for a Windows Communication Foundation (WCF) service. The data in the data contract must participate in round trips. Strict schema validity is not required. You need to ensure that the contract is forward-compatible and allows new data members to be added to it. Which interface should you implement in the data contract class?

- A. ICommunicationObject
- B. IExtension<T>
- C. IExtensibleObject<T>
- D. IExtensibleDataObject

Answer: D

Question: 4.

A Windows Communication Foundation (WCF) application uses the following data contract.

```
[DataContract]
public class Person
{
    [DataMember]
    public string firstName;
    [DataMember]
    public string lastName;
    [DataMember]
    public int age;
    [DataMember]
    public int ID;
}
```

You need to ensure that the following XML segment is generated when the data contract is serialized.

```
<Person>
  <firstName xsi:nil="true" />
  <lastName xsi:nil="true" />
  <ID>999999999</ID>
</Person>
```

Which code segment should you use?

A. [DataMember]
public string firstName;
[DataMember]
public string lastName;
[DataMember(EmitDefaultValue = true)]
public int age = 0 ;
[DataMember(EmitDefaultValue = true)]
public int ID = 999999999;

B. [DataMember(EmitDefaultValue = false)]
public string firstName = null;
[DataMember(EmitDefaultValue = false)]
public string lastName = null;
[DataMember(EmitDefaultValue = true)]
public int age = -1;
[DataMember(EmitDefaultValue = false)]
public int ID = 999999999;

C. [DataMember(EmitDefaultValue = true)]
public string firstName;
[DataMember(EmitDefaultValue = true)]
public string lastName;
[DataMember(EmitDefaultValue = false)]
public int age = -1;
[DataMember(EmitDefaultValue = false)]
public int ID = 999999999;

D. [DataMember]
public string firstName = null;
[DataMember]
public string lastName = null;
[DataMember(EmitDefaultValue = false)]
public int age = 0;
[DataMember(EmitDefaultValue = false)]
public int ID = 999999999;

Answer: D

Question: 5.

You are developing a client that sends several types of SOAP messages to a Windows Communication Foundation (WCF) service method named PostData.

PostData is currently defined as follows.

```
[OperationContract]  
void PostData(Order data);
```

You need to modify PostData so that it can receive any SOAP message.

Which code segment should you use?

- A. [OperationContract(IsOneWay = true, Action = "*", ReplyAction = "*")]
void PostData(Order data);
- B. [OperationContract(IsOneWay = true, Action = "*", ReplyAction = "*")]
void PostData(BodyWriter data);
- C. [OperationContract]
void PostData(BodyWriter data);
- D. [OperationContract]
void PostData(Message data);

Answer: D

Question: 6.

A class named TestService implements the following interface.

```
[ServiceContract]
public interface ITestService {
    [OperationContract]
    DateTime GetServiceTime();
}
```

TestService is hosted in an ASP.NET application.

You need to modify the application to allow the GetServiceTime method to return the data formatted as JSON. It must do this only when the request URL ends in /ServiceTime. What should you do?

A. Add this attribute to the GetServiceTime method.

```
[WebInvoke(Method="POST")]
```

In the web.config file, add this element to system.serviceModel/behaviors/endpointBehaviors.

```
<behavior name="Json">
  <enableWebScript />
</behavior>
```

In the web.config file, configure TestService in the system.serviceModel/services collection as follows.

```
<service name="TestService">
  < endpoint address="/ServiceTime"
contract="TestService"
behaviorConfiguration="Json"
binding="webHttpBinding" />
</service>
```

B. Add this attribute to the GetServiceTime method.

```
[WebInvoke(Method = "GET",
UriTemplate = "/ServiceTime",
ResponseFormat = WebMessageFormat.Json)]
```

In the web.config file, configure TestService in the system.serviceModel/services collection as follows.

```
<service name="TestService">
  <endpoint address="/ServiceTime"
```

```
contract="TestService"
binding="webHttpBinding" />
</service>
```

C. Add this attribute to the GetServiceTime method.

```
[WebGet(
  ResponseFormat= WebMessageFormat.Json,
  UriTemplate="/ServiceTime")]
```

Create a new .svc file named JsonVersion.svc with the following content.

```
<%@ ServiceHost Service="TestService"
  Factory="System.ServiceModel.Activation.WebServiceHostFactory" %>
```

D. Add this attribute to the GetServiceTime method.

```
[WebGet(UriTemplate = "{Json}/ServiceTime")]
```

Create a new .svc file named JsonVersion.svc with the following content.

```
<%@ ServiceHost Service="TestService"
  Factory="System.ServiceModel.Activation.WebServiceHostFactory" %>
```

Answer: C

Question: 7.

You are creating a Windows Communication Foundation (WCF) service that implements operations in a RESTful manner. You need to add a delete operation.

You implement the delete method as follows.

```
void DeleteItems(string id);
```

You need to configure WCF to call this method when the client calls the service with the HTTP DELETE operation. What should you do?

- A. Add the WebInvoke(UriTemplate = "/Items/{id}", Method="DELETE") attribute to the operation.
- B. Add the HttpDelete attribute to the operation.
- C. Replace the string parameter with a RemovedActivityAction parameter.
- D. Replace the return type with RemovedActivityAction.

Answer: A

Question: 8.

You are creating a Windows Communication Foundation (WCF) service that responds using plain-old XML (POX).

You have the following requirements:

You must enable the /catalog.svc/items operation to respond using the POX, JSON, or ATOM formats. You also must ensure that the same URL is used regardless of the result type.

You must determine the response format by using the Accepts HTTP header.

What should you do?

- A. Implement the `IChannelInitializer` interface in the service class.
- B. Implement the `System.Runtime.Serialization.IFormatterConverter` interface in the service class.
- C. Set the `BodyStyle` parameter of the `WebGet` attribute on the operation to `WebMessageBodyStyle.WrappedResponse`.
- D. Set the return type of the operation to `System.ServiceModel.Channels.Message`. Use the current `WebOperationContext` methods to return the data in the required format.

Answer: D

Question: 9.

A Windows Communication Foundation (WCF) solution uses two services to manage a shopping cart. Service A processes messages containing line items that total between \$0 and \$500. Service B processes messages containing line items that total more than \$500. All messages are of equal importance to the business logic. You need to route incoming messages to the appropriate services by using WCF routing. Which two message filters should you add to the router (Each correct answer presents part of the solution? Choose two.)

- A. a message filter with a priority of 100 that will forward messages that total between \$0 and \$500 to Service A
- B. a message filter with a priority of 0 that will forward messages that total between \$0 and \$500 to Service A
- C. a message filter with a priority of 0 that will forward all messages to Service B
- D. a message filter with a priority of 100 that will forward all messages to Service B

Answer: AC

Question: 10.

You have an existing Windows Communication Foundation (WCF) service. You need to ensure that other services are notified when the service is started. What should you do?

- A. Add the following standard endpoint to the service.
`<endpoint name="udpAnnouncementEndpoint" kind="udpDiscoveryEndpoint" />`
- B. Add the following standard endpoint to the service.
`<endpoint name="udpDiscoveryEndpoint" kind="udpAnnouncementEndpoint" />`
- C. Add a service behavior with the following element.
`<serviceDiscovery>
<announcementEndpoints>
<endpoint kind="udpDiscoveryEndpoint" />`


```
</announcementEndpoints>
</serviceDiscovery>
D. Add a service behavior with the following element.
<serviceDiscovery>
  <announcementEndpoints>
    <endpoint kind="udpAnnouncementEndpoint" />
  </announcementEndpoints>
</serviceDiscovery>
```

Answer: D

Question: 11.

You are developing a Windows Communication Foundation (WCF) service that reads messages from a public non-transactional MSMQ queue. You need to configure the service to read messages from the failed-delivery queue. Which URI should you specify in the endpoint configuration settings of the service?

- A. net.msmq://localhost/ msmq\$;FailedMessages
- B. net.msmq://localhost/ msmq\$;DeadLetter
- C. net.msmq://localhost/system\$;DeadXact
- D. net.msmq://localhost/system\$;DeadLetter

Answer: D

Question: 12.

You have an existing Windows Communication Foundation (WCF) service that exposes a service contract over HTTP using explicit binding configuration. You need to expose that contract over HTTP and TCP. What should you do?

- A. Add a net.tcp base address to the host.
- B. Add an endpoint configured with a netTcpBinding.
- C. Add an endpoint behavior named netTcpBehavior to the existing endpoint.
- D. Add a binding configuration to the existing endpoint named netTcpBinding.

Answer: B

Question: 13.

You have an existing Windows Communication Foundation (WCF) Web service. The Web service is not responding to messages larger than 64 KB. You need to ensure that the Web service can accept messages larger than 64 KB without generating errors. What should you do?

- A. Increase the value of `maxReceivedMessageSize` on the endpoint binding.
- B. Increase the value of `maxRequestLength` on the `httpRuntime` element.
- C. Increase the value of `maxBufferSize` on the endpoint binding.
- D. Increase the value of `maxBufferPoolSize` on the endpoint binding.

Answer: A

Question: 14.

A Windows Communication Foundation (WCF) service is responsible for transmitting XML documents between systems. The service has the following requirements:

It must minimize the transmission size by attaching the XML document as is without using escape characters or base64 encoding.

It must interoperate with systems that use SOAP but are not built on the .NET platform.

You need to configure the service to support these requirements.

Which message encoding should you use?

- A. Binary message encoding
- B. MTOM (Message Transmission Optimization Mechanism) message encoding
- C. Text message encoding with message version set to none
- D. Text message encoding with message version set to SOAP 1.2

Answer: B

Question: 15.

You are modifying an existing Windows Communication Foundation (WCF) service that is defined as follows.

```
[ ServiceContract ]
public interface IMessageProcessor {
    [OperationContract]
    void ProcessMessage ();
}
public class MessageProcessor : IMessageProcessor {
    public void ProcessMessage () {
```

```
&  
  SubmitOrder ();  
&  
}  
}
```

SubmitOrder makes a call to another service. The ProcessMessage method does not perform as expected under a heavy load. You need to enable processing of multiple messages. New messages must only be processed when the ProcessMessage method is not processing requests, or when it is waiting for calls to SubmitOrder to return. Which attribute should you apply to the MessageProcessor Class?

- A. CallbackBehavior (ConcurrencyMode = ConcurrencyMode.Reentrant)
- B. CallbackBehavior (ConcurrencyMode = ConcurrencyMode.Multiple)
- C. ServiceBehavior (ConcurrencyMode = ConcurrencyMode.Reentrant)
- D. ServiceBehavior (ConcurrencyMode = ConcurrencyMode.Multiple)

Answer: C

Question: 16.

You are adding a Windows Communication Foundation (WCF) service to an existing application. The application is configured as follows. (Line numbers are included for reference only.)

```
01 <configuration>  
02 < system.serviceModel >  
03 <services>  
04 <service name=" Contoso.Sales.StockService "  
05 behaviorConfiguration =" MetadataBehavior " >  
06 <host>  
07 < baseAddresses >  
08<add  
  baseAddress ="http://contoso.com:8080/StockService" />  
09 </ baseAddresses >  
10 </host>  
11 </service>  
12 </services>  
13 <behaviors>  
14 < serviceBehaviors >  
15 <behavior name=" MetadataBehavior ">  
16 </behavior>  
17 </ serviceBehaviors >  
18 </behaviors>
```

...

You need to configure the service to publish the service metadata.

Which two actions should you perform (Each correct answer presents part of the solution. Choose two.)

A. Add the following XML segment between lines 10 and 11.

```
<endpoint address=""  
  binding= " mexHttpBinding "  
  contract= " IMetadataExchange "  
>  
</>
```

B. Add the following XML segment between lines 10 and 11.

```
<endpoint address=""  
  binding= " basic HttpBinding "  
  contract= " IMetadataExchange "  
>  
</>
```

C. Add the following XML segment between lines 15 and 16.

```
< serviceDiscovery >  
  < announcementEndpoints >  
    <endpoint address=""/>  
  </ announcementEndpoints >  
</ serviceDiscovery >
```

D. Add the following XML segment between lines 15 and 16

```
< serviceMetadata httpGetEnabled ="true" />
```

Answer: AD

Question: 17.

Four Windows Communication Foundation (WCF) services are hosted in Microsoft Internet Information Services (IIS). No behavior configuration exists in the web.config file. You need to configure the application so that every service and endpoint limits the number of concurrent calls to 50 and the number of concurrent sessions to 25. Which XML segment should you add to the system.serviceModel configuration section of the web.config file?

A. <behaviors>

```
< serviceBehaviors >  
  < behavior name= " * " >  
    < serviceThrottling maxConcurrentCalls ="50" maxConcurrentSessions ="25"/>  
  </behavior>  
</ serviceBehaviors >  
</behaviors>
```

B. < behaviors>

```
< serviceBehaviors >  
  <behavior name="default">  
    < serviceThrottling maxConcurrentCalls ="50" maxConcurrentSessions ="25"/>  
  </behavior>  
</ serviceBehaviors >  
</behaviors>
```

C. <behaviors>

```
< serviceBehaviors >  
  <behavior name="">  
    < serviceThrottling maxConcurrentCalls ="50" maxConcurrentSessions ="25"/>
```

```
</behavior>
</ serviceBehaviors >
</behaviors>
D. <behaviors>
  < serviceBehaviors >
    <behavior name=" ALL ">
      < serviceThrottling maxConcurrentCalls ="50" maxConcurrentSessions ="25"/>
    </behavior>
  </ serviceBehaviors >
</behaviors>
```

Answer: C

Question: 18.

A Windows Communication Foundation (WCF) service is self-hosted in a console application. The service implements the `ITimeService` service interface in the `TimeService` class. You need to configure the service endpoint for HTTP communication. How should you define the service and endpoint tags?

A. Define the service tag as follows.

```
<service name=" ITimeService " >
  Define the endpoint tag as follows.
  < endpoint kind =" TimeService "
    address="http://localhost:8080/TimeService"
    binding=" wsHttpBinding "
    contract=" ITimeService"/ >
```

B. Define the service tag as follows.

```
<service name=" TimeService ">
  Define the endpoint tag as follows.
  < endpoint kind=" TimeService "
    address="http://localhost:8080/TimeService"
    binding=" wsHttpBinding "
    contract=" ITimeService " / >
```

C. Define the service tag as follows.

```
<service name=" ITimeService ">
  Define the endpoint tag as follows.
  < endpoint name=" TimeService "
    address="http://localhost:8080/TimeService"
    binding=" wsHttpBinding "
    contract=" ITimeService "/ >
```

D. Define the service tag as follows.

```
<service name=" TimeService ">
  Define the endpoint tag as follows.
  <endpoint address="http://localhost:8080/TimeService"
    binding=" wsHttpBinding "
    contract=" ITimeService "/ >
```

Answer: D

Question: 19.

A Windows Communication Foundation (WCF) service is self-hosted in a console application. The service implements the IDataAccess contract, which is defined in the MyApplication namespace. The service is implemented in a class named DataAccessService, which implements the IDataAccess interface and also is defined in the MyApplication namespace.

The hosting code is as follows. (Line numbers are included for reference only.)

```
01 static void Main(string[] args)
02 {
03     ServiceHost host;
04
05     host.Open();
06     Console.ReadLine();
07     host.Close();
08 }
```

You need to create a ServiceHost instance and assign it to the host variable. You also need to instantiate the service host.

Which line of code should you insert at line 04?

- A. host = new ServiceHost("MyApplication.DataAccessService");
- B. host = new ServiceHost("MyApplication.IDataAccess");
- C. host = new ServiceHost(typeof(IDataAccess));
- D. host = new ServiceHost(typeof(DataAccessService));

Answer: D

Question: 20.

A Windows Communication Foundation (WCF) service implements the following contract.

```
[ ServiceContract ]
public interface IHelloService
{
    [OperationContract]
    [WebGet ( UriTemplate = " helloname ={name}")]
    string SayHello (string name);
}
```

The implementation is as follows.

```
public class HelloService : IHelloService
{
    public string SayHello (string name)
    {
        return "Hello " + name;
    }
}
```

```
}  
}
```

The service is self-hosted, and the hosting code is as follows.

```
Web ServiceHost svcHost = CreateHost ();  
svcHost.Open ();  
Console.ReadLine ();  
svcHost.Close ();
```

You need to implement CreateHost so that the service has a single endpoint hosted at `http://localhost:8000/HelloService`. Which code segment should you use?

A. `WebServiceHost svcHost = new WebServiceHost (typeof (HelloService));`

```
svcHost.AddServiceEndpoint ( typeof ( IHelloService ),  
    new WebHttpBinding ( WebHttpSecurityMode.None ),  
    "http://localhost:8000/HelloService");  
return svcHost ;
```

B. `Uri baseAddress = new Uri("http://localhost:8000/");`

```
WebServiceHost svcHost =  
    new WebServiceHost ( typeof ( HelloService ), baseAddress );  
svcHost.AddServiceEndpoint ( typeof ( IHelloService ),  
    new WebHttpBinding ( WebHttpSecurityMode.None ),  
    " HelloService ");  
return svcHost ;
```

C. `WebServiceHost svcHost = new WebServiceHost (new HelloService ());`

```
svcHost.AddServiceEndpoint ( typeof ( IHelloService ),  
    new WebHttpBinding ( WebHttpSecurityMode.None ),  
    "http://localhost:8000/HelloService");  
return svcHost ;
```

D. `Uri baseAddress = new Uri("http://localhost:8000/");`

```
WebServiceHost svcHost =  
    new WebServiceHost (new HelloService (), baseAddress );  
svcHost.AddServiceEndpoint ( typeof ( IHelloService ),  
    new WebHttpBinding ( WebHttpSecurityMode.None ),  
    " HelloService ");  
return svcHost ;
```

Answer: A

Question: 21.

You are developing a Windows Communication Foundation (WCF) service that will be hosted in Microsoft Internet Information Services (IIS) 7.0. The service must be hosted in an IIS application named Info. You need to enable this service to be hosted in IIS by changing the web.config file. Which XML segment should you add to the web.config file?

- A. < serviceHostingEnvironment >
< serviceActivations >
<add relativeAddress =" Info.svc " service="Info" />
</ serviceActivations >
</ serviceHostingEnvironment >
- B. < serviceHostingEnvironment >
< serviceActivations >
<add relativeAddress ="Info" service=" Info.svc " />
</ serviceActivations >
</ serviceHostingEnvironment >
- C. < serviceHostingEnvironment >
< transportConfigurationTypes >
<add name="Info" transportConfigurationType =" Info.svc " />
</ transportConfigurationTypes >
</ serviceHostingEnvironment >
- D. < serviceHostingEnvironment >
< transportConfigurationTypes >
<add name=" Info.svc " transportConfigurationType =" FileNotRequired " />
</ transportConfigurationTypes >
</ serviceHostingEnvironment >

Answer: A

Question: 22.

You are creating a Windows Communication Foundation (WCF) service. You need to ensure that the service is compatible with ASP.NET to make use of the session state. Which binding should you use?

- A. NetTcp ContextBinding
- B. BasicHttpContextBinding
- C. NetTcp Binding
- D. NetMsmqBinding

Answer: B

Question: 23.

A Windows Communication Foundation (WCF) service has a callback contract. You are developing a client application that will call this service. You must ensure that the client application can interact with the WCF service. What should you do?

- A. On the OperationContractAttribute, set the AsyncPattern property value to true.
- B. On the OperationContractAttribute, set the ReplyAction property value to the endpoint address of the client.
- C. On the client, create a proxy derived from DuplexClientBase<TChannel>.
- D. On the client, use GetCallbackChannel<T>.

Answer: C

Question: 24.

You are creating a Windows Communication Foundation (WCF) service.

You have the following requirements:

Messages must be sent over TCP.

The service must support transactions.

Messages must be encoded using a binary encoding.

Messages must be secured using Windows stream-based security.

You need to implement a custom binding for the service.

In which order should the binding stack be configured?

- A. tcpTransport
windowsStreamSecurity
transactionFlow
binaryMessageEncoding
- B. transactionFlow
binaryMessageEncoding
windowsStreamSecurity
tcpTransport
- C. windowsStreamSecurity
tcpTransport
binaryMessageEncoding
transactionFlow
- D. binaryMessageEncoding
transactionFlow
tcpTransport
windowsStreamSecurity

Answer: B

Question: 25.

A Windows Communication Foundation (WCF) client configuration file contains the following XML segment in the system.serviceModel element.

```
<client>
<endpoint address=" net.tcp://server/ContosoService "
binding=" netTcpBinding "
contract=" Contoso. IContosoService "
name=" netTcp " />
<endpoint address=" net.pipe://localhost/ContosoService "
binding=" netNamedPipeBinding "
contract=" Contoso. IContosoService "
name=" netPipe " />
</client>
```

You need to create a channel factory that can send messages to the endpoint listening at net.pipe://localhost/ContosoService. Which code segment should you use?

- A. ChannelFactory < Contoso. IContosoService > factory =
new ChannelFactory < Contoso. IContosoService >(" Contoso. IContosoService ");
- B. ChannelFactory < Contoso. IContosoService > factory =
new ChannelFactory < Contoso. IContosoService >(" netNamedPipeBinding ");
- C. ChannelFactory < Contoso. IContosoService > factory =
new ChannelFactory < Contoso. IContosoService >(" netPipe ");
- D. ChannelFactory < Contoso. IContosoService > factory =
new ChannelFactory < Contoso. IContosoService >(
" net.pipe://localhost/ContosoService ");

Answer: C

Question: 26.

A Windows Communication Foundation (WCF) solution uses the following contracts. (Line numbers are included for reference only.)

```
01 [ServiceContract(CallbackContract = typeof(INameService))]
02 public interface IGreetingService
03 {
04 [OperationContract]
05 string GetMessage();
06 }
07
08 [ServiceContract]
09 public interface INameService
```

```
10 {
11 [OperationContract]
12 string GetName();
13 }
```

When the client calls GetMessage on the service interface, the service calls GetName on the client callback. In the client, the class NameService implements the callback contract.

The client channel is created as follows.

```
22 In stanceContext callbackContext =
  new InstanceContext(new NameService("client"));
&
25 DuplexChannelFactory<IGreetingService> factory =
  new DuplexChannelFactory<IGreetingService>(
  typeof(NameService), binding, address);
26 IGreetingService greetingService = factory.CreateChannel();
```

You need to ensure that the service callback is processed by the instance of NameService. What are two possible ways to achieve this goal (Each correct answer presents a complete solution? Choose two.)

A. Change line 25 to the following code segment.

```
DuplexChannelFactory<IGreetingService> factory =
  new DuplexChannelFactory<IGreetingService>(
  callbackContext, binding, address);
```

B. Change line 26 to the following code segment.

```
IGreetingService greetingService =
  factory.CreateChannel(callbackContext);
```

C. Add the following code segment after line 26.

```
callbackContext.IncomingChannels.Add((IDuplexChannel)greetingService);
```

D. Add the following code segment after line 26.

```
callbackContext.OutgoingChannels.Add((IDuplexChannel)greetingService);
```

Answer: AB

Question: 27.

A Windows Communication Foundation (WCF) service implements the following contract. (Line numbers are included for reference only.)

```
01 [ServiceContract]
02 public interface IDataAccessService
03 {
04 [OperationContract]
05 void PutMessage(string message);
06
07 [OperationContract]
08 [FaultContract(typeof(TimeoutFaultException))]
09 [FaultContract(typeof(FaultException))]
10 string[] SearchMessages(string search);
```

11 }

The implementation of the SearchMessages method throws TimeoutFaultException exceptions for database timeouts. The implementation of the SearchMessages method also throws an Exception for any other issue it encounters while processing the request. These exceptions are received on the client side as generic FaultException exceptions. You need to implement the error handling code for SearchMessages and create a new channel on the client only if the channel faults. What should you do?

- A. Catch and handle both TimeoutFaultException and FaultException.
- B. Catch both TimeoutFaultException and FaultException. Create a new channel in both cases.
- C. Catch and handle TimeoutFaultException. Catch FaultException and create a new channel.
- D. Catch and handle FaultException. Catch TimeoutFaultException and create a new channel.

Answer: C

Question: 28.

An ASP.NET application hosts a RESTful Windows Communication Foundation (WCF) service at /Services/Contoso.svc. The service provides a JavaScript resource to clients. You have an explicit reference to the JavaScript in your page markup as follows.

```
<script type="text/javascript" src="/Services/Contoso.svc/js" />
```

You need to retrieve the debug version of the service JavaScript. What should you do?

- A. In the <%@ ServiceHost %> header for /Services/Contoso.svc, set the Debug attribute to true.
- B. In the <%@ Page %> header, set the Debug attribute to true.
- C. In the script tag, add a debug attribute and set its value to true.
- D. In the script tag, append debug to the src attribute.

Answer: D

Question: 29.

You are consuming a Windows Communication Foundation (WCF) service. The service interface is defined as follows.

```
[DataContract(Namespace = "")]  
public class Item  
{  
&  
}  
[ServiceContract(Namespace = "")]  
public interface ICatalog  
{
```

```
[OperationContract]
[WebInvoke(Method = "POST", UriTemplate = "/Item")]
Item UpdateItem(Item item);
}
```

The client application receives a `WebResponse` named `response` with the response from the service. You need to deserialize this response into a strongly typed object representing the return value of the method. Which code segment should you use?

- A. `DataContractSerializer s = new DataContractSerializer(typeof(Item));`
`Item item = s.ReadObject(response.GetResponseStream()) as Item;`
- B. `BinaryFormatter f = new BinaryFormatter();`
`Item item = f.Deserialize(response.GetResponseStream()) as Item;`
- C. `XmlDictionaryReader r = JsonReaderWriterFactory.CreateJsonReader(`
`response.GetResponseStream(),`
`XmlDictionaryReaderQuotas.Max);`
`DataContractSerializer s = new DataContractSerializer(typeof(Item));`
`Item item = s.ReadObject(r) as Item;`
- D. `DataContractJsonSerializer s = new DataContractJsonSerializer(typeof(Item));`
`Item item = s.ReadObject(response.GetResponseStream()) as Item;`

Answer: A

Question: 30.

You are developing an application to update a user's social status. You need to consume the service using Windows Communication Foundation (WCF).

The client configuration is as follows.

```
<system.serviceModel>
<bindings>
<webHttpBinding>
<binding name="SocialConfig">
<security mode="TransportCredentialOnly">
<transport clientCredentialType="Basic"
realm="Social API" />
</security>
</binding>
</webHttpBinding>
</bindings>
<client>
<endpoint address="http://contoso.com"
binding="webHttpBinding"
bindingConfiguration="SocialConfig"
contract="ISocialStatus"
name="SocialClient" />
</client>
</system.serviceModel>
```

The service contract is defined as follows.

```
[ServiceContract]
```

```
public interface ISocialStatus
{
[OperationContract]
[WebInvoke(UriTemplate =
"/statuses/update.xml?status={text}")]
void UpdateStatus(string text);
}
```

Which code segment should you use to update the social status?

A. using (WebChannelFactory<ISocialStatus> factory =
new WebChannelFactory<ISocialStatus>("SocialClient"))

```
{
factory.Credentials.UserName.UserName = user.Name;
factory.Credentials.UserName.Password = user.Password;
ISocialStatus socialChannel = factory.CreateChannel();
socialChannel.UpdateStatus(newStatus);
}
```

B. using (ChannelFactory<ISocialStatus> factory =
new WebChannelFactory<ISocialStatus>(typeof(ISocialStatus)))

```
{
factory.Credentials.UserName.UserName = user.Name;
factory.Credentials.UserName.Password = user.Password;
ISocialStatus socialChannel = factory.CreateChannel();
socialChannel.UpdateStatus(newStatus);
}
```

C. using (ChannelFactory<ISocialStatus> factory =
new ChannelFactory<ISocialStatus>("POST"))

```
{
factory.Credentials.Windows.ClientCredential.UserName =
user.Name;
factory.Credentials.Windows.ClientCredential.SecurePassword.
SetAt(0, Convert.ToChar(user.Password) );
ISocialStatus socialChannel = factory.CreateChannel();
socialChannel.UpdateStatus(newStatus);
}
```

D. using (WebChannelFactory<ISocialStatus> factory =
new WebChannelFactory<ISocialStatus>(typeof(ISocialClient)))

```
{
factory.Credentials.Windows.ClientCredential.UserName =
user.Name;
factory.Credentials.Windows.ClientCredential.SecurePassword.
SetAt(0, Convert.ToChar(user.Password) );
ISocialStatus socialChannel = factory.CreateChannel();
socialChannel.UpdateStatus(newStatus);
}
```

Answer: A

Question: 31.

A Windows Communication Foundation (WCF) client application is consuming an RSS syndication feed from a blog. You have a SyndicationFeed variable named feed. The application iterates through the items as follows. (Line numbers are included for reference only.)

```
01 foreach (SyndicationItem item in feed.Items)
```

```
02 {
```

```
03 }
```

You need to display the content type and body of every syndication item to the console. Which two lines of code should you insert between lines 02 and 03?

- A. Console.WriteLine(item.Content.Type);
Console.WriteLine(((TextSyndicationContent)item.Content).Text);
- B. Console.WriteLine(item.Content.GetType());
Console.WriteLine(((TextSyndicationContent)item.Content).Text);
- C. Console.WriteLine(item.Content.Type);
Console.WriteLine(item.Content.ToString());
- D. Console.WriteLine(item.Content.GetType());
Console.WriteLine(item.Content.ToString());

Answer: A

Question: 32.

The endpoint of a Windows Communication Foundation (WCF) service uses basicHttpBinding for its binding. Your company's policies have changed to require that messages not be sent in clear text. You must ensure that all messages are encrypted when traveling across the network. What should you do?

- A. Set the ProtectionLevel property on the service contract and update the binding attribute in the endpoint element of the configuration file to wsHttpBinding.
- B. Set the ProtectionLevel property on the service contract and update the bindingConfiguration attribute in the endpoint element of the configuration file to webHttpBinding.
- C. Set the PrincipalPermissionAttribute on the service contract and update the binding attribute in the endpoint element of the configuration file to wsHttpBinding.
- D. Set the PrincipalPermissionAttribute on the service contract and update the bindingConfiguration attribute in the endpoint element of the configuration file to wsHttpBinding.

Answer: A

Question: 33.

You are creating a Windows Communication Foundation (WCF) service to process orders. The data contract for the order is defined as follows.

```
[DataContract]
public class Order
{
    &
    [DataMember]
    public string CardHolderName { get; set; }
    [DataMember]
    public string CreditCardNumber { get; set; }
}
```

You have the following requirements:

Enable the transmission of the contents of Order from the clients to the service. Ensure that the contents of CreditCardNumber are not sent across the network in clear text. Ensure that the contents of CreditCardNumber are accessible by the service to process the order. You need to implement the service to meet these requirements. What should you do?

- A. Add a DataProtectionPermission attribute to the CreditCardNumber property and set the ProtectData property to true.
- B. Convert the DataContract to a MessageContract and set the ProtectionLevel property to EncryptAndSign.
- C. Change the data type of CreditCardNumber from string to SecureString.
- D. Implement the CreditCardNumber property getter and setter. In the setter, run the value of the CreditCardNumber through the MD5CryptoServiceProvider class TransformBlock method.

Answer: B

Question: 34.

You are creating a Windows Communication Foundation (WCF) service based on WSHttpBinding. New audit requirements dictate that callers must be authenticated on every call to ensure that their credentials have not been revoked. You need to ensure that the service will not cache the security request token. What should you do?

- A. Apply a ServiceBehavior attribute to the service implementation class with the InstanceContextMode property set to Single.
- B. In the message security configuration, change clientCredentialType from IssuedToken to UserName.
- C. In the message security configuration, set establishSecurityContext to false.
- D. At the end of every operation, call the SessionStateUtility.RaiseSessionEnd method.

Answer: C

Question: 35.

You are creating a Windows Communication Foundation (WCF) service that accepts messages from clients when they are started. The message is defined as follows.

```
[MessageContract]
public class Agent
{
    public string CodeName { get; set; }
    public string SecretHandshake { get; set; }
}
```

You have the following requirements:

The CodeName property must be sent in clear text. The service must be able to verify that the property value was not changed after being sent by the client. The SecretHandshake property must not be sent in clear text and must be readable by the service. What should you do?

- A. Add a MessageBodyMember attribute to the CodeName property and set the ProtectionLevel to Sign. Add a MessageBodyMember attribute to the SecretHandshake property and set the ProtectionLevel to EncryptAndSign.
- B. Add a DataProtectionPermission attribute to the each property and set the ProtectData property to true.
- C. Add an XmlText attribute to the CodeName property and set the DataType property to Signed. Add a PasswordPropertyText attribute to the SecretHandshake property and set its value to true.
- D. Add an ImmutableObject attribute to the CodeName property and set its value property to true. Add a Browsable attribute to the SecretHandshake property and set its value to false.

Answer: A

Question: 36.

You are creating an ASP.NET Web application that hosts several Windows Communication Foundation (WCF) services. The services have ASP.NET Compatibility Mode enabled. Users authenticate with the Web application by using a cookie-based ASP.NET Forms Authentication model.

You add a service file named Authentication.svc that contains the following code segment.

```
<%@ ServiceHost
    Service="System.Web.Application Services.AuthenticationService"
    Factory="System.Web.ApplicationServices.ApplicationServicesHostFactory" %>
```

You need to ensure that users can access the WCF services without having to re-authenticate.

Which two configuration settings should you add (Each is part of a complete solution. Choose two.)

- A. In the system.web.extensions/scripting/webServices/authenticationService element, set the enabled attribute to true.
- B. In the system.web.extensions/scripting/webServices/profileService element, set the enabled attribute to true.

C. Add a service endpoint with `basicHttpBinding` for the contract `System.Web.ApplicationServices.AuthenticationService`.

D. Add a custom service behavior named `AuthenticationServiceTypeBehaviors` with a `serviceAuthenticationManager` element that has `serviceAuthenticationManagerType` set to `System.Web.Security.SqlMembershipProvider`.

Answer: AC

Question: 37.

A self-hosted Windows Communication Foundation (WCF) service uses a secure HTTP binding with a custom principal permission mode. The binding requires users to provide their Windows logon credentials. You need to retrieve the identity of the caller. What are two possible properties you can use to achieve this goal (Each correct answer presents a complete solution? Choose two.)

- A. `Thread.CurrentPrincipal.Identity.Name`
- B. `HttpContext.Current.User.Identity.Name`
- C. `ServiceSecurityContext.Current.PrimaryIdentity.Name`
- D. `OperationContext.Current.ServiceSecurityContext.PrimaryIdentity.Name`

Answer: CD

Question: 38.

You are creating a Windows Communication Foundation (WCF) service that implements the following service contract.

```
[ServiceContract]
public interface IOrderProcessing
{
    [OperationContract]
    void ApproveOrder(int id);
}
```

You need to ensure that only users with the Manager role can call the `ApproveOrder` method. What should you do?

- A. In the method body, check the `Rights.PossessProperty` property to see if it contains `Manager`.
- B. Add a `PrincipalPermission` attribute to the method and set the `Roles` property to `Manager`.
- C. Add a `SecurityPermission` attribute to the method and set the `SecurityAction` to `Demand`.
- D. In the method body, create a new instance of `WindowsClaimSet`. Use the `FindClaims` method to locate a `claimType` named `Role` with a `right` named `Manager`.

Answer: B

Question: 39.

You are developing a Windows Communication Foundation (WCF) service. The service needs to access out-of-process resources. You need to ensure that the service accesses these resources on behalf of the originating caller. What should you do?

- A. Set the value of `ServiceSecurityContext.Current.WindowsIdentity.ImpersonationLevel` to `TokenImpersonationLevel.Impersonation`.
- B. Set the value of `ServiceSecurityContext.Current.WindowsIdentity.ImpersonationLevel` to `TokenImpersonationLevel.Delegation`.
- C. Set the `PrincipalPermissionAttribute` on the service contract and update the binding attribute in the endpoint element of the configuration file to `wsHttpBinding`.
- D. Set the `PrincipalPermissionAttribute` on the service contract and update the `bindingConfiguration` attribute in the endpoint element of the configuration file to `wsHttpBinding`.

Answer: B

Question: 40.

A Windows Communication Foundation (WCF) service that handles corporate accounting must be changed to comply with government regulations of auditing and accountability. You need to configure the WCF service to execute under the Windows logged-on identity of the calling application. What should you do?

- A. Within the service configuration, add a `serviceAuthorization` behavior to the service, and set `impersonateCallerForAllOperations` to `true`.
- B. Within the service configuration, add a `serviceAuthenticationManager` behavior to the service, and set `serviceAuthenticationManagerType` to `Impersonate`.
- C. Within the service configuration, add a `serviceSecurityAudit` behavior to the service, and set `serviceAuthorizationAuditLevel` to `SuccessOrFailure`.
- D. Within the service configuration, add a `serviceCredentials` behavior to the service, and set `type` to `Impersonate`.

Answer: A

Question: 41.

A Windows Communication Foundation (WCF) service is required to log all authorization attempts to the Windows Event Log. You need to configure a behavior and apply it to the service to support this requirement. Which behavior should you configure and apply

- A. serviceAuthenticationManager
- B. serviceAuthorization
- C. serviceCredentials
- D. serviceSecurityAudit

Answer: D

Question: 42.

You have a secured Windows Communication Foundation (WCF) service. You need to track unsuccessful attempts to access the service. What should you do?

- A. Set the serviceAuthorizationManagerType attribute of the serviceAuthorization behavior to Message.
- B. Set the includeExceptionDetailInFaults attribute of the serviceDebug behavior to true.
- C. Set the Mode attribute of the security configuration element to Message.
- D. Set the messageAuthenticationAuditLevel attribute of the serviceSecurityAudit behavior to Failure.

Answer: D

Question: 43.

A Windows Communication Foundation (WCF) solution uses the following contract to share a message across its clients. (Line numbers are included for reference only.)

```
01[ServiceContract]
02public interface ITeamMessageService
03{
04[OperationContract]
05string GetMessage();
06
07[OperationContract]
08void PutMessage(string message);
09}
```

The code for the service class is as follows.

```

10 public class TeamMessageService : ITeamMessageService
11 {
12     Guid key = Guid.NewGuid();
13     string message = "Today's Message";
14     public string GetMessage()
15     {
16         return string.Format("Message:{0}. Key:{1}",
17             message, key);
18     }
19     public void PutMessage(string message)
20     {
21         this.message = message;
22     }
23 }

```

The service is self-hosted. The hosting code is as follows.

```

24 ServiceHost host =
new ServiceHost(typeof(TeamMessageService));
25 BasicHttpBinding binding =
new BasicHttpBinding(BasicHttpSecurityMode.None);
26 host.AddServiceEndpoint(
"MyApplication.ITeamMessageService", binding,
"http://localhost:12345");
27 host.Open();

```

You need to ensure that all clients calling GetMessage will retrieve the updated string if the message is updated by any client calling PutMessage. What should you do?

A. Add the following attribute to the TeamMessageService class, before line 10.

```
[ServiceBehavior(InstanceContextMode = InstanceContextMode.Single)]
```

B. Add the following attribute to the TeamMessageService class, before line 10.

```
[ServiceBehavior(InstanceContextMode =
InstanceContextMode.PerSession)]
```

Then change the binding definition on the service at line 25, and on the client to the following.

```
WSHttpBinding binding = new WSHttpBinding(SecurityMode.None);
```

```
binding.ReliableSession.Enabled = true;
```

C. Pass a service instance to the instancing code in line 24, as follows.

```
ServiceHost host = new ServiceHost(new TeamMessageService());
```

D. Redefine the message string in line 13, as follows.

```
static string message = "Today's Message";
```

Then change the implementation of PutMessage in lines 19-22 to the following.

```
public void PutMessage(string message)
```

```
{
```

```
TeamMessageService.message = message;
```

```
}
```

Answer: A

Question: 44.

A Windows Communication Foundation (WCF) solution exposes the following service over a TCP binding. (Line numbers are included for reference only.)

```
01 [ServiceContract]
02 [ServiceBehavior(ConcurrencyMode =
  ConcurrencyMode.Multiple)]
03 public class DataAccessService
04 {
05 [OperationContract]
06 public void PutMessage(string message)
07 {
08 MessageDatabase.PutMessage(message);
09 }
10 [OperationContract]
11 public string[] SearchMessages(string search)
12 {
13 return MessageDatabase.SearchMessages(search);
14 }
15 }
```

MessageDatabase supports a limited number of concurrent executions of its methods. You need to change the service to allow up to the maximum number of executions of the methods of MessageDatabase. This should be implemented without preventing customers from connecting to the service. What should you do?

A. Change the service behavior as follows.

```
[ServiceBehavior(ConcurrencyMode = ConcurrencyMode.Multiple,
InstanceContextMode = InstanceContextMode.Single)]
```

B. Change the service behavior as follows.

```
[ServiceBehavior(ConcurrencyMode = ConcurrencyMode.Single,
InstanceContextMode = InstanceContextMode.PerSession)]
```

C. Add a throttling behavior to the service, and configure the maxConcurrentCalls.

D. Add a throttling behavior to the service, and configure the maxConcurrentSessions.

Answer: C

Question: 45.

A service implements the following contract. (Line numbers are included for reference only.)

```
01 [ServiceContract(SessionMode = SessionMode.Required)]
02 public interface IContosoService
03 {
04 [OperationContract(IsOneWay = true, IsInitiating = true)]
05 void OperationOne(string value);
```

```
06
07 [OperationContract(IsOneWay = true, IsInitiating = false)]
08 void OperationTwo(string value);
09 }
```

The service is implemented as follows.

```
20 class ContosoService : IContosoService
21 {
22 public void OperationOne(string value) { & }
23
24 public void OperationTwo(string value) { & }
25 }
```

ContosoService uses NetMsmqBinding to listen for messages. The queue was set up to use transactions for adding and removing messages. You need to ensure that OperationOne and OperationTwo execute under the same transaction context when they are invoked in the same session. What should you do?

A. Insert the following attribute to OperationOne on IContosoService.

```
[TransactionFlow(TransactionFlowOption.Mandatory)]
```

Insert the following attribute to OperationTwo on IContosoService.

```
[TransactionFlow(TransactionFlowOption.Mandatory)]
```

B. Insert the following attribute to OperationOne on ContosoService.

```
[OperationBehavior(TransactionScopeRequired = true,
TransactionAutoComplete = false)]
```

Insert the following attribute to OperationTwo on ContosoService.

```
[OperationBehavior(TransactionScopeRequired = true,
TransactionAutoComplete = true)]
```

C. Add the following XML segment to the application config file in the system.serviceModel/bindings configuration section.

```
<netMsmqBinding>
<binding name="contosoTx" durable="true" receiveContextEnabled="true" />
</netMsmqBinding>
```

Then use the NetMsmqBinding named contosoTx to listen for messages from the clients.

D. Add the following XML segment to the application config file in the system.serviceModel/bindings configuration section.

```
<customBinding>
<binding name="contosoTx">
<transactionFlow />
<binaryMessageEncoding />
<msmqTransport durable="true" />
</binding>
</customBinding>
```

Then use the CustomBinding named contosoTx to listen for messages from the clients.

Answer: B

Question: 46.

A WCF service code is implemented as follows. (Line numbers are included for reference only.)

```
01 [ServiceContract]
02 [ServiceBehavior(InstanceContextMode =
03 InstanceContextMode.Single)]
04 public class CalculatorService
05 {
06 [OperationContract]
07 public double Calculate(double op1, string op, double op2)
08 {
09     &
10     24 }
11     25 }
```

You need to decrease the response time of the service. What are two possible ways to achieve this goal (Each correct answer presents a complete solution. Choose two.)

A. Change the service behavior to the following.

```
[ServiceBehavior(
InstanceContextMode = InstanceContextMode.Single,
ConcurrencyMode = ConcurrencyMode.Multiple)]
```

B. Change the service behavior to the following.

```
[ServiceBehavior(InstanceContextMode = InstanceContextMode.PerCall)]
```

C. Require the clients use threads, the Parallel Task Library, or other mechanism to issue service calls in parallel.

D. Require the clients to use async operations when calling the service.

Answer: AB

Question: 47.

You are creating a Windows Communication Foundation (WCF) service application. The application needs to service many clients and requests simultaneously. The application also needs to ensure subsequent individual client requests provide a stateful conversation. You need to configure the service to support these requirements. Which attribute should you add to the class that is implementing the service?

A. [ServiceBehavior (InstanceContextMode = InstanceContextMode.PerSession , ConcurrencyMode = ConcurrencyMode.Single)]

B. [ServiceBehavior (InstanceContextMode = InstanceContextMode.PerCall , ConcurrencyMode = ConcurrencyMode.Reentrant)]

- C. [ServiceBehavior (InstanceContextMode = InstanceContextMode.PerSession , ConcurrencyMode = ConcurrencyMode.Multiple)]
- D. [ServiceBehavior (InstanceContextMode = InstanceContextMode.PerCall , ConcurrencyMode = ConcurrencyMode.Multiple)]

Answer: C

Question: 48.

A Windows Communication Foundation (WCF) service only accepts messages that are signed and encrypted. A client application is not receiving expected responses from the service. You need to enable logging to verify that the messages from the client are signed and encrypted. You also need to see what each message looks like before the message body is deserialized into a .NET object. What should you do?

A. Configure the System.ServiceModel trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>
<messageLogging
logEntireMessage="true"
logMessagesAtServiceLevel="true"
logMessagesAtTransportLevel="true" />
</diagnostics>
```

B. Configure the System.ServiceModel trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>
<messageLogging
logEntireMessage="true"
logMessagesAtServiceLevel="true" />
</diagnostics>
```

C. Configure the System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>
<messageLogging
logEntireMessage="true"
logMessagesAtServiceLevel="true"
logMessagesAtTransportLevel="true" />
</diagnostics>
```

D. Configure the System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>
<messageLogging
logMessagesAtServiceLevel="true"
logMessagesAtTransportLevel="true" />
</diagnostics>
```

Answer: C

Question: 49.

Your Windows Communication Foundation (WCF) client application uses HTTP to communicate with the service. You need to enable message logging and include all security information such as tokens and nonces in logged messages. What should you do?

A. In the application configuration file, add the `logKnownPii` attribute to the message logging diagnostics source and set the value of the attribute to true.

Generate the `ContosoService` class using the Add Service Reference wizard.

Add a reference to `System.ServiceModel.Routing.dll`.

Add the following code segment.

```
ContosoService client = new ContosoService();
```

```
SoapProcessingBehavior behavior = new SoapProcessingBehavior();
```

```
behavior.ProcessMessages = true;
```

```
client.Endpoint.Behaviors.Add(behavior);
```

B. In the application configuration file, add the following XML segment to the `system.serviceModel` configuration section group.

```
<diagnostics>
```

```
<messageLogging logMessagesAtTransportLevel="true"
```

```
logEntireMessage="true" />
```

```
</diagnostics>
```

C. In the machine configuration file, add the following XML segment to the `system.serviceModel` configuration section.

```
<machineSettings enableLoggingKnownPii="true" />
```

Generate the `ContosoService` class using the Add Service Reference wizard.

Add the following code segment.

```
ContosoService client = new ContosoService();
```

```
client.Endpoint.Behaviors.Add(new CallbackDebugBehavior(true));
```

D. In the machine configuration file, add the following XML segment to the `system.serviceModel` configuration section.

```
<machineSettings enableLoggingKnownPii="true" />
```

In the application configuration file, add the `logKnownPii` attribute to the message logging diagnostics source and set the value of the attribute to true.

In the application configuration file, add the following XML segment to the `system.serviceModel` configuration section group.

```
<diagnostics>
```

```
<messageLogging logMessagesAtTransportLevel="true"/>
```

```
</diagnostics>
```

Answer: D

Question: 50.

A Windows Communication Foundation (WCF) service has the following contract.

```
[ServiceContract(Namespace="http://contoso.com")]
public interface IShipping
{
    [OperationContract]
    string DoWork(int id);
}
```

This is one of several service contracts hosted by your application. All endpoints use SOAP 1.2 bindings with WS-Addressing 1.0. The System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section is configured with one listener. You need to make sure that only the messages that are returned from the DoWork operation are logged. Which XML segment should you add to the system.serviceModel/diagnostics/messageLogging/filters configuration element?

- A. `<add xmlns:addr="http://www.w3.org/2005/08/addressing">
//addr:Action[text() =
'http://contoso.com/IShipping/DoWorkResponse']
</add>`
- B. `<add xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
//soap:Action[text() =
'http://contoso.com/IShipping/DoWorkResponse']
</add>`
- C. `<add xmlns:addr="http://www.w3.org/2005/08/addressing">
//addr:Action[text() =
'http://contoso.com/IShipping/DoWork']
</add>`
- D. `<add xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
//soap:Action[text() =
'http://contoso.com/IShipping/DoWork']
</add>`

Answer: A

Question: 51.

You are moving a Windows Communication Foundation (WCF) service into production. You need to be able to monitor the health of the service. You only want to enable all performance counter instances exposed by the ServiceModelService 4.0.0.0 counter group. Which element should you add to the system.serviceModel section in the application configuration file?

- A. <diagnostics performanceCounters="ServiceOnly" />
- B. <diagnostics wmiProviderEnabled="true" performanceCounters="Off" />
- C. <diagnostics performanceCounters="All" />
- D. <diagnostics wmiProviderEnabled="true" />

Answer: A

Question: 52.

You are using tracing to diagnose run-time issues. When you look at the traces for the service in SvcTraceViewer.exe, you see what is shown in the exhibit. (Click the Exhibit button.) The exception trace is selected in SvcTraceViewer.exe. You need to interpret the trace results to determine where the error occurred and what to do next. What should you do?

A. This issue occurred in the ServiceHost during ServiceHost.Open(). Enable WMI by adding the following configuration to the system.serviceModel configuration section in the application configuration file.

```
<diagnostics wmiProviderEnabled="true" />
```

Restart the application and inspect the endpoints visible through WMI.

B. This issue occurred in the ServiceHost during ServiceHost.Open(). Compare the security settings for any endpoints that use an MSMQ transport to the security configuration of the MSMQ queue used by the endpoint.

C. This issue occurred at the ServiceHost when receiving a message. Compare the security configurations on the client and server to make sure that they are compatible.

D. This issue occurred at the ServiceHost when accepting an initial set of messages from MSMQ. Log all messages sent between the client and the server.

Answer: B

Question: 53.

You are implementing a Windows Communication Foundation (WCF) service contract named IContosoService in a class named ContosoService. The service occasionally fails due to an exception being thrown at the service. You need to send the stack trace of any unhandled exceptions to clients as a fault message. What should you do?

A. In the application configuration file on the client, add the following XML segment to the system.serviceModel/behaviors configuration section group.

```
<endpointBehaviors>  
  <behavior name=" debug ">  
    < callback Debug includeExceptionDetailInFaults="true" />  
  </behavior>  
</endpointBehaviors>
```

Associate the debug behavior with any endpoints that need to return exception details.

B. In the application configuration file on the service and all the clients, add the following XML segment to the system.diagnostics/sources configuration section group.

```
<source name="System.ServiceModel" switchValue="Error" propagateActivity="true">
  <listeners>
  <add name="ServiceModelTraceListener"
  initializeData="app_tracelog.svclog"
  type="System.Diagnostics.XmlWriterTraceListener" />
  </listeners>
</source>
```

C. Apply the following attribute to the ContosoService class.

```
[ServiceBehavior(IncludeExceptionDetailInFaults = true)]
```

D. For each OperationContract exposed by IContosoService, apply the following attribute.

```
[FaultContract(typeof(Exception))]
```

Answer: C

Question: 54.

A Windows Communication Foundation (WCF) application exposes a service as a SOAP endpoint for consumption by cross-platform clients. During integration testing, you find that one of the clients is not generating the correct messages to the WCF application. In order to debug the issue and fix the communication, you need to configure the service to log messages received from the client. What should you do?

A. Set an etwTracking behavior on the service and configure a listener for the System.ServiceModel trace source.

B. Set an etwTracking behavior on the service and configure a listener for the System.ServiceModel.MessageLogging trace source.

C. Enable messageLogging in the System.ServiceModel diagnostics element configuration and configure a listener for the System.ServiceModel.MessageLogging trace source.

D. Enable messageLogging in the System.ServiceModel diagnostics element configuration and configure a listener for the System.ServiceModel trace source.

Answer: C

Question: 55.

A Windows Communication Foundation (WCF) service interacts with the database of a workflow engine. Data access authorization is managed by the database, which raises security exceptions if a user is unauthorized to access it. You need to ensure that the application transmits the exceptions raised by the database to the client that is calling the service. Which behavior should you configure and apply to the service?

- A. routing
- B. serviceDebug
- C. serviceSecurityAudit
- D. workflowUnhandledException

Answer: B

Question: 56.

You develop a Windows Communication Foundation (WCF) service to generate reports. Client applications call the service to initiate report generation but do not wait for the reports to be generated. The service does not provide any status to the client applications.

The service class is defined as follows. (Line numbers are included for reference only.)

```
01[ServiceContract]
02public class ReportGeneratorService
03{
04
05private int GenerateReports(int clientID)
06{
07&
08return 0;
09}
10}
```

You need to ensure that client applications can initiate reports without waiting for status. Which two actions should you perform (Each correct answer presents part of the solution. Choose two.)

- A. Insert the following code at line 04.
[OperationContract(IsOneWay=true)]
- B. Insert the following code at line 04.
[OperationContract(AsyncPattern=false)]
- C. At line 05, change the GenerateReports method from private to public.
- D. Remove line 08. At line 05, change the return type of GenerateReports method to void.

Answer: AD

Question: 57.

Your company has a Windows Communication Foundation (WCF) service at the URL `http://services.contoso.com/OrderLookupService.svc`.

The `<system.serviceModel>` section of the configuration file is as follows. (Line numbers are included for reference only.)

```
01<system.serviceModel>
02<behaviors>
03<serviceBehaviors>
04<behavior>
05 <serviceDebug
   includeExceptionDetailInFaults="false"/>
06
07</behavior>
08</serviceBehaviors>
09</behaviors>
10 <serviceHostingEnvironment
   multipleSiteBindingsEnabled="true" />
11 </system.serviceModel>
```

You need to ensure that the service publishes the WSDL description at `http://services.contoso.com/OrderLookupService.svcwsdl`. What should you do?

A. Change the `serviceDebug` element at line 05 as follows.

```
<serviceDebug includeExceptionDetailInFaults="true"/>
```

B. Insert the following element at line 06.

```
<serviceDiscovery>
<announcementEndpoints>
  <endpoint name="wsdlAnnouncement"
    kind="udpAnnouncementEndpoint" />
</announcementEndpoints>
</serviceDiscovery>
```

C. Insert the following element at line 06.

```
<serviceMetadata httpGetEnabled="true" />
```

D. Insert the following element at line 06.

```
<serviceMetadata httpGetEnabled="false" />
```

Answer: C

Question: 58.

You are developing a Windows Communication Foundation (WCF) service. The service operation takes a customer number as the only argument and returns information about the customer. The service requires a security token in the header of the message. You need to create a message contract for the service. Which code segment should you use?

A. [ServiceContract]

```
public interface IService
{
    [OperationContract]
    CustomerInformation GetCustomerInformation(Header header,
        int customerNumber);
}
```

[DataContract]

```
public class CustomerInformation
{
    &
}
```

[MessageContract]

```
public class Header
{
    [MessageHeader]
    public string SecurityTag;
}
```

B. [ServiceContract]

```
public interface IService
{
    [OperationContract]
    CustomerInformation GetCustomerInformation(Header header,
        int customerNumber);
}
```

[MessageContract]

```
public class CustomerInformation
{
    &
}
```

[MessageContract]

```
public class Header
{
    [MessageHeader]
    public string SecurityTag;
}
```

C. [ServiceContract]

```
public interface IService
{
```



```
[OperationContract]
CustomerInformation GetCustomerInformation(
    CustomerNumber request);
}
[DataContract]
public class CustomerInformation
{
    &
}
[MessageContract]
public class CustomerNumber
{
    [MessageHeader]
    public string SecurityTag;
    [MessageBodyMember]
    public int CustomerNumberElement;
}
D. [ServiceContract]
public interface IService
{
    [OperationContract]
    CustomerInformation GetCustomerInformation(
        CustomerNumber request);
}
[MessageContract]
public class CustomerInformation
{
    &
}
[MessageContract]
public class CustomerNumber
{
    [MessageHeader]
    public string SecurityTag;
    [MessageBodyMember]
    public int CustomerNumberElement;
}
```

Answer: D

Question: 59.

You are developing a Windows Communication Foundation (WCF) service that contains the following operation contract.

```
[OperationContract]
```

```
CustomerNames GetCustomerNames();
```

The operation returns customer names.

You need to develop a definition for the operation contract that produces XML with the following structure.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header />
  <s:Body>
    <Names xmlns=http://tempuri.org/
      xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
      xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <a:string>Customer1</a:string>
      <a:string>Customer2</a:string>
      <a:string>Customer3</a:string>
    </Names>
  </s:Body>
</s:Envelope>
```

Which code segment should you use?

A. [DataContract(IsWrapped = false)]

```
public class CustomerNames
{
  [MessageBodyMember]
  public string[] Names;
}
```

B. [DataContract(WrapperName = "")]

```
public class CustomerNames
{
  [MessageBodyMember]
  public string[] Names;
}
```

C. [DataContract]

```
public class CustomerNames
{
  [DataMember]
  public string[] Names;
}
```

D. [DataContract]

```
public class CustomerNames
{
  [DataMember(IsRequired = false)]
  public string[] Names;
}
```

Answer: A

Question: 60.

A Windows Communication Foundation (WCF) service handles online order processing for your company.

You discover that many requests are being made with invalid account numbers. You create a class named `AccountNumberValidator` that has a method named `Validate`. Before the message is processed, you need to validate account numbers with `AccountNumberValidator` and reject messages with invalid account numbers. You create a new class that implements the `IParameterInspector` interface. Which code segment should you use in this class?

```
A. public void AfterCall(string operationName,
    object[] outputs,
    object returnValue,
    object correlationState)
{
    string accountNumber = GetAccountNumber(outputs);
    var validator = new AccountNumberValidator();
    if(!validator.Validate(accountNumber))
    {
        throw new FaultException();
    }
}
public object BeforeCall(string operationName,
    object[] inputs)
{
    return null;
}

B. public void AfterCall(string operationName,
    object[] outputs,
    object returnValue,
    object correlationState)
{
    return;
}
public object BeforeCall(string operationName,
    object[] inputs)
{
    string accountNumber = GetAccountNumber(inputs);
    var validator = new AccountNumberValidator();
    if (!validator.Validate(accountNumber))
    {
        throw new FaultException();
    }
    return null;
}
```

```

C. public void AfterCall(string operationName,
    object[] outputs,
    object returnValue,
    object correlationState)
{
    string accountNumber = GetAccountNumber(outputs);
    var validator = new AccountNumberValidator();
    if(!validator.Validate(accountNumber))
    {
        returnValue = new FaultException();
    }
}
public object BeforeCall(string operationName,
    object[] inputs)
{
    return null;
}
D. public void AfterCall(string operationName,
    object[] outputs,
    object returnValue,
    object correlationState)
{
    return;
}
public object BeforeCall(string operationName,
    object[] inputs)
{
    string accountNumber = GetAccountNumber(inputs);
    var validator = new AccountNumberValidator();
    if (!validator.Validate(accountNumber))
    {
        return new FaultException();
    }
}

```

Answer: B

Question: 61.

You are creating a client application and configuring it to call a Windows Communication Foundation (WCF) service. When the application is deployed, it will be configured to send all messages to a WCF routing service. You need to ensure that the application can consume the target service after the application is deployed. What should you do?

- A. In the client application, add a service reference to the router service. In the client binding configuration, specify the address of the router service.
- B. In the client application, add a service reference to the target service. In the client binding configuration, specify the address of the target service.
- C. In the client application, add a service reference to the router service. In the client binding configuration, specify the address of the target service.
- D. In the client application, add a service reference to the target service. In the client binding configuration, specify the address of the router service.

Answer: D

Question: 62.

You create a service and deploy it on a network in a building named Building1. You will deploy the service to Building2.

The service in Building1 is configured using the following discovery scopes.

```
<scopes>
  <add
    scope="http://contoso.com/Chicago/Building1"/>
  <add
    scope="ldap:///ou=Building1,ou=Chicago,o=contoso,c=us"/>
</scopes>
```

The service in Building2 will be configured using the following discovery scopes.

```
<scopes>
  <add
    scope="http://contoso.com/Chicago/Building2"/>
  <add
    scope="ldap:///ou=Building2,ou=Chicago,o=contoso,c=us"/>
</scopes>
```

You need to ensure that the client application can discover the service in Building1 or the service in Building2. Which scopes should you add to the client configuration file?

- A. <scopes>
<add scope="http://contoso.com/Chicago/*"/>
</scopes>
- B. <scopes>
<add scope="http://contoso.com/Chicago"/>
</scopes>
- C. <scopes>
<add
 scope="ldap:///ou=Building,ou=Chicago,o=contoso,c=us"/>
</scopes>
- D. <scopes>
<add
 scope="ldap:///ou=*,o=contoso,c=us"/>
</scopes>

Answer: B

Question: 63.

You are modifying a Windows Communication Foundation (WCF) service that issues security tokens. The service is accessible through the named pipe protocol. No endpoints are added in the service code. The configuration file for the service is as follows. (Line numbers are included for reference only.)

```
01<configuration>
02<system.serviceModel>
03<services>
04<service name="Contoso.TokenService">
05
06<host>
07<baseAddresses>
08
09<add
  baseAddress="net.pipe://www.contoso.com/tokenpipe" />
10</baseAddresses>
11</host>
12</service>
13</services>
14</system.serviceModel>
15 </configuration>
```

You need to ensure that new and existing client applications can access the service through HTTP and named pipes. What should you do?

- A. Insert the following line at line 05.
<endpoint address=http://www.contoso.com
binding="wsHttpBinding"
contract="Contoso.TokenService" />
- B. Insert the following line at line 05.
<endpoint address=http://www.contoso.com
binding="basicHttpBinding"
contract="Contoso.TokenService" />
- C. Insert the following line at line 08.
<add baseAddress="http://www.contoso.com" />
- D. Insert the following line at line 08.
<add baseAddress="net.tcp://www.contoso.com:8090" />

Answer: C

Question: 64.

Your company has an existing Windows Communication Foundation (WCF) service that allows business partners to place orders. The service uses netMsmqBinding. You find that processing every order in its own transaction is causing a delay. You need to ensure that the service is configured to process multiple orders in one transaction. What should you do?

- A. Use <serviceThrottling> service behavior and set the maxConcurrentCalls attribute.
- B. Use <transactedBatching> endpoint behavior and set the batchSize attribute.
- C. Use <dispatcherSynchronizationBehavior> endpoint behavior and set the maxPendingReceives attribute.
- D. Use <synchronousReceive> endpoint behavior.

Answer: B

Question: 65.

You are developing a Windows Communication Foundation (WCF) service that contains the following code segment.

```
[ServiceContract ]
public interface ICustomerService
{
    &
}
public class CustomerService : ICustomerService
{
    &
}
```

The service is self-hosted in a console application. Older client applications access the service at `http://contoso.com:8080/CustomerService/V1`. Newer client applications access the service at `http://contoso.com:8080/CustomerService/V2`. You need to ensure that any client application can access the service at either address. Which code segment should you use?

- A. `Uri serviceAddress1 = new Uri("http://contoso.com:8080/CustomerService/V1");`
`Uri serviceAddress2 = new Uri("http://contoso.com:8080/CustomerService/V2");`
`ServiceHost host = new ServiceHost(typeof(ICustomerService), new Uri[] { serviceAddress1, serviceAddress2 });`
- B. `Uri serviceAddress1 = new Uri("http://contoso.com:8080/CustomerService/V1");`
`Uri serviceAddress2 =`

```
new Uri("http://contoso.com:8080/CustomerService/V2");
ServiceHost host =
new ServiceHost(typeof(CustomerService),
new Uri[] { serviceAddress1, serviceAddress2 });
C. Uri serviceAddress =
new Uri("http://contoso.com:8080/");
ServiceHost host =
new ServiceHost(typeof(CustomerService),
new Uri[] { serviceAddress });
host.AddServiceEndpoint(typeof(ICustomerService),
new BasicHttpBinding(), "CustomerService/V1");
host.AddServiceEndpoint(typeof(ICustomerService),
new BasicHttpBinding(), "CustomerService/V2");
D. Uri serviceAddress =
new Uri("http://contoso.com:8080/");
ServiceHost host =
new ServiceHost(typeof(ICustomerService),
new Uri[] { serviceAddress });
host.AddServiceEndpoint(typeof(CustomerService),
new BasicHttpBinding(), "CustomerService/V1");
host.AddServiceEndpoint(typeof(CustomerService),
new BasicHttpBinding(), "CustomerService/V2");
```

Answer: C

Question: 66.

You are hosting a Windows Communication Foundation (WCF) service under Microsoft Internet Information Services (IIS) 7.0.

You have set up a Web site in IIS Manager. The physical path is C:\wwwroot\Calendar. There is a Calendar.svc file in the C:\wwwroot\Calendar folder. It contains the following directive.

```
<%@ ServiceHost Language="C#" Debug="true" Service="Calendar.Calendar"
CodeBehind="Calendar.svc.cs" %>
```

The Calendar.svc.cs file contains the source for the Calendar class in the Calendar namespace. You compile this code into the Calendar.dll file. You need to deploy your service to the Web site. What should you do?

- A. Copy the Calendar.dll file to the C:\wwwroot\Calendar\code folder.
- B. Copy the Calendar.dll file to the C:\wwwroot\Calendar\bin folder.
- C. Copy the Calendar.svc.cs file to the C:\wwwroot\Calendar\bin folder.
- D. Copy the Calendar.svc.cs file to the C:\wwwroot\Calendar\code folder.

Answer: B

Question: 67.

You are developing a client application that uses the following code to consume a Windows Communication Foundation (WCF) service. (Line numbers are included for reference only.)

```
01BasicHttpBinding myBinding = new BasicHttpBinding();
02EndpointAddress myEndpointAddress =
  new EndpointAddress(
  "http://contoso.com/TaxService.svc");
03
04ITaxService client = channelFactory.CreateChannel();
05string data = client.GetData(1);
You need to consume the service.
Which code segment should you insert at line 03?
```

- A. `var channelFactory = new ChannelFactory<ITaxService>();`
- B. `var channelFactory = new ChannelFactory<ITaxService>(myBinding);`
- C. `var channelFactory = new ChannelFactory<ITaxService>(myBinding, myEndpointAddress);`
- D. `var channelFactory = new ChannelFactory<ITaxService>("http://contoso.com/TaxService.svc");`

Answer: C

Question: 68.

You need to modify a client application that consumes a Windows Communication Foundation (WCF) service. The service metadata is no longer available. You need to modify the previously generated proxy to include asynchronous calls to the service. What should you do?

- A. Update the service reference with the Generate asynchronous operations option.
- B. Create a partial class for the previously generated proxy and include the new asynchronous methods.
- C. Create a class with the same name as the previously generated proxy and add the new asynchronous methods. Add the new class to a namespace that is different from the original proxy.
- D. Create a class with the same name as the previously generated proxy and add the new asynchronous methods as partial methods. Add the new class to a namespace that is different from the original proxy.

Answer: B

Question: 69.

You develop a Windows Communication Foundation (WCF) service that interacts with Microsoft Message Queuing (MSMQ). The service requires sessions. You need to create a custom binding that enables messages sent to the queue to be viewed when you are using a listener tool. Which binding elements should you use?

- A. textMessageEncoding and msmqTransport in this order
- B. textMessageEncoding and msmqIntegrationTransport in this order
- C. msmqTransport and textMessageEncoding in this order
- D. msmqIntegrationTransport and textMessageEncoding in this order

Answer: A

Question: 70.

You are developing a client application that consumes a Windows Communication Foundation (WCF) service.

You use the svcutil.exe utility to create a proxy for the service. You use the svcutil.exe switches that generate asynchronous calls. GetFlight is a service operation that takes no parameters and returns a string. The GetFlightCallback method must be called when the service operation returns.

You create an instance of the client proxy with the following code.

```
var client = new TravelServiceClient();
```

You need to ensure that a callback is received when the GetFlight operation is called asynchronously. Which code segment should you use?

- A. `client.BeginGetFlight(GetFlightCallback, null);`
`client.GetFlight();`
- B. `client.GetFlight();`
`client.BeginGetFlight(GetFlightCallback, null);`
- C. `client.GetFlightCompleted +=`
`new EventHandler<GetFlightCompletedEventArgs>(`
`GetFlightCallback);`
`client.GetFlightAsync();`
- D. `IAsyncResult asyncResult = client.BeginGetFlight(`
`GetFlightCallback, client);`
`client.EndGetFlight(asyncResult);`

Answer: C

Question: 71.

You are creating a Windows Communication Foundation (WCF) service. The service endpoints change frequently. On the service, you add a new `ServiceDiscoveryBehavior` to the `Behaviors` collection of the `ServiceHost` `Description` property. You need to ensure that client applications can communicate with the service and discover changes to the service endpoints. What should you do?

- A. Add a new `ServiceDiscoveryBehavior` to the `Behaviors` collection in the client application.
- B. Add a new `AnnouncementClient` to the `Behaviors` collection in the client application.
- C. Use the `FindCriteria` class and the `UdpDiscoveryEndpoint` class to set up the binding in the client application.
- D. Use the `DiscoveryProxy` class and the `EndpointDiscoveryMetadata` class to set up the binding in the client application.

Answer: C

Question: 72.

You are hosting a Windows Communication Foundation (WCF) service at `http://www.contoso.com` for a law enforcement agency. The agency adds operations to support sending biometric fingerprint data via non-buffered streaming. The service data is not routed between intermediaries. The WCF binding you are using by default does not support encryption. You need to ensure that fingerprint data is not disclosed when it is passed over the network. What should you do?

- A. Use `basicHttpBinding` with message security to `https://www.contoso.com`.
- B. Use `basicHttpBinding` over transport security at `https://www.contoso.com`.
- C. Use `wsHttpBinding` over message security at `https://www.contoso.com`.
- D. Use `wsHttpBinding` over transport security at `http://www.contoso.com`.

Answer: B

Question: 73.

You are maintaining a Windows Communication Foundation (WCF) service that uses a custom `UserNamePassword` class to authenticate clients. The service certificate is hosted in the deployment server store for trusted root certificate authorities and has a `Subject` value of `TaxServiceKey`. Other service certificates hosted on the server also use `TaxServiceKey` as a `Subject` value. You need to ensure that the service identifies itself with a certificate whose subject name and distinguished names are `TaxServiceKey`. Which code segment should you use?

- A. HostInstance.Credentials.ServiceCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.My, X509FindType.FindBySubjectName, "CN=TaxServiceKey");
- B. HostInstance.Credentials.ServiceCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.AuthRoot, X509FindType.FindBySubjectName, "CN=TaxServiceKey");
- C. HostInstance.Credentials.ServiceCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.My, X509FindType.FindBySubjectDistinguishedName, "CN=TaxServiceKey");
- D. HostInstance.Credentials.ServiceCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.Root, X509FindType.FindBySubjectDistinguishedName, "CN=TaxServiceKey");

Answer: D

Question: 74.

You are developing a Windows Communication Foundation (WCF) service that returns location information for authorized law enforcement agencies. The service contract is as follows.

```
[ServiceContract]
public interface IMappingService
{
    [OperationContract]
    long[] GetLocationCoordinates(String cityName);
    [OperationContract]
    long[] GetLocationOfCitizen(String ssn);
}
```

Users are authenticated and impersonated. The system uses ASP.NET roles. The members of law enforcement are members of the LawEnforcement role. You need to ensure that only members of the LawEnforcement role can call these methods. What are two possible ways to achieve this goal (Each correct answer presents a complete solution? Choose two.)

- A. Add a PrincipalPermissionAttribute to each method that should be available only to members of law enforcement. Set its SecurityAction to Demand and set the role equal to LawEnforcement.
- B. Use the CurrentPrincipal property of the thread. Call the IsInRole method specifying LawEnforcement as a parameter.
- C. Create a GenericPrincipal specifying Thread.CurrentPrincipal.Identity as the IIdentityParameter and LawEnforcement as the only value for the Roles parameter.
- D. At the beginning of each method, enumerate each ClaimSet in a new WindowsClaimSet. Use the FindClaims method to locate a claim type named Role with a right named LawEnforcement.

Answer: AB

Question: 75.

You develop a Windows Communication Foundation (WCF) service that employees use to access bonus information. You define the following service contract. (Line numbers are included for reference only.)

```
01[ServiceContract(SessionMode = SessionMode.Required)]
02public interface IFinancialService
03{
04[OperationContract]
05string Login(int employeeID, string passwordHash);
06
07[OperationContract]
08double GetBonus(int month);
09
10[OperationContract(IsTerminating = true)]
11void Logout();
12 }
```

Client applications can invoke methods without logging in. You need to ensure that the client applications invoke Login before invoking any other method. You also need to ensure that client applications cannot consume the service after invoking Logout. Which two actions should you perform (Each correct answer presents part of the solution. Choose two.)

- A. Replace line 04 with the following code.
[OperationContract(IsInitiating = false)]
- B. Replace line 04 with the following code.
[OperationContract(IsInitiating = true,
IsTerminating = true)]
- C. Replace line 07 with the following code.
[OperationContract(IsInitiating = false)]
- D. Replace line 10 with the following code.
[OperationContract(IsInitiating = false,
IsTerminating = true)]

Answer: CD

Question: 76.

You are developing a Windows Communication Foundation (WCF) service that allows customers to update financial data. The service contract is defined as follows. (Line numbers are included for reference only.)

```
01[ServiceContract]
02public interface IDataUpdate
03{
04[OperationContract]
05[TransactionFlow(TransactionFlowOption.Mandatory)]
06void Update(string accountNumber, double amount);
07}
08
09class UpdateService : IDataUpdate
10{
11[OperationBehavior(TransactionScopeRequired = true,
TransactionAutoComplete = false)]
12public void Update(string accountNumber,
double amount)
13{
14&
15}
16}
17
```

You need to ensure that the service is invoked within a transaction. What should you do?

A. Replace line 01 with the following code.

```
[ServiceContract(
    SessionMode = SessionMode.NotAllowed)]
```

B. Replace line 01 with the following code.

```
[ServiceContract(
    SessionMode = SessionMode.Required)]
```

C. Insert the following code at line 08.

```
[ServiceBehavior(
    TransactionAutoCompleteOnSessionClose = false)]
```

D. Insert the following code at line 08.

```
[ServiceBehavior(
    ReleaseServiceInstanceOnTransactionComplete = false)]
```

Answer: B

Question: 77.

You are developing a Windows Communication Foundation (WCF) service that is hosted by a Windows Forms application. The ServiceHost instance is created in the Form constructor. You need to ensure that the service is not blocked while the UI thread is busy. What should you do?

- A. Decorate the service implementation class with the following line of code.
[ServiceBehavior(UseSynchronizationContext = false)]
- B. Decorate the service implementation class with the following line of code.
[ServiceBehavior(ConcurrencyMode = ConcurrencyMode.Multiple)]
- C. Call the Invoke method of the form and supply a delegate.
- D. Call the BeginInvoke method of the form and supply a delegate.

Answer: A

Question: 78.

You are developing a Windows Communication Foundation (WCF) service. You must record all available information for the first 1,000 messages processed, even if they are malformed. You need to configure the message logging section of the configuration file. Which configuration segment should you use?

- A. `<messageLogging logEntireMessage="true" logMalformedMessages="true" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="true" maxMessagesToLog="1000"/>`
- B. `<messageLogging logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="true" maxMessagesToLog="1000"/>`
- C. `<messageLogging logEntireMessage="false" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="false" maxMessagesToLog="1000"/>`
- D. `<messageLogging logMalformedMessages="true" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="false" maxMessagesToLog="1000"/>`

Answer: A

Question: 79.

You are developing a Windows Communication Foundation (WCF) service. The service configuration file has a <System.Diagnostics> element defined. You need to ensure that all security audit information, trace logging, and message logging failures are recorded. Which configuration segment should you add to the <System.Diagnostics> element

A. <sources>

```
<source name="System.ServiceModel"
  switchValue="Information, ActivityTracing"
  propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
</source>
<source name="System.ServiceModel.MessageLogging"
  propagateActivity="true">
<listeners>
<add name="text" />
</listeners>
</source>
</sources>
```

B. <sources>

```
<source name="System.ServiceModel"
  switchValue="Information, ActivityTracing"
  propagateActivity="true" />
<source name="System.ServiceModel.MessageLogging"
  propagateActivity="true" />
</sources>
```

C. <sources>

```
<source name="System.ServiceModel"
  switchValue="Information, ActivityTracing"
  propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
</source>
<source name="System.ServiceModel.MessageLogging"
  propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
</source>
</sources>
<sharedListeners>
<add name="xml"
```



```
type="System.Diagnostics.XmlWriterTraceListener"
initializeData="& " />
</sharedListeners>
D. <sources>
<source name="System.ServiceModel"
switchValue="Information, ActivityTracing"
propagateActivity="true" />
<source name="System.ServiceModel.MessageLogging"
propagateActivity="true" />
</sources>
<sharedListeners>
<add name="xml"
type="System.Diagnostics.XmlWriterTraceListener"
initializeData="& " />
</sharedListeners>
```

Answer: C

Question: 80.

You develop a Windows Communication Foundation (WCF) service. You enable all performance counters and run multiple calls to the service. The service must isolate session data for each user. You need to monitor the instancing behavior used in the service. Which performance counter should you monitor?

- A. ServiceModelService 4.0.0.0\Calls
- B. ServiceModelService 4.0.0.0\Instances
- C. ASP.NET State Service\State Server Sessions Active
- D. ASP.NET State Service\State Server Sessions Total

Answer: B

Question: 81.

You create a Windows Communication Foundation (WCF) service and deploy it with wsHttpBinding and message security enabled. You create an intermediate WCF service for logging messages sent to the primary service. The intermediate service is called via the clientVia endpoint behavior. The primary service is receiving malformed data from a client application. You need to enable inspection of the malformed data and prevent message tampering. What should you do?

- A. Specify a protection level of None in the service contract for the intermediate service. Disable message and transport security from the client application configuration file.
- B. Specify a protection level of Sign in the service contract for the intermediate service. Disable transport security from the client application configuration file.
- C. Modify the binding on the intermediate service to use netNamedPipeBinding.
- D. Modify the binding on the intermediate service to use webHttpBinding.

Answer: B

70-513VB

Question: 1

You are creating a Windows Communication Foundation (WCF) service that is implemented as follows. (Line numbers are included for reference only.)

```
01 <ServiceContract(>  
02 <ServiceBehavior(IncludeExceptionDetailInFaults:=True)>  
03 Public Class OrderService  
04  
05 <OperationContract(>  
06 Public Sub SubmitOrder(ByVal anOrder As Order)  
07  
08 Try  
09 &  
10 Catch ex As DivideByZeroException  
11  
12 End Try  
13 End Sub  
14  
15 End Class
```

You need to ensure that the stack trace details of the exception are not included in the error information sent to the client. What should you do?

A. Replace line 11 with the following line.

```
Throw
```

B. Replace line 11 with the following line.

```
Throw New FaultException(Of Order)(anOrder, ex.ToString())
```

After line 05, add the following line.

```
<FaultContract(GetType(FaultException(Of Order)))>
```

C. Replace line 11 with the following line.

```
Throw ex
```

After line 05, add the following line.

```
<FaultContract(GetType(FaultException(Of Order)))>
```

D. Replace line 11 with the following line.

```
Throw New FaultException(Of Order)(  
anOrder, "Divide by zero exception")
```

Answer: D

Question: 2

You are creating a Windows Communication Foundation (WCF) service. You do not want to expose the internal implementation at the service layer. You need to expose the following class as a service named Arithmetic with an operation named Sum.

```
Public Class Calculator
Public Function Add(ByVal x As Integer,
ByVal y As Integer) As Integer
End Function
End Class
```

Which code segment should you use?

A. <ServiceContract(Namespace:="Arithmetic")>

```
Public Class Calculator
<OperationContract(Action:="Sum")>
Public Function Add(ByVal x As Integer,
ByVal y As Integer) As Integer
&
End Function
End Class
```

B. <ServiceContract(ConfigurationName:="Arithmetic")>

```
Public Class Calculator
<OperationContract(Action:="Sum")>
Public Function Add(ByVal x As Integer,
ByVal y As Integer) As Integer
&
End Function
End Class
```

C. <ServiceContract(Name:="Arithmetic")>

```
Public Class Calculator
<OperationContract(Name:="Sum")>
Public Function Add(ByVal x As Integer,
ByVal y As Integer) As Integer
&
End Function
End Class
```

D. <ServiceContract(Name:="Arithmetic")>

```
Public Class Calculator
<OperationContract(ReplyAction:="Sum")>
Public Function Add(ByVal x As Integer,
ByVal y As Integer) As Integer
&
End Function
End Class
```

Answer: C

Question: 3

You are developing a data contract for a Windows Communication Foundation (WCF) service. The data in the data contract must participate in round trips. Strict schema validity is not required. You need to ensure that the contract is forward-compatible and allows new data members to be added to it. Which interface should you implement in the data contract class?

- A. ICommunicationObject
- B. IExtension<T>
- C. IExtensibleObject<T>
- D. IExtensibleDataObject

Answer: D

Question: 4

A Windows Communication Foundation (WCF) application uses the following data contract.

```
<DataContract(>  
Public Class Person  
<DataMember(>  
Public firstName As String  
<DataMember(>  
Public lastName As String  
<DataMember(>  
Public age As Integer  
<DataMember(>  
Public ID As Integer  
End Class
```

You need to ensure that the following XML segment is generated when the data contract is serialized.

```
<Person>  
<firstName xsi:nil= " true " />  
<lastName xsi:nil= " true " />  
<ID>999999999</ID>  
</Person>
```

Which code segment should you use?

- A. <DataMember(>
Public firstName As String
<DataMember(>
Public lastName As String
<DataMember(EmitDefaultValue:=True)>

Public age As Integer = 0
<DataMember(EmitDefaultValue:=True)>
Public ID As Integer = 999999999
B. <DataMember(EmitDefaultValue:=False)>
Public firstName As String = Nothing
<DataMember(EmitDefaultValue:=False)>
Public lastName As String = Nothing
<DataMember(EmitDefaultValue:=True)>
Public age As Integer = -1
<DataMember(EmitDefaultValue:=False)>
Public ID As Integer = 999999999
C. <DataMember(EmitDefaultValue:=True)>
Public firstName As String
<DataMember(EmitDefaultValue:=True)>
Public lastName As String
<DataMember(EmitDefaultValue:=False)>
Public age As Integer = -1
<DataMember(EmitDefaultValue:=False)>
Public ID As Integer = 999999999
D. <DataMember(>
Public firstName As String = Nothing
<DataMember(>
Public lastName As String = Nothing
<DataMember(EmitDefaultValue:=False)>
Public age As Integer = 0
<DataMember(EmitDefaultValue:=False)>
Public ID As Integer = 999999999

Answer: D

Question: 5

You are developing a client that sends several types of SOAP messages to a Windows Communication Foundation (WCF) service method named PostData.

PostData is currently defined as follows.

```
<OperationContract(>
```

```
Sub PostData(ByVal data As Order)
```

You need to modify PostData so that it can receive any SOAP message.

Which code segment should you use?

A. <OperationContract(IsOneWay:=True, Action:="", ReplyAction:="")>

```
Sub PostData(ByVal data As Order)
```

B. <OperationContract(IsOneWay:=True, Action:="", ReplyAction:="")>

```
Sub PostData(ByVal data As BodyWriter)
```

C. <OperationContract(>

```
Sub PostData(ByVal data As BodyWriter)
```

D. <OperationContract(>
Sub PostData(ByVal data As Message)

Answer: D

Question: 6

A class named TestService implements the following interface.

```
<ServiceContract(>  
Public Interface ITestService  
<OperationContract(>  
Function GetServiceTime() As DateTime  
End Interface
```

TestService is hosted in an ASP.NET application.

You need to modify the application to allow the GetServiceTime method to return the data formatted as JSON. It must do this only when the request URL ends in /ServiceTime. What should you do?

A. Add this attribute to the GetServiceTime method.

```
<WebInvoke(Method:="POST")>
```

In the web.config file, add this element to system.serviceModel/behaviors/endpointBehaviors.

```
<behavior name="Json">  
<enableWebScript />  
</behavior>
```

In the web.config file, configure TestService in the system.serviceModel/services collection as follows.

```
<service name="TestService">  
<endpoint address="/ServiceTime"  
contract="TestService"  
behaviorConfiguration="Json"  
binding="webHttpBinding" />  
</service>
```

B. Add this attribute to the GetServiceTime method.

```
<WebInvoke(Method:="GET",  
UriTemplate:="/ServiceTime",  
ResponseFormat:=WebMessageFormat.Json)>
```

In the web.config file, configure TestService in the system.serviceModel/services collection as follows.

```
<service name="TestService">  
<endpoint address="/ServiceTime"  
contract="TestService"  
binding="webHttpBinding" />  
</service>
```

C. Add this attribute to the GetServiceTime method.

```
<WebGet(  
ResponseFormat:=WebMessageFormat.Json,
```

UriTemplate:="/ServiceTime")>

Create a new .svc file named JsonVersion.svc with the following content.

```
<%@ ServiceHost Service="TestService"
```

```
Factory="System.ServiceModel.Activation.WebServiceHostFactory" %>
```

D. Add this attribute to the GetServiceTime method.

```
<WebGet(UriTemplate:="{Json}/ServiceTime")>
```

Create a new .svc file named JsonVersion.svc with the following content.

```
<%@ ServiceHost Service="TestService"
```

```
Factory="System.ServiceModel.Activation.WebServiceHostFactory" %>
```

Answer: C

Question: 7

You are creating a Windows Communication Foundation (WCF) service that implements operations in a RESTful manner. You need to add a delete operation.

You implement the delete method as follows.

```
Sub DeleteItems(ByVal id As String)
```

You need to configure WCF to call this method when the client calls the service with the HTTP DELETE operation.

What should you do?

- A. Add the WebInvoke(UriTemplate:="/Items/{id}", Method:="DELETE") attribute to the operation.
- B. Add the HttpDelete attribute to the operation.
- C. Replace the string parameter with a RemovedActivityAction parameter.
- D. Change the Sub statement to Function and specify RemovedActivityAction as the return type.

Answer: A

Question: 8

You are creating a Windows Communication Foundation (WCF) service that responds using plain-old XML (POX).

You have the following requirements:

You must enable the /catalog.svc/items operation to respond using the POX, JSON, or ATOM formats. You also must ensure that the same URL is used regardless of the result type.

You must determine the response format by using the Accepts HTTP header.

What should you do?

- A. Implement the IChannelInitializer interface in the service class.
- B. Implement the System.Runtime.Serialization.IFormatterConverter interface in the service class.
- C. Set the BodyStyle parameter of the WebGet attribute on the operation to WebMessageBodyStyle.WrappedResponse.

D. Set the return type of the operation to `System.ServiceModel.Channels.Message`. Use the current `WebOperationContext` methods to return the data in the required format.

Answer: D

Question: 9

A Windows Communication Foundation (WCF) solution uses two services to manage a shopping cart. Service A processes messages containing line items that total between \$0 and \$500. Service B processes messages containing line items that total more than \$500. All messages are of equal importance to the business logic. You need to route incoming messages to the appropriate services by using WCF routing. Which two message filters should you add to the router? (Each correct answer presents part of the solution. Choose two.)

- A. a message filter with a priority of 100 that will forward messages that total between \$0 and \$500 to Service A
- B. a message filter with a priority of 0 that will forward messages that total between \$0 and \$500 to Service A
- C. a message filter with a priority of 0 that will forward all messages to Service B
- D. a message filter with a priority of 100 that will forward all messages to Service B

Answer: A C

Question: 10

You have an existing Windows Communication Foundation (WCF) service. You need to ensure that other services are notified when the service is started. What should you do?

- A. Add the following standard endpoint to the service.
`<endpoint name="udpAnnouncementEndpoint" kind="udpDiscoveryEndpoint" />`
- B. Add the following standard endpoint to the service.
`<endpoint name="udpDiscoveryEndpoint" kind="udpAnnouncementEndpoint" />`
- C. Add a service behavior with the following element.
`<serviceDiscovery>
<announcementEndpoints>
<endpoint kind="udpDiscoveryEndpoint" />
</announcementEndpoints>
</serviceDiscovery>`
- D. Add a service behavior with the following element.
`<serviceDiscovery>
<announcementEndpoints>`

```
<endpoint kind="udpAnnouncementEndpoint" />
</announcementEndpoints>
</serviceDiscovery>
```

Answer: D

Question: 11

You are developing a Windows Communication Foundation (WCF) service that reads messages from a public non-transactional MSMQ queue. You need to configure the service to read messages from the failed-delivery queue. Which URI should you specify in the endpoint configuration settings of the service?

- A. net.msmq://localhost/msmq\$;FailedMessages
- B. net.msmq://localhost/msmq\$;DeadLetter
- C. net.msmq://localhost/system\$;DeadXact
- D. net.msmq://localhost/system\$;DeadLetter

Answer: D

Question: 12

You have an existing Windows Communication Foundation (WCF) service that exposes a service contract over HTTP using explicit binding configuration. You need to expose that contract over HTTP and TCP. What should you do?

- A. Add a net.tcp base address to the host.
- B. Add an endpoint configured with a netTcpBinding.
- C. Add an endpoint behavior named netTcpBehavior to the existing endpoint.
- D. Add a binding configuration to the existing endpoint named netTcpBinding.

Answer: B

Question: 13

You have an existing Windows Communication Foundation (WCF) Web service. The Web service is not responding to messages larger than 64 KB. You need to ensure that the Web service can accept messages larger than 64 KB without generating errors. What should you do?

- A. Increase the value of maxReceivedMessageSize on the endpoint binding.
- B. Increase the value of maxRequestLength on the httpRuntime element.
- C. Increase the value of maxBufferSize on the endpoint binding.
- D. Increase the value of maxBufferPoolSize on the endpoint binding.

Answer: A

Question: 14

A Windows Communication Foundation (WCF) service is responsible for transmitting XML documents between systems. The service has the following requirements:

It must minimize the transmission size by attaching the XML document as is without using escape characters or base64 encoding. It must interoperate with systems that use SOAP but are not built on the .NET platform. You need to configure the service to support these requirements. Which message encoding should you use?

- A. Binary message encoding
- B. MTOM (Message Transmission Optimization Mechanism) message encoding
- C. Text message encoding with message version set to none
- D. Text message encoding with message version set to SOAP 1.2

Answer: B

Question: 15

You are modifying an existing Windows Communication Foundation (WCF) service that is defined as follows.

```
< ServiceContract ()>
Public Interface IMessageProcessor
<OperationContract ()>
Sub ProcessMessage ()
End Interface
Public Class MessageProcessor
Implements IMessageProcessor
Public Sub ProcessMessage () _
```

```
Implements IMessageProcessor.ProcessMessage
&
SubmitOrder ()
&
End Sub
End Class
```

SubmitOrder makes a call to another service. The ProcessMessage method does not perform as expected under a heavy load. You need to enable processing of multiple messages. New messages must only be processed when the ProcessMessage method is not processing requests, or when it is waiting for calls to SubmitOrder to return. Which attribute should you apply to the MessageProcessor class?

- A. CallbackBehavior (ConcurrencyMode := ConcurrencyMode.Reentrant)
- B. CallbackBehavior (ConcurrencyMode := ConcurrencyMode.Multiple)
- C. ServiceBehavior (ConcurrencyMode := ConcurrencyMode.Reentrant)
- D. ServiceBehavior (ConcurrencyMode := ConcurrencyMode.Multiple)

Answer: C

Question: 16

You are adding a Windows Communication Foundation (WCF) service to an existing application. The application is configured as follows. (Line numbers are included for reference only.)

```
01 <configuration>
02 <system.serviceModel>
03 <services>
04 <service name="Contoso.Sales.StockService"
05 behaviorConfiguration="MetadataBehavior">
06 <host>
07 <baseAddresses>
08 <add
baseAddress="http://contoso.com:8080/StockService" />
09 </baseAddresses>
10 </host>
11 </service>
12 </services>
13 <behaviors>
14 <serviceBehaviors>
15 <behavior name="MetadataBehavior">
16 </behavior>
17 </serviceBehaviors>
18 </behaviors>
```

You need to configure the service to publish the service metadata. Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

- A. Add the following XML segment between lines 10 and 11.

```
<endpoint address=""  
binding="mexHttpBinding"  
contract="IMetadataExchange"  
>
```

B. Add the following XML segment between lines 10 and 11.

```
<endpoint address=""  
binding="basicHttpBinding"  
contract="IMetadataExchange"  
>
```

C. Add the following XML segment between lines 15 and 16.

```
<serviceDiscovery>  
<announcementEndpoints>  
<endpoint address=""/>  
</announcementEndpoints>  
</serviceDiscovery>
```

D. Add the following XML segment between lines 15 and 16

```
<serviceMetadata httpGetEnabled="true"/>
```

Answer: A D

Question: 17

Four Windows Communication Foundation (WCF) services are hosted in Microsoft Internet Information Services (IIS). No behavior configuration exists in the web.config file. You need to configure the application so that every service and endpoint limits the number of concurrent calls to 50 and the number of concurrent sessions to 25. Which XML segment should you add to the system.serviceModel configuration section of the web.config file?

A. <behaviors>

```
<serviceBehaviors>  
<behavior name="*">  
<serviceThrottling maxConcurrentCalls="50" maxConcurrentSessions="25"/>  
</behavior>  
</serviceBehaviors>  
</behaviors>
```

B. <behaviors>

```
<serviceBehaviors>  
<behavior name="default">  
<serviceThrottling maxConcurrentCalls="50" maxConcurrentSessions="25"/>  
</behavior>  
</serviceBehaviors>  
</behaviors>
```

C. <behaviors>

```
<serviceBehaviors>  
<behavior name="">  
<serviceThrottling maxConcurrentCalls="50" maxConcurrentSessions="25"/>
```

```
</behavior>
</serviceBehaviors>
</behaviors>
D. <behaviors>
<serviceBehaviors>
<behavior name="ALL">
<serviceThrottling maxConcurrentCalls="50" maxConcurrentSessions="25"/>
</behavior>
</serviceBehaviors>
</behaviors>
```

Answer: C

Question: 18

A Windows Communication Foundation (WCF) service is self-hosted in a console application. The service implements the `ITimeService` service interface in the `TimeService` class. You need to configure the service endpoint for HTTP communication. How should you define the service and endpoint tags?

A. Define the service tag as follows.

```
<service name="ITimeService">
  Define the endpoint tag as follows.
  < endpoint kind="TimeService"
  address="http://localhost:8080/TimeService"
  binding="wsHttpBinding"
  contract="ITimeService"/>
```

B. Define the service tag as follows.

```
<service name="TimeService">
  Define the endpoint tag as follows.
  < endpoint kind="TimeService"
  address="http://localhost:8080/TimeService"
  binding="wsHttpBinding"
  contract="ITimeService"/>
```

C. Define the service tag as follows.

```
<service name="ITimeService">
  Define the endpoint tag as follows.
  < endpoint name="TimeService"
  address="http://localhost:8080/TimeService"
  binding="wsHttpBinding"
  contract="ITimeService"/>
```

D. Define the service tag as follows.

```
<service name="TimeService">
  Define the endpoint tag as follows.
  <endpoint address="http://localhost:8080/TimeService"
  binding="wsHttpBinding"
```

```
contract="ITimeService"/>
```

Answer: D

Question: 19

A Windows Communication Foundation (WCF) service is self-hosted in a console application. The service implements the IDataAccess contract, which is defined in the MyApplication namespace. The service is implemented in a class named DataAccessService, which implements the IDataAccess interface and also is defined in the MyApplication namespace.

The hosting code is as follows. (Line numbers are included for reference only.)

```
01 Shared Sub Main(ByVal args() As String)
02
03 Dim host As ServiceHost
04
05 host.Open()
06 Console.ReadLine()
07 host.Close()
08
09 End Sub
```

You need to create a ServiceHost instance and assign it to the host variable. You also need to instantiate the service host. Which line of code should you insert at line 04?

- A. host = New ServiceHost("MyApplication.DataAccessService")
- B. host = New ServiceHost("MyApplication.IDataAccess")
- C. host = New ServiceHost(GetType(IDataAccess))
- D. host = New ServiceHost(GetType(DataAccessService))

Answer: D

Question: 20

A Windows Communication Foundation (WCF) service implements the following contract.

```
<ServiceContract(>
Public Interface IHelloService
<OperationContract(>
<WebGet(UriTemplate:="hello?name={name}")>
Function SayHello(ByVal name As String) As String
End Interface
```

The implementation is as follows.

```
Public Class HelloService
Implements IHelloService
Public Function SayHello(ByVal name As String) As String _
```

Implements IHelloService.SayHello

Return "Hello " & name

End Function

End Class

The service is self-hosted, and the hosting code is as follows.

```
Dim svcHost As WebServiceHost = CreateHost()
```

```
svcHost.Open()
```

```
Console.ReadLine()
```

```
svcHost.Close()
```

You need to implement CreateHost so that the service has a single endpoint hosted at http://localhost:8000/HelloService. Which code segment should you use?

A.

```
Dim svcHost As WebServiceHost = New WebServiceHost(GetType(HelloService))
```

```
svcHost.AddServiceEndpoint(GetType(IHelloService),
```

```
New WebHttpBinding(WebHttpSecurityMode.None),
```

```
"http://localhost:8000/HelloService")
```

```
Return svcHost
```

Dim baseAddress As Uri = New Uri("http://localhost:8000/")

B.

```
Dim svcHost As WebServiceHost =
```

```
New WebServiceHost(GetType(HelloService), baseAddress)
```

```
svcHost.AddServiceEndpoint(GetType(IHelloService),
```

```
New WebHttpBinding(WebHttpSecurityMode.None),
```

```
"HelloService")
```

```
Return svcHost
```

C.

```
Dim svcHost As WebServiceHost = New WebServiceHost(New HelloService())
```

```
svcHost.AddServiceEndpoint(GetType(IHelloService),
```

```
New WebHttpBinding(WebHttpSecurityMode.None),
```

```
"http://localhost:8000/HelloService")
```

```
Return svcHost
```

Dim baseAddress As Uri = New Uri("http://localhost:8000/")

D.

```
Dim svcHost As WebServiceHost =
```

```
New WebServiceHost(New HelloService(), baseAddress)
```

```
svcHost.AddServiceEndpoint(GetType(IHelloService),
```

```
New WebHttpBinding(WebHttpSecurityMode.None),
```

```
"HelloService")
```

```
Return svcHost
```

Answer: A

Question: 21

You are developing a Windows Communication Foundation (WCF) service that will be hosted in Microsoft Internet Information Services (IIS) 7.0. The service must be hosted in an IIS application named Info. You need to enable this service to be hosted in IIS by changing the web.config file. Which XML segment should you add to the web.config file?

A. <serviceHostingEnvironment>
<serviceActivations>
<add relativeAddress="Info.svc" service="Info" />
</serviceActivations>
</serviceHostingEnvironment>

B. <serviceHostingEnvironment>
<serviceActivations>
<add relativeAddress="Info" service="Info.svc" />
</serviceActivations>
</serviceHostingEnvironment>

C. <serviceHostingEnvironment>
<transportConfigurationTypes>
<add name="Info" transportConfigurationType="Info.svc" />
</transportConfigurationTypes>
</serviceHostingEnvironment>

D. <serviceHostingEnvironment>
<transportConfigurationTypes>
<add name="Info.svc" transportConfigurationType="FileNotRequired" />
</transportConfigurationTypes>
</serviceHostingEnvironment>

Answer: A

Question: 22

You are creating a Windows Communication Foundation (WCF) service. You need to ensure that the service is compatible with ASP.NET to make use of the session state. Which binding should you use?

- A. NetTcp ContextBinding
- B. BasicHttpContextBinding
- C. NetTcp Binding
- D. NetMsmqBinding

Answer: B

Question: 23

A Windows Communication Foundation (WCF) service has a callback contract. You are developing a client application that will call this service. You must ensure that the client application can interact with the WCF service. What should you do?

- A. On the OperationContractAttribute, set the AsyncPattern property value to True.
- B. On the OperationContractAttribute, set the ReplyAction property value to the endpoint address of the client.
- C. On the client, create a proxy derived from DuplexClientBase(Of TChannel).
- D. On the client, use GetCallbackChannel(Of T).

Answer: C

Question: 24

You are creating a Windows Communication Foundation (WCF) service.

You have the following requirements:

Messages must be sent over TCP.

The service must support transactions.

Messages must be encoded using a binary encoding.

Messages must be secured using Windows stream-based security.

You need to implement a custom binding for the service.

In which order should the binding stack be configured?

- A. tcpTransport
windowsStreamSecurity
transactionFlow
binaryMessageEncoding
transactionFlow
binaryMessageEncoding
windowsStreamSecurity
- B. tcpTransport
windowsStreamSecurity
- C. tcpTransport
binaryMessageEncoding
transactionFlow
binaryMessageEncoding
transactionFlow
- D. tcpTransport
windowsStreamSecurity

Answer: B

Question: 25

A Windows Communication Foundation (WCF) client configuration file contains the following XML segment in the system.serviceModel element.

```
<client>
<endpoint address=" net.tcp://server/ContosoService "
binding=" netTcpBinding "
contract=" Contoso. IContoso Service "
name=" netTcp " />
<endpoint address=" net.pipe://localhost/ContosoService "
binding=" netNamedPipeBinding "
contract=" Contoso. IContoso Service "
name=" netPipe " />
</client>
```

You need to create a channel factory that can send messages to the endpoint listening at net.pipe://localhost/ContosoService. Which code segment should you use?

- A. Dim factory As ChannelFactory (Of Contoso. IContosoService) =
New ChannelFactory (Of Contoso. IContosoService)(" Contoso. IContosoService ")
- B. Dim factory As ChannelFactory (Of Contoso. IContosoService) =
New ChannelFactory (Of Contoso. IContosoService)(" netNamedPipeBinding ")
- C. Dim factory As ChannelFactory (Of Contoso. IContosoService) =
New ChannelFactory (Of Contoso. IContosoService)(" netPipe ")
- D. Dim factory As ChannelFactory (Of Contoso. IContosoService) =
New ChannelFactory (Of Contoso. IContosoService)(
" net.pipe://localhost/ContosoService ")

Answer: C

Question: 26

A Windows Communication Foundation (WCF) solution uses the following contracts. (Line numbers are included for reference only.)

```
01 <ServiceContract(CallbackContract:=GetType(INameService))>
02 Public Interface IGreetingService
03
04 <OperationContract()>
05 Function GetMessage() As String
06 End Interface
07
08 <ServiceContract()>
09 Public Interface INameService
```

```
10
11 <OperationContract(>
12 Function GetName() As String
13 End Interface
```

When the client calls GetMessage on the service interface, the service calls GetName on the client callback. In the client, the class NameService implements the callback contract.

The client channel is created as follows.

```
22 Dim callbackContext As InstanceContext =
New InstanceContext(New NameService("client"))
&
25 Dim factory As DuplexChannelFactory(Of IGreetingService) =
New DuplexChannelFactory(Of IGreetingService)(
GetType(NameService), binding, address)
26 Dim greetingService As IGreetingService =
factory.CreateChannel()
```

You need to ensure that the service callback is processed by the instance of NameService. What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two.)

A. Change line 25 to the following code segment.

```
Dim factory As DuplexChannelFactory(Of IGreetingService) =
New DuplexChannelFactory(Of IGreetingService)(
callbackContext, binding, address)
```

B. Change line 26 to the following code segment.

```
Dim greetingService As IGreetingService =
factory.CreateChannel(callbackContext)
```

C. Add the following code segment after line 26.

```
callbackContext.IncomingChannels.Add(
DirectCast(greetingService, IDuplexChannel))
```

D. Add the following code segment after line 26.

```
callbackContext.OutgoingChannels.Add(
DirectCast(greetingService, IDuplexChannel))
```

Answer: AB

Question: 27

A Windows Communication Foundation (WCF) service implements the following contract. (Line numbers are included for reference only.)

```
01 <ServiceContract(>
02 Public Interface IDataAccessService
03
04 <OperationContract(>
05 Sub PutMessage(ByVal message As String)
06
07 <OperationContract(>
```

```
08 <FaultContract(GetType(TimeoutFaultException))>
09 <FaultContract(GetType(FaultException))>
10 Function SearchMessages(ByVal search As String) As String ()
11
12 End Interface
```

The implementation of the SearchMessages method throws TimeoutFaultException exceptions for database timeouts. The implementation of the SearchMessages method also throws an Exception for any other issue it encounters while processing the request. These exceptions are received on the client side as generic FaultException exceptions. You need to implement the error handling code for SearchMessages and create a new channel on the client only if the channel faults. What should you do?

- A. Catch and handle both TimeoutFaultException and FaultException.
- B. Catch both TimeoutFaultException and FaultException. Create a new channel in both cases.
- C. Catch and handle TimeoutFaultException. Catch FaultException and create a new channel.
- D. Catch and handle FaultException. Catch TimeoutFaultException and create a new channel.

Answer: C

Question: 28

An ASP.NET application hosts a RESTful Windows Communication Foundation (WCF) service at /Services/Contoso.svc . The service provides a JavaScript resource to clients. You have an explicit reference to the JavaScript in your page markup as follows.

```
<script type="text/javascript" src="/Services/Contoso.svc/js" />
```

You need to retrieve the debug version of the service JavaScript. What should you do?

- A. In the <%@ ServiceHost %> header for /Services/Contoso.svc, set the Debug attribute to true.
- B. In the <%@ Page %> header, set the Debug attribute to true.
- C. In the script tag, add a debug attribute and set its value to true.
- D. In the script tag, append debug to the src attribute.

Answer: D

Question: 29

You are consuming a Windows Communication Foundation (WCF) service. The service interface is defined as follows.

```
<DataContract(Namespace:='')>
Public Class Item
&
End Class
<ServiceContract(Namespace:='')>
Public Interface ICatalog
<OperationContract(>
<WebInvoke(Method:="POST", UriTemplate:="/Item")>
Function UpdateItem(ByVal item As Item) As Item
End Interface
```

The client application receives a `WebResponse` named `response` with the response from the service. You need to deserialize this response into a strongly typed object representing the return value of the method. Which code segment should you use?

- A. `Dim s As DataContractSerializer = New DataContractSerializer(GetType(Item))`
`Dim item As Item = DirectCast(s.ReadObject(response.GetResponseStream()), Item)`
- B. `Dim f As BinaryFormatter = New BinaryFormatter()`
`Dim item As Item = DirectCast(f.Deserialize(response.GetResponseStream()), Item)`
- C. `Dim r As XmlDictionaryReader = JsonSerializerFactory.CreateJsonReader(response.GetResponseStream(), XmlDictionaryReaderQuotas.Max)`
- D. `Dim s As DataContractSerializer = New DataContractSerializer(GetType(Item))`
`Dim item As Item = DirectCast(s.ReadObject(r), Item)`
- E. `Dim s As DataContractJsonSerializer = New DataContractJsonSerializer(GetType(Item))`
`Dim item As Item = DirectCast(s.ReadObject(response.GetResponseStream()), Item)`

Answer: A

Question: 30

You are developing an application to update a user's social status. You need to consume the service using Windows Communication Foundation (WCF).

The client configuration is as follows.

```
<system.serviceModel>
<bindings>
<webHttpBinding>
<binding name="SocialConfig">
<security mode="TransportCredentialOnly">
<transport clientCredentialType="Basic"
realm="Social API" />
</security>
</binding>
</webHttpBinding>
</bindings>
<client>
<endpoint address=" http:// contoso .com "
binding="webHttpBinding"
bindingConfiguration="SocialConfig"
contract="ISocialStatus"
name="SocialClient" />
</client>
</system.serviceModel>
```

The service contract is defined as follows.

```
<ServiceContract(>
Public Interface ISocialStatus
<OperationContract(>
<WebInvoke(UriTemplate:="/statuses/update.xml?status={text}")>
Sub UpdateStatus(ByVal text As String)
End Interface
```

Which code segment should you use to update the social status?

```
A. Using factory As WebChannelFactory(Of ISocialStatus) =
New WebChannelFactory(Of ISocialStatus)("SocialClient")
factory.Credentials.UserName.UserName = user.Name
factory.Credentials.UserName.Password = user.Password
Dim socialChannel As ISocialStatus = factory.CreateChannel()
socialChannel.UpdateStatus(newStatus)
End Using
```

```
B. Using factory As ChannelFactory(Of ISocialStatus) =
New WebChannelFactory(Of ISocialStatus)(GetType(ISocialStatus))
factory.Credentials.UserName.UserName = user.Name
factory.Credentials.UserName.Password = user.Password
Dim socialChannel As ISocialStatus = factory.CreateChannel()
socialChannel.UpdateStatus(newStatus)
```

```

End Using
C. Using factory As ChannelFactory(Of ISocialStatus) =
New ChannelFactory(Of ISocialStatus)("POST")
factory.Credentials.Windows.ClientCredential.UserName =
user.Name
factory.Credentials.Windows.ClientCredential.SecurePassword.SetAt(
0, user.Password)
Dim socialChannel As ISocialStatus = factory.CreateChannel()
socialChannel.UpdateStatus(newStatus)
End Using
D. Using factory As WebChannelFactory(Of ISocialStatus) =
New WebChannelFactory(Of ISocialStatus)(GetType(ISocialClient))
factory.Credentials.Windows.ClientCredential.UserName =
user.Name
factory.Credentials.Windows.ClientCredential.SecurePassword.SetAt(
0, user.Password)
Dim socialChannel As ISocialStatus = factory.CreateChannel()
socialChannel.UpdateStatus(newStatus)
End Using

```

Answer: A

Question: 31

A Windows Communication Foundation (WCF) client application is consuming an RSS syndication feed from a blog. You have a SyndicationFeed variable named feed. The application iterates through the items as follows. (Line numbers are included for reference only.)

```

01 For Each item As SyndicationItem In feed.Items
02
03
04 Next

```

You need to display the content type and body of every syndication item to the console. Which two lines of code should you insert between lines 02 and 03?

- A. Console.WriteLine(item.Content.Type)
Console.WriteLine(DirectCast(item.Content, TextSyndicationContent).Text)
- B. Console.WriteLine(item.Content.GetType())
Console.WriteLine(DirectCast(item.Content, TextSyndicationContent).Text)
- C. Console.WriteLine(item.Content.Type)
Console.WriteLine(item.Content.ToString())
- D. Console.WriteLine(item.Content.GetType())
Console.WriteLine(item.Content.ToString())

Answer: A

Question: 32

The endpoint of a Windows Communication Foundation (WCF) service uses `basicHttpBinding` for its binding. Your company's policies have changed to require that messages not be sent in clear text. You must ensure that all messages are encrypted when traveling across the network. What should you do?

- A. Set the `ProtectionLevel` property on the service contract and update the binding attribute in the endpoint element of the configuration file to `wsHttpBinding`.
- B. Set the `ProtectionLevel` property on the service contract and update the `bindingConfiguration` attribute in the endpoint element of the configuration file to `webHttpBinding`.
- C. Set the `PrincipalPermissionAttribute` on the service contract and update the binding attribute in the endpoint element of the configuration file to `wsHttpBinding`.
- D. Set the `PrincipalPermissionAttribute` on the service contract and update the `bindingConfiguration` attribute in the endpoint element of the configuration file to `wsHttpBinding`.

Answer: A

Question: 33

You are creating a Windows Communication Foundation (WCF) service to process orders. The data contract for the order is defined as follows.

```
<DataContract ()>
Public Class Order
&
<DataMember ()>
Public Property CardHolderName As String
<DataMember ()>
Public Property CreditCardNumber As String
End Class
```

You have the following requirements:

Enable the transmission of the contents of `Order` from the clients to the service.

Ensure that the contents of `CreditCardNumber` are not sent across the network in clear text.

Ensure that the contents of `CreditCardNumber` are accessible by the service to process the order.

You need to implement the service to meet these requirements.

What should you do?

- A. Add a `DataProtectionPermission` attribute to the `CreditCardNumber` property and set the `ProtectData` property to `True`.
- B. Convert the `DataContract` to a `MessageContract` and set the `ProtectionLevel` property to `EncryptAndSign`.
- C. Change the data type of `CreditCardNumber` from `String` to `SecureString`.
- D. Implement the `CreditCardNumber` property getter and setter. In the setter, run the value of the `CreditCardNumber` through the `MD5CryptoServiceProvider` class `TransformBlock` method.

Answer: B

Question: 34

You are creating a Windows Communication Foundation (WCF) service based on WSHttpBinding. New audit requirements dictate that callers must be authenticated on every call to ensure that their credentials have not been revoked. You need to ensure that the service will not cache the security request token. What should you do?

- A. Apply a ServiceBehavior attribute to the service implementation class with the InstanceContextMode property set to Single.
- B. In the message security configuration, change clientCredentialType from IssuedToken to UserName.
- C. In the message security configuration, set establishSecurityContext to false.
- D. At the end of every operation, call the SessionStateUtility.RaiseSessionEnd method.

Answer: C

Question: 35

You are creating a Windows Communication Foundation (WCF) service that accepts messages from clients when they are started. The message is defined as follows.

```
<MessageContract(>  
Public Class Agent  
Public Property CodeName As String  
Public Property SecretHandshake As String  
End Class
```

You have the following requirements:

The CodeName property must be sent in clear text. The service must be able to verify that the property value was not changed after being sent by the client. The SecretHandshake property must not be sent in clear text and must be readable by the service. What should you do?

- A. Add a MessageBodyMember attribute to the CodeName property and set the ProtectionLevel to Sign. Add a MessageBodyMember attribute to the SecretHandshake property and set the ProtectionLevel to EncryptAndSign.
- B. Add a DataProtectionPermission attribute to the each property and set the ProtectData property to True.
- C. Add an XmlText attribute to the CodeName property and set the DataType property to Signed. Add a PasswordPropertyText attribute to the SecretHandshake property and set its value to True.
- D. Add an ImmutableObject attribute to the CodeName property and set its value property to True. Add a Browsable attribute to the SecretHandshake property and set its value to False.

Answer: A

Question: 36

You are creating an ASP.NET Web application that hosts several Windows Communication Foundation (WCF) services. The services have ASP.NET Compatibility Mode enabled. Users authenticate with the Web application by using a cookie-based ASP.NET Forms Authentication model.

You add a service file named Authentication.svc that contains the following code segment.

```
<%@ ServiceHost
```

```
Service="System.Web.ApplicationServices.AuthenticationService"
```

```
Factory="System.Web.ApplicationServices.ApplicationServicesHostFactory" %>
```

You need to ensure that users can access the WCF services without having to re-authenticate.

Which two configuration settings should you add? (Each is part of a complete solution. Choose two.)

- A. In the `system.web.extensions/scripting/webServices/authenticationService` element, set the `enabled` attribute to `true`.
- B. In the `system.web.extensions/scripting/webServices/profileService` element, set the `enabled` attribute to `true`.
- C. Add a service endpoint with `basicHttpBinding` for the contract `System.Web.ApplicationServices.AuthenticationService`.
- D. Add a custom service behavior named `AuthenticationServiceTypeBehaviors` with a `serviceAuthenticationManager` element that has `serviceAuthenticationManagerType` set to `System.Web.Security.SqlMembershipProvider`.

Answer: AC

Question: 37

A self-hosted Windows Communication Foundation (WCF) service uses a secure HTTP binding with a custom principal permission mode. The binding requires users to provide their Windows logon credentials. You need to retrieve the identity of the caller. What are two possible properties you can use to achieve this goal? (Each correct answer presents a complete solution. Choose two.)

- A. `Thread.CurrentPrincipal.Identity.Name`
- B. `HttpContext.Current.User.Identity.Name`
- C. `ServiceSecurityContext.Current.PrimaryIdentity.Name`
- D. `OperationContext.Current.ServiceSecurityContext.PrimaryIdentity.Name`

Answer: CD

Question: 38

You are creating a Windows Communication Foundation (WCF) service that implements the following service contract.

```
<ServiceContract(>
```

```
Public Interface IOrderProcessing
```

```
<OperationContract(>
```

```
Sub ApproveOrder(ByVal id As Integer)
```

```
End Interface
```

You need to ensure that only users with the Manager role can call the ApproveOrder method. What should you do?

- A. In the method body, check the Rights.PossessProperty property to see if it contains Manager.
- B. Add a PrincipalPermission attribute to the method and set the Roles property to Manager.
- C. Add a SecurityPermission attribute to the method and set the SecurityAction to Demand.
- D. In the method body, create a new instance of WindowsClaimSet. Use the FindClaims method to locate a claimType named Role with a right named Manager.

Answer: B

Question: 39

You are developing a Windows Communication Foundation (WCF) service. The service needs to access out-of-process resources. You need to ensure that the service accesses these resources on behalf of the originating caller. What should you do?

- A. Set the value of ServiceSecurityContext.Current.WindowsIdentity.ImpersonationLevel to TokenImpersonationLevel.Impersonation.
- B. Set the value of ServiceSecurityContext.Current.WindowsIdentity.ImpersonationLevel to TokenImpersonationLevel.Delegation.
- C. Set the PrincipalPermissionAttribute on the service contract and update the binding attribute in the endpoint element of the configuration file to wsHttpBinding.
- D. Set the PrincipalPermissionAttribute on the service contract and update the bindingConfiguration attribute in the endpoint element of the configuration file to wsHttpBinding.

Answer: B

Question: 40

A Windows Communication Foundation (WCF) service that handles corporate accounting must be changed to comply with government regulations of auditing and accountability. You need to configure the WCF service to execute under the Windows logged-on identity of the calling application. What should you do?

- A. Within the service configuration, add a serviceAuthorization behavior to the service, and set impersonateCallerForAllOperations to true.
- B. Within the service configuration, add a serviceAuthenticationManager behavior to the service, and set serviceAuthenticationManagerType to Impersonate.
- C. Within the service configuration, add a serviceSecurityAudit behavior to the service, and set serviceAuthorizationAuditLevel to SuccessOrFailure.
- D. Within the service configuration, add a serviceCredentials behavior to the service, and set type to Impersonate.

Answer: A

Question: 41

A Windows Communication Foundation (WCF) service is required to log all authorization attempts to the Windows Event Log. You need to configure a behavior and apply it to the service to support this requirement. Which behavior should you configure and apply?

- A. serviceAuthenticationManager
- B. serviceAuthorization
- C. serviceCredentials
- D. serviceSecurityAudit

Answer: D

Question: 42

You have a secured Windows Communication Foundation (WCF) service. You need to track unsuccessful attempts to access the service. What should you do?

- A. Set the serviceAuthorizationManagerType attribute of the serviceAuthorization behavior to Message.
- B. Set the includeExceptionDetailInFaults attribute of the serviceDebug behavior to true.
- C. Set the Mode attribute of the security configuration element to Message.
- D. Set the messageAuthenticationAuditLevel attribute of the serviceSecurityAudit behavior to Failure.

Answer: D

Question: 43

A Windows Communication Foundation (WCF) solution uses the following contract to share a message across its clients. (Line numbers are included for reference only.)

```
01 <ServiceContract(>  
02 Public Interface ITeamMessageService  
03  
04 <OperationContract(>  
05 Function GetMessage() As String  
06  
07 <OperationContract(>  
08 Sub PutMessage(ByVal message As String)  
09 End Interface
```

The code for the service class is as follows.

```
10 Public Class TeamMessageService  
11 Implements ITeamMessageService  
12  
13 Dim key As Guid = Guid.NewGuid()  
14 Dim message As String = "Today_ s Message"  
15  
16 Public Function GetMessage() As String _  
17 Implements ITeamMessageService.GetMessage  
18  
19 Return String.Format("Message:{0}. Key:{1}", message, key)  
20 End Function  
21  
22 Public Sub PutMessage(ByVal message As String) _  
23 Implements ITeamMessageService.PutMessage  
24  
25 Me.message = message  
26 End Sub  
27  
28 End Class
```

The service is self-hosted. The hosting code is as follows.

```
29 Dim host As ServiceHost =  
New ServiceHost(GetType(TeamMessageService))  
30 Dim binding As BasicHttpBinding =  
New BasicHttpBinding(BasicHttpSecurityMode.None)  
31 host.AddServiceEndpoint(  
"MyApplication.ITeamMessageService", binding,  
"http://localhost:12345")  
32 host.Open()
```

You need to ensure that all clients calling GetMessage will retrieve the updated string if the message is updated by any client calling PutMessage. What should you do?

- A. Add the following attribute to the TeamMessageService class, before line 10.
<ServiceBehavior(InstanceContextMode:=InstanceContextMode.Single)>
- B. Add the following attribute to the TeamMessageService class, before line 10002E
<ServiceBehavior(InstanceContextMode:=
InstanceContextMode.PerSession)>
- C. Pass a service instance to the instancing code in line 29, as follows.
Dim host As ServiceHost = New ServiceHost(New TeamMessageService())
Redefine the message string in line 14, as follows.
Shared message As String = "Today_ s Message"
- D. Then change the implementation of PutMessage in lines 22-26 to the following.
Public Sub PutMessage(ByVal message As String) _
Implements ITeamMessageService.PutMessage
TeamMessageService.message = message
End Sub

Answer: A

Question: 44

A Windows Communication Foundation (WCF) solution exposes the following service over a TCP binding. (Line numbers are included for reference only.)

```
01 <ServiceContract(>  
02 <ServiceBehavior(ConcurrencyMode:=ConcurrencyMode.Multiple)>  
03 Public Class DataAccessService  
04  
05 <OperationContract(>  
06 Public Sub PutMessage(ByVal message As String)  
07 MessageDatabase.PutMessage(message)  
08 End Sub  
09  
10 <OperationContract(>  
11 Public Function SearchMessages(ByVal search As String) _  
12 As String()  
13 Return MessageDatabase.SearchMessages(search)  
14 End Function  
15  
16 End Class
```

MessageDatabase supports a limited number of concurrent executions of its methods. You need to change the service to allow up to the maximum number of executions of the methods of MessageDatabase. This should be implemented without preventing customers from connecting to the service. What should you do?

A. Change the service behavior as follows.

```
<ServiceBehavior(ConcurrencyMode:=ConcurrencyMode.Multiple,  
B. InstanceContextMode:=InstanceContextMode.Single)>
```

Change the service behavior as follows.

```
<ServiceBehavior(ConcurrencyMode:=ConcurrencyMode.Single,  
InstanceContextMode:=InstanceContextMode.PerSession)>
```

C. Add a throttling behavior to the service, and configure the maxConcurrentCalls.

D. Add a throttling behavior to the service, and configure the maxConcurrentSessions.

Answer: C

Question: 45

A service implements the following contract. (Line numbers are included for reference only.)

```
01 <ServiceContract(SessionMode:=SessionMode.Required)>  
02 Public Interface IContosoService  
03  
04 <OperationContract(IsOneWay:=True, IsInitiating:=True)>  
05 Sub OperationOne(ByVal value As String)  
06  
07 <OperationContract(IsOneWay:=True, IsInitiating:=False)>  
08 Sub OperationTwo(ByVal value As String)  
09  
10 End Interface
```

The service is implemented as follows.

```
20 Class ContosoService  
21 Implements IContosoService  
22  
23 Public Sub OperationOne(ByVal value As String) _  
24 Implements IContosoService.OperationOne  
25 &  
26 End Sub  
27  
28 Public Sub OperationTwo(ByVal value As String) _  
29 Implements IContosoService.OperationTwo  
30 &  
31 End Sub  
End Class
```

ContosoService uses NetMsmqBinding to listen for messages. The queue was set up to use transactions for adding and removing messages. You need to ensure that OperationOne and OperationTwo execute under the same transaction context when they are invoked in the same session. What should you do?

A. Insert the following attribute to OperationOne on IContosoService.

```
<TransactionFlow(TransactionFlowOption.Mandatory)>
```

Insert the following attribute to OperationTwo on IContosoService.

```
<TransactionFlow(TransactionFlowOption.Mandatory)>
```

B. Insert the following attribute to OperationOne on ContosoService.

```
<OperationBehavior(TransactionScopeRequired:=True,  
TransactionAutoComplete:=False)>
```

Insert the following attribute to OperationTwo on ContosoService.

```
<OperationBehavior(TransactionScopeRequired:=True,  
TransactionAutoComplete:=True)>
```

C. Add the following XML segment to the application config file in the system.serviceModel/bindings configuration section.

```
<netMsmqBinding>
```

```
<binding name="contosoTx" durable="true" receiveContextEnabled="true" />
```

```
</netMsmqBinding>
```

Then use the NetMsmqBinding named contosoTx to listen for messages from the clients.

D. Add the following XML segment to the application config file in the system.serviceModel/bindings configuration section.

```
<customBinding>
```

```
<binding name="contosoTx">
```

```
<transactionFlow />
```

```
<binaryMessageEncoding />
```

```
<msmqTransport durable="true" />
```

```
</binding>
```

```
</customBinding>
```

Then use the CustomBinding named contosoTx to listen for messages from the clients.

Answer: B

Question: 46

A WCF service code is implemented as follows. (Line numbers are included for reference only.)

```
01 <ServiceContract(>
```

```
02 <ServiceBehavior(
```

```
03 InstanceContextMode:=InstanceContextMode.Single)>
```

```
04 Public Class CalculatorService
```

```
05
```

```
06 <OperationContract(>
```

```
07 Public Function Calculate(ByVal op1 As Double,
```

```
08 ByVal op As String, ByVal op2 As Double) As Double
```

```
&
```

```
24 End Function
```

```
25
```

```
26 End Class
```

You need to decrease the response time of the service. What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two.)

A. Change the service behavior to the following.

```
<ServiceBehavior(
```

B. InstanceContextMode:=InstanceContextMode.Single,
ConcurrencyMode:=ConcurrencyMode.Multiple)>

C. Change the service behavior to the following.

```
<ServiceBehavior(InstanceContextMode:=InstanceContextMode.PerCall)>
```

D. Require the clients use threads, the Parallel Task Library, or other mechanism to issue service calls in parallel.

Require the clients to use async operations when calling the service.

Answer: A B

Question: 47

You are creating a Windows Communication Foundation (WCF) service application. The application needs to service many clients and requests simultaneously. The application also needs to ensure subsequent individual client requests provide a stateful conversation. You need to configure the service to support these requirements. Which attribute should you add to the class that is implementing the service?

A. <ServiceBehavior(InstanceContextMode:=InstanceContextMode.PerSession,
ConcurrencyMode:=ConcurrencyMode.Single)>

B. <ServiceBehavior(InstanceContextMode:=InstanceContextMode.PerCall,
ConcurrencyMode:=ConcurrencyMode.Reentrant)>

C. <ServiceBehavior(InstanceContextMode:=InstanceContextMode.PerSession,
ConcurrencyMode:=ConcurrencyMode.Multiple)>

D. <ServiceBehavior(InstanceContextMode:=InstanceContextMode.PerCall,
ConcurrencyMode:=ConcurrencyMode.Multiple)>

Answer: C

Question: 48

A Windows Communication Foundation (WCF) service only accepts messages that are signed and encrypted. A client application is not receiving expected responses from the service. You need to enable logging to verify that the messages from the client are signed and encrypted. You also need to see what each message looks like before the message body is deserialized into a .NET object. What should you do?

A. Configure the System.ServiceModel trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>
```

```
<messageLogging
```

```
logEntireMessage="true"
```

```
logMessagesAtServiceLevel="true"
```

```
logMessagesAtTransportLevel="true" />
```

```
</diagnostics>
```

B. Configure the System.ServiceModel trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>  
<messageLogging  
logEntireMessage="true"  
logMessagesAtServiceLevel="true" />  
</diagnostics>
```

C. Configure the System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>  
<messageLogging  
logEntireMessage="true"  
logMessagesAtServiceLevel="true"  
logMessagesAtTransportLevel="true" />  
</diagnostics>
```

D. Configure the System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section. In the system.serviceModel configuration, add the following XML segment.

```
<diagnostics>  
<messageLogging  
logMessagesAtServiceLevel="true"  
logMessagesAtTransportLevel="true" />  
</diagnostics>
```

Answer: C

Question: 49

Your Windows Communication Foundation (WCF) client application uses HTTP to communicate with the service. You need to enable message logging and include all security information such as tokens and nonces in logged messages. What should you do?

A. In the application configuration file, add the logKnownPii attribute to the message logging diagnostics source and set the value of the attribute to true.

Generate the ContosoService class using the Add Service Reference wizard.

Add a reference to System.ServiceModel.Routing.dll.

Add the following code segment.

```
Dim client As ContosoService = New ContosoService()  
Dim behavior As SoapProcessingBehavior = New SoapProcessingBehavior()  
behavior.ProcessMessages = True  
client.Endpoint.Behaviors.Add(behavior)
```

B. In the application configuration file, add the following XML segment to the system.serviceModel configuration section group.

```
<diagnostics>  
<messageLogging logMessagesAtTransportLevel="true"  
logEntireMessage="true" />  
</diagnostics>
```

C. In the machine configuration file, add the following XML segment to the system.serviceModel configuration section.

```
<machineSettings enableLoggingKnownPii="true" />
```

Generate the ContosoService class using the Add Service Reference wizard.

Add the following code segment.

```
Dim client As ContosoService = New ContosoService()
```

```
client.Endpoint.Behaviors.Add(New CallbackDebugBehavior(True))
```

In the machine configuration file, add the following XML segment to the system.serviceModel configuration section.

```
<machineSettings enableLoggingKnownPii="true" />
```

D. In the application configuration file, add the logKnownPii attribute to the message logging diagnostics source and set the value of the attribute to true.

In the application configuration file, add the following XML segment to the system.serviceModel configuration section group.

```
<diagnostics>
```

```
<messageLogging logMessagesAtTransportLevel="true"/>
```

```
</diagnostics>
```

Answer: D

Question: 50

A Windows Communication Foundation (WCF) service has the following contract.

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

```
Public Interface IShipping
```

```
<OperationContract(>
```

```
Function DoWork(ByVal id As Integer) As String
```

```
End Interface
```

This is one of several service contracts hosted by your application. All endpoints use SOAP 1.2 bindings with WS-Addressing 1.0. The System.ServiceModel.MessageLogging trace source in the system.diagnostics configuration section is configured with one listener.

You need to make sure that only the messages that are returned from the DoWork operation are logged.

Which XML segment should you add to the system.serviceModel/diagnostics/messageLogging/filters configuration element?

A. <add xmlns:addr="http://www.w3.org/2005/08/addressing">

```
//addr:Action[text() =
```

```
'http://contoso.com/IShipping/DoWorkResponse']
```

```
</add>
```

B. <add xmlns:soap="http://www.w3.org/2003/05/soap-envelope">

```
//soap:Action[text() =
```

```
'http://contoso.com/IShipping/DoWorkResponse']
```

```
</add>
```

C. <add xmlns:addr="http://www.w3.org/2005/08/addressing">

```
//addr:Action[text() =
```

```
'http://contoso.com/IShipping/DoWork']
```

```
</add>
```

```
D. <add xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
//soap:Action[text() =
'http://contoso.com/IShipping/DoWork']
</add>
```

Answer: A

Question: 51

You are moving a Windows Communication Foundation (WCF) service into production. You need to be able to monitor the health of the service. You only want to enable all performance counter instances exposed by the ServiceModelService 4.0.0.0 counter group. Which element should you add to the system.serviceModel section in the application configuration file?

- A. <diagnostics performanceCounters="ServiceOnly" />
- B. <diagnostics wmiProviderEnabled="true" performanceCounters="Off" />
- C. <diagnostics performanceCounters="All" />
- D. <diagnostics wmiProviderEnabled="true" />

Answer: A

Question: 52

You are using tracing to diagnose run-time issues. When you look at the traces for the service in SvcTraceViewer.exe, you see what is shown in the exhibit. (Click the Exhibit button.) The exception trace is selected in SvcTraceViewer.exe. You need to interpret the trace results to determine where the error occurred and what to do next.

What should you do?

- A. This issue occurred in the ServiceHost during ServiceHost.Open(). Enable WMI by adding the following configuration to the system.serviceModel configuration section in the application configuration file. <diagnostics wmiProviderEnabled="true" /> Restart the application and inspect the endpoints visible through WMI.
- B. This issue occurred in the ServiceHost during ServiceHost.Open(). Compare the security settings for any endpoints that use an MSMQ transport to the security configuration of the MSMQ queue used by the endpoint.
- C. This issue occurred at the ServiceHost when receiving a message. Compare the security configurations on the client and server to make sure that they are compatible.
- D. This issue occurred at the ServiceHost when accepting an initial set of messages from MSMQ. Log all messages sent between the client and the server.

Answer: B

Question: 53

You are implementing a Windows Communication Foundation (WCF) service contract named `IContosoService` in a class named `ContosoService`. The service occasionally fails due to an exception being thrown at the service. You need to send the stack trace of any unhandled exceptions to clients as a fault message. What should you do?

A. In the application configuration file on the client, add the following XML segment to the `system.serviceModel/behaviors` configuration section group.

```
<endpointBehaviors>
<behavior name=" debug ">
< callback Debug includeExceptionDetailInFaults="true" />
</behavior>
</endpointBehaviors>
```

B. Associate the debug behavior with any endpoints that need to return exception details.

C. In the application configuration file on the service and all the clients, add the following XML segment to the `system.diagnostics/sources` configuration section group.

```
<source name="System.ServiceModel" switchValue=" Error " propagateActivity="true">
<listeners>
<add name="ServiceModelTraceListener"
initializeData="app_tracelog.svclog"
type="System.Diagnostics.XmlWriterTraceListener" />
</listeners>
</source>
```

D. Apply the following attribute to the `ContosoService` class.

```
<ServiceBehavior(IncludeExceptionDetailInFaults:=True)>
```

For each `OperationContract` exposed by `IContosoService`, apply the following attribute.

```
<FaultContract(GetType(Exception))>
```

Answer: C

Question: 54

A Windows Communication Foundation (WCF) application exposes a service as a SOAP endpoint for consumption by cross-platform clients. During integration testing, you find that one of the clients is not generating the correct messages to the WCF application. In order to debug the issue and fix the communication, you need to configure the service to log messages received from the client. What should you do?

A. Set an `etwTracking` behavior on the service and configure a listener for the `System.ServiceModel` trace source.

B. Set an `etwTracking` behavior on the service and configure a listener for the `System.ServiceModel.MessageLogging` trace source.

C. Enable `messageLogging` in the `System.ServiceModel` diagnostics element configuration and configure a listener for the `System.ServiceModel.MessageLogging` trace source.

D. Enable `messageLogging` in the `System.ServiceModel` diagnostics element configuration and configure a listener for the `System.ServiceModel` trace source.

Answer: C

Question: 55

A Windows Communication Foundation (WCF) service interacts with the database of a workflow engine. Data access authorization is managed by the database, which raises security exceptions if a user is unauthorized to access it. You need to ensure that the application transmits the exceptions raised by the database to the client that is calling the service. Which behavior should you configure and apply to the service?

- A. routing
- B. serviceDebug
- C. serviceSecurityAudit
- D. workflowUnhandledException

Answer: B

Question: 56

You develop a Windows Communication Foundation (WCF) service to generate reports. Client applications call the service to initiate report generation but do not wait for the reports to be generated. The service does not provide any status to the client applications. The service class is defined as follows. (Line numbers are included for reference only.)

```
01<ServiceContract(>  
02Public Class ReportGeneratorService  
03  
04Private Function GenerateReports(  
ByVal clientID As Integer) As Integer  
05&  
06Return 0  
07End Function  
08  
09End Class
```

You need to ensure that client applications can initiate reports without waiting for status. Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

- A. Insert the following code at line 03.
<OperationContract(IsOneWay:=True)>
- B. Insert the following code at line 03.
<OperationContract(AsyncPattern:=False)>
- C. At line 04, change the GenerateReports method from Private to Public.
- D. Remove line 06. At line 04, change the type of GenerateReports method to Sub. Remove the code `_ As Integer_` from the method definition.

Answer: A D

Question: 57

Your company has a Windows Communication Foundation (WCF) service at the URL `http://services.contoso.com/OrderLookupService.svc`.

The `<system.serviceModel>` section of the configuration file is as follows. (Line numbers are included for reference only.)

```
01<system.serviceModel>
02<behaviors>
03<serviceBehaviors>
04<behavior>
05 <serviceDebug
includeExceptionDetailInFaults="false"/>
06
07</behavior>
08</serviceBehaviors>
09</behaviors>
10 <serviceHostingEnvironment
multipleSiteBindingsEnabled="true" />
11 </system.serviceModel>
```

You need to ensure that the service publishes the WSDL description at `http://services.contoso.com/OrderLookupService.svc?wsdl`. What should you do?

A. Change the `serviceDebug` element at line 05 as follows.

```
<serviceDebug includeExceptionDetailInFaults="true"/>
```

B. Insert the following element at line 06.

```
<serviceDiscovery>
<announcementEndpoints>
<endpoint name="wsdlAnnouncement"
kind="udpAnnouncementEndpoint" />
</announcementEndpoints>
</serviceDiscovery>
```

C. Insert the following element at line 06.

```
<serviceMetadata httpGetEnabled="true" />
```

D. Insert the following element at line 06.

```
<serviceMetadata httpGetEnabled="false" />
```

Answer: C

Question: 58

You are developing a Windows Communication Foundation (WCF) service. The service operation takes a customer number as the only argument and returns information about the customer. The service requires a security token in the header of the message. You need to create a message contract for the service. Which code segment should you use?

A. <ServiceContract(>
Public Interface IService
<OperationContract(>
Function GetCustomerInformation(
ByVal header As Header,
ByVal customerNumber As Integer) As CustomerInformation
End Interface
<DataContract(>
Public Class CustomerInformation
&
End Class
<MessageContract(>
Public Class Header
<MessageHeader(>
Public SecurityTag As String
End Class

B. <ServiceContract(>
Public Interface IService
<OperationContract(>
Function GetCustomerInformation(
ByVal header As Header,
ByVal customerNumber As Integer) As CustomerInformation
End Interface
<MessageContract(>
Public Class CustomerInformation
&
End Class
<MessageContract(>
Public Class Header
<MessageHeader(>
Public SecurityTag As String
End Class

C. <ServiceContract(>
Public Interface IService
<OperationContract(>
Function GetCustomerInformation(
ByVal request As CustomerNumber) As CustomerInformation
End Interface
<DataContract(>

```

Public Class CustomerInformation
&
End Class
<MessageContract(>
Public Class CustomerNumber
<MessageHeader(>
Public SecurityTag As String
<MessageBodyMember(>
Public CustomerNumberElement As Integer
End Class
D. <ServiceContract(>
Public Interface IService
<OperationContract(>
Function GetCustomerInformation(
ByVal request As CustomerNumber) As CustomerInformation
End Interface
<MessageContract(>
Public Class CustomerInformation
&
End Class
<MessageContract(>
Public Class CustomerNumber
<MessageHeader(>
Public SecurityTag As String
<MessageBodyMember(>
Public CustomerNumberElement As Integer
End Class

```

Answer: D

Question: 59

You are developing a Windows Communication Foundation (WCF) service that contains the following operation contract.

```

<OperationContract(>
Function GetCustomerNames() As CustomerNames

```

The operation returns customer names.

You need to develop a definition for the operation contract that produces XML with the following structure.

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Header />
<s:Body>
<Names xmlns=http://tempuri.org/
xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
<a:string>Customer1</a:string>

```

```
<a:string>Customer2</a:string>
<a:string>Customer3</a:string>
</Names>
</s:Body>
</s:Envelope>
```

Which code segment should you use?

- A. <MessageContract(IsWrapped:=False)>
Public Class CustomerNames
<MessageBodyMember(>
Public Names() As String
End Class
- B. <MessageContract(WrapperName:="")>
Public Class CustomerNames
<MessageBodyMember(>
Public Names() As String
End Class
- C. <DataContract(>
Public Class CustomerNames
<DataMember(>
Public Names() As String
End Class
- D. <DataContract(>
Public Class CustomerNames
<DataMember(IsRequired:=False)>
Public Names() As String
End Class

Answer: A

Question: 60

A Windows Communication Foundation (WCF) service handles online order processing for your company.

You discover that many requests are being made with invalid account numbers. You create a class named AccountNumberValidator that has a method named Validate. Before the message is processed, you need to validate account numbers with AccountNumberValidator and reject messages with invalid account numbers. You create a new class that implements the IParameterInspector interface. Which code segment should you use in this class?

- A. Public Sub AfterCall(
ByVal operationName As String,
ByVal outputs() As Object,
ByVal returnValue As Object,
ByVal correlationState As Object) _
Implements IParameterInspector.AfterCall

```

Dim accountNumber As String = GetAccountNumber(outputs)
Dim validator As AccountNumberValidator =
New AccountNumberValidator()
If (Not validator.Validate(accountNumber)) Then
Throw New FaultException()
End If
End Sub
Public Function BeforeCall(
ByVal operationName As String,
ByVal inputs() As Object) As Object _
Implements IParameterInspector.BeforeCall
Return Nothing
End Function
B. Public Sub AfterCall(
ByVal operationName As String,
ByVal outputs() As Object,
ByVal returnValue As Object,
ByVal correlationState As Object) _
Implements IParameterInspector.AfterCall
Return
End Sub
Public Function BeforeCall(
ByVal operationName As String,
ByVal inputs() As Object) As Object _
Implements IParameterInspector.BeforeCall
Dim accountNumber As String = GetAccountNumber(inputs)
Dim validator As AccountNumberValidator =
New AccountNumberValidator()
If (Not validator.Validate(accountNumber)) Then
Throw New FaultException()
End If
Return Nothing
End Function
C. Public Sub AfterCall(
ByVal operationName As String,
ByVal outputs() As Object,
ByVal returnValue As Object,
ByVal correlationState As Object) _
Implements IParameterInspector.AfterCall
Dim accountNumber As String = GetAccountNumber(outputs)
Dim validator As AccountNumberValidator =
New AccountNumberValidator()
If (Not validator.Validate(accountNumber)) Then
returnValue = New FaultException()
End If
End Sub
Public Function BeforeCall(
ByVal operationName As String,

```

```
ByVal inputs() As Object) As Object _
Implements IParameterInspector.BeforeCall
Return Nothing
End Function
D. Public Sub AfterCall(
ByVal operationName As String,
ByVal outputs() As Object,
ByVal returnValue As Object,
ByVal correlationState As Object) _
Implements IParameterInspector.AfterCall
Return
End Sub
Public Function BeforeCall(
ByVal operationName As String,
ByVal inputs() As Object) As Object _
Implements IParameterInspector.BeforeCall
Dim accountNumber As String = GetAccountNumber(inputs)
Dim validator As AccountNumberValidator =
New AccountNumberValidator()
If (Not validator.Validate(accountNumber)) Then
Return New FaultException()
End If
End Function
```

Answer: B

Question: 61

You are creating a client application and configuring it to call a Windows Communication Foundation (WCF) service. When the application is deployed, it will be configured to send all messages to a WCF routing service. You need to ensure that the application can consume the target service after the application is deployed. What should you do?

- A. In the client application, add a service reference to the router service. In the client binding configuration, specify the address of the router service.
- B. In the client application, add a service reference to the target service. In the client binding configuration, specify the address of the target service.
- C. In the client application, add a service reference to the router service. In the client binding configuration, specify the address of the target service.
- D. In the client application, add a service reference to the target service. In the client binding configuration, specify the address of the router service.

Answer: D

Question: 62

You create a service and deploy it on a network in a building named Building1. You will deploy the service to Building2. The service in Building1 is configured using the following discovery scopes.

```
<scopes>
<add
scope="http://contoso.com/Chicago/Building1"/>
<add
scope="ldap:///ou=Building1,ou=Chicago,o=contoso,c=us"/>
</scopes>
```

The service in Building2 will be configured using the following discovery scopes.

```
<scopes>
< add
scope="http://contoso.com/Chicago/Building2"/>
< add
scope="ldap:///ou=Building2,ou=Chicago,o=contoso,c=us"/>
</scopes>
```

You need to ensure that the client application can discover the service in Building1 or the service in Building2. Which scopes should you add to the client configuration file?

- A. <scopes>
<add scope="http://contoso.com/Chicago/*"/>
</scopes>
- B. <scopes>
<add scope="http://contoso.com/Chicago"/>
</scopes>
- C. <scopes>
< add
scope="ldap:///ou=Building,ou=Chicago,o=contoso,c=us"/>
</scopes>
- D. <scopes>
< add
scope="ldap:///ou=*,o=contoso,c=us"/>
</scopes>

Answer: B

Question: 63

You are modifying a Windows Communication Foundation (WCF) service that issues security tokens. The service is accessible through the named pipe protocol. No endpoints are added in the service code. The configuration file for the service is as follows. (Line numbers are included for reference only.)

```
01<configuration>
02<system.serviceModel>
03<services>
04<service name="Contoso.TokenService">
05
06<host>
07<baseAddresses>
08
09<add
baseAddress="net.pipe://www.contoso.com/tokenpipe" />
10</baseAddresses>
11</host>
12</service>
13</services>
14</system.serviceModel>
15 </configuration>
```

You need to ensure that new and existing client applications can access the service through HTTP and named pipes. What should you do?

- A. Insert the following line at line 05.
<endpoint address=http://www.contoso.com
binding="wsHttpBinding"
contract="Contoso.TokenService" />
- B. Insert the following line at line 05.
<endpoint address="http://www.contoso.com"
binding="basicHttpBinding"
contract="Contoso.TokenService" />
- C. Insert the following line at line 08.
<add baseAddress="http://www.contoso.com" />
- D. Insert the following line at line 08.
<add baseAddress="net.tcp://www.contoso.com:8090" />

Answer: C

Question: 64

Your company has an existing Windows Communication Foundation (WCF) service that allows business partners to place orders. The service uses netMsmqBinding. You find that processing every order in its own transaction is causing a delay. You need to ensure that the service is configured to process multiple orders in one transaction. What should you do?

- A. Use <serviceThrottling> service behavior and set the maxConcurrentCalls attribute.
- B. Use <transactedBatching> endpoint behavior and set the batchSize attribute.
- C. Use <dispatcherSynchronizationBehavior> endpoint behavior and set the maxPendingReceives attribute.
- D. Use <synchronousReceive> endpoint behavior.

Answer: B

Question: 65

You are developing a Windows Communication Foundation (WCF) service that contains the following code segment.

```
<ServiceContract(>  
Public Interface ICustomerService  
&  
End Interface  
Public Class CustomerService  
Implements ICustomerService  
&  
End Class
```

The service is self-hosted in a console application. Older client applications access the service at `http://contoso.com:8080/CustomerService/V1`. Newer client applications access the service at `http://contoso.com:8080/CustomerService/V2`. You need to ensure that any client application can access the service at either address. Which code segment should you use?

- A.

```
Dim serviceAddress1 As Uri =  
New Uri("http://contoso.com:8080/CustomerService/V1")  
Dim serviceAddress2 As Uri =  
New Uri("http://contoso.com:8080/CustomerService/V2")  
Dim host As ServiceHost =  
New Service Host(GetType(ICustomerService),  
New Uri() {serviceAddress1, serviceAddress2})
```
- B.

```
Dim serviceAddress1 As Uri =  
New Uri("http://contoso.com:8080/CustomerService/V1")  
Dim serviceAddress2 As Uri =  
New Uri("http://contoso.com:8080/CustomerService/V2")
```



```

Dim host As ServiceHost =
New ServiceHost(GetType(CustomerService),
New Uri() {serviceAddress1, serviceAddress2})
C. Dim serviceAddress As Uri =
New Uri("http://contoso.com:8080/")
Dim host As ServiceHost =
New ServiceHost(GetType(CustomerService),
New Uri() {serviceAddress})
host.AddServiceEndpoint(GetType(ICustomerService),
New BasicHttpBinding(), "CustomerService/V1")
host.AddServiceEndpoint(GetType(ICustomerService),
New BasicHttpBinding(), "CustomerService/V2")
D. Dim serviceAddress As Uri =
New Uri("http://contoso.com:8080/")
Dim host As ServiceHost =
New ServiceHost(GetType(ICustomerService),
New Uri() {serviceAddress})
host.AddServiceEndpoint(GetType(CustomerService),
New BasicHttpBinding(), "CustomerService/V1")
host.AddServiceEndpoint(GetType(CustomerService),
New BasicHttpBinding(), "CustomerService/V2")

```

Answer: C

Question: 66

You are hosting a Windows Communication Foundation (WCF) service under Microsoft Internet Information Services (IIS) 7.0.

You have set up a Web site in IIS Manager. The physical path is C:\wwwroot\Calendar. There is a Calendar.svc file in the C:\wwwroot\Calendar folder. It contains the following directive.

```
<%@ ServiceHost Language="VB" Debug="true" Service="Calendar.Calendar"
CodeBehind="Calendar.svc.vb" %>
```

The Calendar.svc.vb file contains the source for the Calendar class in the Calendar namespace. You compile this code into the Calendar.dll file. You need to deploy your service to the Web site.

What should you do?

- A. Copy the Calendar.dll file to the C:\wwwroot\Calendar\code folder.
- B. Copy the Calendar.dll file to the C:\wwwroot\Calendar\bin folder.
- C. Copy the Calendar.svc.vb file to the C:\wwwroot\Calendar\bin folder.
- D. Copy the Calendar.svc.vb file to the C:\wwwroot\Calendar\code folder.

Answer: B

Question: 67

You are developing a client application that uses the following code to consume a Windows Communication Foundation (WCF) service. (Line numbers are included for reference only.)

```
01 Dim myBinding As BasicHttpBinding =  
New BasicHttpBinding()  
02 Dim myEndpointAddress As EndpointAddress =  
New EndpointAddress(  
"http://contoso.com/TaxService.svc")  
03  
04 Dim client As ITaxService =  
channelFactory.CreateChannel()  
05 Dim data As String = client.GetData(1)
```

You need to consume the service.

Which code segment should you insert at line 03?

- A. Dim channelFactory =
New ChannelFactory(Of ITaxService)()
- B. Dim channelFactory =
New ChannelFactory(Of ITaxService)(myBinding)
Dim channelFactory =
- C. New ChannelFactory(Of ITaxService)
(myBinding, myEndpointAddress)
channelFactory =
- D. New ChannelFactory(Of ITaxService)
("http://contoso.com/TaxService.svc")

Answer: C

Question: 68

You need to modify a client application that consumes a Windows Communication Foundation (WCF) service. The service metadata is no longer available. You need to modify the previously generated proxy to include asynchronous calls to the service. What should you do?

- A. Update the service reference with the Generate asynchronous operations option.
- B. Create a partial class for the previously generated proxy and include the new asynchronous methods.
- C. Create a class with the same name as the previously generated proxy and add the new asynchronous methods. Add the new class to a namespace that is different from the original proxy.
- D. Create a class with the same name as the previously generated proxy and add the new asynchronous methods as partial methods. Add the new class to a namespace that is different from the original proxy.

Answer: B

Question: 69

You develop a Windows Communication Foundation (WCF) service that interacts with Microsoft Message Queuing (MSMQ). The service requires sessions. You need to create a custom binding that enables messages sent to the queue to be viewed when you are using a listener tool. Which binding elements should you use?

- A. textMessageEncoding and msmqTransport in this order
- B. textMessageEncoding and msmqIntegrationTransport in this order
- C. msmqTransport and textMessageEncoding in this order
- D. msmqIntegrationTransport and textMessageEncoding in this order

Answer: A

Question: 70

You are developing a client application that consumes a Windows Communication Foundation (WCF) service. You use the svcutil.exe utility to create a proxy for the service. You use the svcutil.exe switches that generate asynchronous calls. GetFlight is a service operation that takes no parameters and returns a string. The GetFlightCallback method must be called when the service operation returns.

You create an instance of the client proxy with the following code.

```
Dim client As TravelServiceClient = New TravelServiceClient()
```

You need to ensure that a callback is received when the GetFlight operation is called asynchronously. Which code segment should you use?

- A.

```
client.BeginGetFlight(AddressOf GetFlightCallback, Nothing)
client.GetFlight()
client.GetFlight()
client.BeginGetFlight(AddressOf GetFlightCallback, Nothing)
```
- B.

```
AddHandler client.GetFlightCompleted,
New EventHandler(Of GetFlightCompletedEventArgs)
(AddressOf GetFlightCallback)
client.GetFlightAsync()
```
- C.

```
Dim asyncResult As IAsyncResult = client.BeginGetFlight(
D. 

```
AddressOf GetFlightCallback, client)
client.EndGetFlight(asyncResult)
```


```

Answer: C

Question: 71

You are creating a Windows Communication Foundation (WCF) service. The service endpoints change frequently. On the service, you add a new `ServiceDiscoveryBehavior` to the `Behaviors` collection of the `ServiceHost` `Description` property. You need to ensure that client applications can communicate with the service and discover changes to the service endpoints. What should you do?

- A. Add a new `ServiceDiscoveryBehavior` to the `Behaviors` collection in the client application.
- B. Add a new `AnnouncementClient` to the `Behaviors` collection in the client application.
- C. Use the `FindCriteria` class and the `UdpDiscoveryEndpoint` class to set up the binding in the client application.
- D. Use the `DiscoveryProxy` class and the `EndpointDiscoveryMetadata` class to set up the binding in the client application.

Answer: C

Question: 72

You are hosting a Windows Communication Foundation (WCF) service at `http://www.contoso.com` for a law enforcement agency. The agency adds operations to support sending biometric fingerprint data via non-buffered streaming. The service data is not routed between intermediaries. The WCF binding you are using by default does not support encryption. You need to ensure that fingerprint data is not disclosed when it is passed over the network. What should you do?

- A. Use `basicHttpBinding` with message security to `https://www.contoso.com`.
- B. Use `basicHttpBinding` over transport security at `https://www.contoso.com`.
- C. Use `wsHttpBinding` over message security at `https://www.contoso.com`.
- D. Use `wsHttpBinding` over transport security at `http://www.contoso.com`.

Answer: B

Question: 73

You are maintaining a Windows Communication Foundation (WCF) service that uses a custom `UserNamePassword` class to authenticate clients. The service certificate is hosted in the deployment server store for trusted root certificate authorities and has a `Subject` value of `TaxServiceKey`. Other service certificates hosted on the server also use `TaxServiceKey` as a `Subject` value. You need to ensure that the service identifies itself with a certificate whose subject name and distinguished names are `TaxServiceKey`. Which code segment should you use?

- A. HostInstance.Credentials.ServiceCertificate.SetCertificate (StoreLocation.LocalMachine, StoreName.My, X509FindType.FindBySubjectName, "CN=TaxServiceKey")
- B. HostInstance.Credentials.ServiceCertificate.SetCertificate (StoreLocation.LocalMachine, StoreName.AuthRoot, X509FindType.FindBySubjectName, "CN=TaxServiceKey")
- C. HostInstance.Credentials.ServiceCertificate.SetCertificate (StoreLocation.LocalMachine, StoreName.My, X509FindType.FindBySubjectDistinguishedName, "CN=TaxServiceKey")
- D. HostInstance.Credentials.ServiceCertificate.SetCertificate (StoreLocation.LocalMachine, StoreName.Root, X509FindType.FindBySubjectDistinguishedName, "CN=TaxServiceKey")

Answer: D

Question: 74

You are developing a Windows Communication Foundation (WCF) service that returns location information for authorized law enforcement agencies. The service contract is as follows.

```
<ServiceContract(>
Public Interface IMappingService
<OperationContract(>
Function GetLocationCoordinates(
ByVal cityName As String) As Long()
<OperationContract(>
Function GetLocationOfCitizen(
ByVal ssn As String) As Long()
End Interface
```

Users are authenticated and impersonated. The system uses ASP.NET roles. The members of law enforcement are members of the LawEnforcement role. You need to ensure that only members of the LawEnforcement role can call these methods. What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two.)

- A. Add a PrincipalPermissionAttribute to each method that should be available only to members of law enforcement. Set its SecurityAction to Demand and set the role equal to LawEnforcement.
- B. Use the CurrentPrincipal property of the thread. Call the IsInRole method specifying LawEnforcement as a parameter.
- C. Create a GenericPrincipal specifying Thread.CurrentPrincipal.Identity as the IIdentityParameter and LawEnforcement as the only value for the Roles parameter.
- D. At the beginning of each method, enumerate each ClaimSet in a new WindowsClaimSet. Use the FindClaims method to locate a claim type named Role with a right named LawEnforcement.

Answer: A B

Question: 75

You develop a Windows Communication Foundation (WCF) service that employees use to access bonus information. You define the following service contract. (Line numbers are included for reference only.)

```
01<ServiceContract(SessionMode:=SessionMode.Required)>
02Public Interface IFinancialService
03
04<OperationContract(>
05 Function Login(
06     ByVal employeeID As Integer,
07     ByVal passwordHash As String) As String
08Function GetBonus(ByVal month As Integer) As Double
09
10<OperationContract(IsTerminating:=True)>
11Sub Logout()
12
13 End Interface
```

Client applications can invoke methods without logging in. You need to ensure that the client applications invoke Login before invoking any other method. You also need to ensure that client applications cannot consume the service after invoking Logout. Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

- A. Replace line 04 with the following code.
<OperationContract(IsInitiating:=False)>
- B. Replace line 04 with the following code.
<OperationContract(IsInitiating:=True, IsTerminating:=True)>
- C. Replace line 07 with the following code.
<OperationContract(IsInitiating:=False)>
- D. Replace line 10 with the following code.
<OperationContract(IsInitiating:=False, IsTerminating:=True)>

Answer: C D

Question: 76

You are developing a Windows Communication Foundation (WCF) service that allows customers to update financial data. The service contract is defined as follows. (Line numbers are included for reference only.)

```
01<ServiceContract(>  
02Public Interface IDataUpdate  
03  
04<OperationContract(>  
05<TransactionFlow(TransactionFlowOption.Mandatory)>  
06 Sub Update(  
ByVal accountNumber As String,  
ByVal amount As Double)  
07  
08End Interface  
09  
10Class UpdateService  
11Implements IDataUpdate  
12  
13 <OperationBehavior(  
TransactionScopeRequired:=True,  
TransactionAutoComplete:=False)>  
14 Public Sub Update(  
ByVal accountNumber As String,  
ByVal amount As Double)  
Implements IDataUpdate.Update  
15&  
16End Sub  
17  
18 End Class
```

You need to ensure that the service is invoked within a transaction. What should you do?

A. Replace line 01 with the following code.

```
<ServiceContract(  
SessionMode:=SessionMode.NotAllowed)>
```

B. Replace line 01 with the following code.

```
<ServiceContract(  
SessionMode:=SessionMode.Required)>
```

C. Insert the following code at line 09.

```
<ServiceBehavior(  
TransactionAutoCompleteOnSessionClose:=False)>
```

D. Insert the following code at line 09.

```
<ServiceBehavior(  
ReleaseServiceInstanceOnTransactionComplete:=False)>
```

Answer: B

Question: 77

You are developing a Windows Communication Foundation (WCF) service that is hosted by a Windows Forms application. The ServiceHost instance is created in the Form constructor. You need to ensure that the service is not blocked while the UI thread is busy. What should you do?

- A. Decorate the service implementation class with the following line of code.
<ServiceBehavior(UseSynchronizationContext:=False)>
- B. Decorate the service implementation class with the following line of code.
<ServiceBehavior(ConcurrencyMode:=ConcurrencyMode.Multiple)>
- C. Call the Invoke method of the form and supply a delegate.
- D. Call the BeginInvoke method of the form and supply a delegate.

Answer: A

Question: 78

You are developing a Windows Communication Foundation (WCF) service. You must record all available information for the first 1,000 messages processed, even if they are malformed. You need to configure the message logging section of the configuration file. Which configuration segment should you use?

- A. <messageLogging logEntireMessage="true" logMalformedMessages="true" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="true" maxMessagesToLog="1000"/>
- B. <messageLogging logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="true" maxMessagesToLog="1000"/>
- C. <messageLogging logEntireMessage="false" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="false" maxMessagesToLog="1000"/>
- D. <messageLogging logMalformedMessages="true" logMessagesAtServiceLevel="true" logMessagesAtTransportLevel="false" maxMessagesToLog="1000"/>

Answer: A

Question: 79

You are developing a Windows Communication Foundation (WCF) service. The service configuration file has a <System.Diagnostics> element defined. You need to ensure that all security audit information, trace logging, and message logging failures are recorded. Which configuration segment should you add to the <System.Diagnostics> element?

A. <sources>

```
<source name="System.ServiceModel"
switchValue="Information, ActivityTracing"
propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
```

B. </source>

```
<source name="System.ServiceModel.MessageLogging"
propagateActivity="true">
<listeners>
<add name="text" />
</listeners>
</source>
</sources>
```

C. <sources>

```
<source name="System.ServiceModel"
switchValue="Information, ActivityTracing"
propagateActivity="true" />
<source name="System.ServiceModel.MessageLogging"
propagateActivity="true" />
</sources>
```

D. <sources>

```
<source name="System.ServiceModel"
switchValue="Information, ActivityTracing"
propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
```

E. </source>

```
<source name="System.ServiceModel.MessageLogging"
propagateActivity="true">
<listeners>
<add name="xml" />
</listeners>
</source>
</sources>
<sharedListeners>
```

```
<add name="xml"
type="System.Diagnostics.XmlWriterTraceListener"
initializeData="& " />
</sharedListeners>
F. <sources>
<source name="System.ServiceModel"
switchValue="Information, ActivityTracing"
propagateActivity="true" />
<source name="System.ServiceModel.MessageLogging"
propagateActivity="true" />
</sources>
<sharedListeners>
<add name="xml"
type="System.Diagnostics.XmlWriterTraceListener"
initializeData="& " />
</sharedListeners>
```

Answer: C

Question: 80

You develop a Windows Communication Foundation (WCF) service. You enable all performance counters and run multiple calls to the service. The service must isolate session data for each user. You need to monitor the instancing behavior used in the service. Which performance counter should you monitor?

- A. ServiceModelService 4.0.0.0\Calls
- B. ServiceModelService 4.0.0.0\Instances
- C. ASP.NET State Service\State Server Sessions Active
- D. ASP.NET State Service\State Server Sessions Total

Answer: B

Question: 81

You create a Windows Communication Foundation (WCF) service and deploy it with wsHttpBinding and message security enabled. You create an intermediate WCF service for logging messages sent to the primary service. The intermediate service is called via the clientVia endpoint behavior. The primary service is receiving malformed data from a client application. You need to enable inspection of the malformed data and prevent message tampering. What should you do?

- A. Specify a protection level of None in the service contract for the intermediate service. Disable message and transport security from the client application configuration file.
- B. Specify a protection level of Sign in the service contract for the intermediate service. Disable transport security from the client application configuration file.
- C. Modify the binding on the intermediate service to use netNamedPipeBinding.
- D. Modify the binding on the intermediate service to use webHttpBinding.

Answer: B
