Developing and Implementing Web Applications with Microsoft Visual Basic .NET and Microsoft Visual Studio .NET

70-305

This study guide is arranged according to Microsoft exam objectives. The whole study guide is divided into seven parts as shown in the table.

Part Name	Questions
Part 1 Creating User Services	10
Part 2 Creating and Managing Components and .NET Assemblies	5
Part 3 Consuming and Manipulating Data	10
Part 4 Testing and Debugging	10
Part 5 Deploying a Web Application	5
Part 6 Maintaining and Supporting a Web Application	3
Part 7 Configuring and Securing a Web Application	10

Part 1 Creating User Services

Question 1.
You create an ASP.NET page named ITCertKeysCalendar.aspx that shows scheduling information for projects in your company. The page is accessed from various other ASP and ASP.NET pages hosted throughout the company's intranet. All employees on the intranet use Internet Explorer.

ITCertKeysCalendar.aspx has a Calendar control at the top of the page. Listed below the Calendar control is detailed information about project schedules on the data selected. When a user selects a date in the calendar, the page is refreshed to show the project schedule details for the newly selected date.

Users report that after viewing two or more dates on ITCertKeysCalendar.aspx, they need to click the browser's Back button several times in order to return to the page they were viewing prior to accessing ITCertKeysCalendar.aspx.

You need to modify ITCertKeysCalendar.aspx so that the users need to click the Back button only once.
What should you do?

A. **Add the following statement to the Page.Load event handler for ITCertKeysCalendar.aspx:**
Response.Expires(0)
B. **Add the following statement to the Page.Load event handler for ITCertKeysCalendar.aspx:**
Response.Cache.SetExpires (DateTime.Now())
C. **Add the following attribute to the Page directive for ITCertKeysCalendar.aspx:**
EnableViewState="True"
D. **Add the following attribute to the Page directive for ITCertKeysCalendar.aspx:**
SmartNavigation="True"

Answer: D

Explanation:
the user's experience of the page by performing the following:
• retaining only the last page state in the browser's history.
This is what is required in this scenario.
• eliminating the flash caused by navigation.
• persisting the scroll position when moving from page to page.
• persisting element focus between navigations.

Option A:
This is not a page expiration problem.

Option B:
This is not a caching problem.

Option C:
The Page.EnableViewState property Gets or sets a value indicating whether the page maintains its view state, and the view state of any server controls it contains, when the current page request ends.

Question 2.
You are a Web developer for ITCertKeys. You are creating an online inventory Web site to be used by employees in Germany and the United States. When a user selects a specific item from the inventory, the site needs to display the cost of the item in both United States currency and German currency. The cost must be displayed appropriately for each locale.

You want to create a function to return the currency in the correct format based on the input parameter.

Which code should you use?

A.
```
Function MyGetDisplayValue(value As Double, _
inputRegion As String) As String
Dim display As String
Dim region As RegionInfo region = New RegionInfo(inputRegion).
display = value.ToString("C")
display += region.CurrencySymbol
Return display
End Function
```
B.
```
Function MyGetDisplayValue(value As Double, _
inputCulture As String) As String
Dim display As String
Dim Local Format As NumberFormatInfo = _
CType(NumberFormatInfo.CurrentInfo.Clone(), _
NumberFormatInfo)
display = value.ToString("C", LocalFormat)
Return display
End Function
```
C.
```
Function MyGetDisplayValue(value As Double, _
inputRegion As String) As String
Dim display As String
Dim region As RegionInfo
region = New RegionInfo(inputRegion)
display = value.ToString("C")
display += region.ISOCurrencySymbol
Return display
End Function
```
D.
```
Function MyGetDisplayValue(value As Double, _
inputCulture As String) As String
Dim display As String
Dim culture As CultureInfo
culture = New CultureInfo(inputCulture)
display = value.ToString("C", culture)
Return display
End Function
```

Answer: D

Explanation:
We create a new CultureInfo object based on the inputCulture parameter. We then produce the result with "C" constant, representing the current culture, and the new CultureInfo object: display = value.ToString("C", culture)

The CultureInfo Class contains culture-specific information, such as the language, country/region, calendar, and cultural conventions associated with a specific culture. This class also provides the

information required for performing culture-specific operations, such as casing, formatting dates and numbers, and comparing strings.

Option B:
The NumberFormatInfo class defines how currency, decimal separators, and other numeric symbols are formatted and displayed based on culture. However, we should create a CultureInfo object, not a NumberFormatInfo object).

Option A, C:
We should use the CultureInfo class not the RegionInfo class.

In contrast to CultureInfo, RegionInfo does not represent preferences of the user and does not depend on the user's language or culture.

Question 3.
You are a Web developer for an airline company ITCertKeys Travels. You are developing a Web site for customers who participate in ITCertKeys's frequent flyer program.

The frequent flyer program includes three levels of award for customers. The levels are named Emerald, Ruby, and Diamond. For each award level, the page contains content specific to that award level. The page contents are contained in three user controls, which are named Emerald.ascx, Ruby.ascx, and Diamond.ascx.

You want to dynamically load and display the proper page header based on the value contained in a variable named awardLevel. The awardLevel variable is a property of the page.
In addition, you want to minimize the amount of memory resources each page uses.
Which code should you use in the Page.Load event handler?

A.
```
Dim headerControl as UserControl
Select Case awardLevel
Case "Emerald"
headerControl = LoadControl("Emerald.ascx")
Case "Ruby"
headerControl = LoadControl("Ruby.ascx")
Case "Diamond"
headerControl = LoadControl("Diamond.ascx")
End Select
Controls.Add(headerControl)
```
B.
```
Dim headerControl As UserControl
Select Case awardLevel
Case "Emerald"
headerControl = LoadControl("Emerald.ascx")
Case "Ruby"
headerControl = LoadControl("Ruby.ascx")
Case "Diamond"
headerControl = LoadControl("Diamond.ascx")
End Select
```
C.
```
emeraldHeaderControl.Visible = False
rubyHeaderControl.Visible = False
diamondHeaderControl.Visible = False
Select Case awardLevel
Case "Emerald"
emeraldHeaderControl.Visible = True
Case "Ruby"
rubyHeaderControl.Visible = True.
Case "Diamond"
```

diamondHeaderControl.Visible = True
End Select

D. Dim emeraldHeaderControl As UserControl
Dim rubyHeaderControl As UserControl
Dim diamondHeaderControl As UserControl
emeraldHeaderControl = LoadControl("Emerald.ascx")
rubyHeaderControl = LoadControl("Ruby.ascx")
diamondHeaderControl = LoadControl("Diamond.ascx")
Select Case awardLevel
Case "Emerald"
Controls.Add(emeraldHeaderControl)
Case "Ruby"
Controls.Add(rubyHeaderControl)
Case "Diamond"
Controls.Add(diamondHeaderControl)
End Select

Answer: A

Explanation:
The TemplateControl.LoadControl method obtains a UserControl object from a user control file.

Option B:
We must add the control in order to display it.

Option C:
We must load the user controls.

Option D:
Loading all three controls increase the demand on the system resoures.

Question 4.
You are creating an ASP.NET application for the mortgage services department of ITCertKeys. The application will be used for generating documents that are required during the closing process of a home purchase.
Woodgrove Bank already has a component written in Visual Basic .NET that identifies which forms are required to be printed based on a set of criteria specified by the closing agent. The name of the component namespace is ITCertKeys.Mortgage. The name of the class is Closing.
You create an ASP.NET page named Purchase.aspx. You add a reference to the assembly that contains the ITCertKeys.Mortgage namespace. The code-behind file for Purchase.aspx includes the following code:
Imports ITCertKeys.Mortgage
You add a method to the code-behind file to instantiate the Closing class. Which code segment should you include in the method to instantiate the class?

A. Dim myClosing As New Closing().
B. Dim myClosing As Closing
closing = Server.CreateObject("Closing")
C. Dim myClosing As System.Object
closing = Server.CreateObject("Closing")
D. Dim myType As Type = _
Type.GetTypeFromProgID("ITCertKeys.Mortgage.Closing" _
, "localhost", True)

Answer: A

Explanation:
We simply instantiate an object with the class with the New constructor.
Web Forms pages have code-behind files associated with them. These files are created automatically when you create a new Web form. They have the same base name as the Web form with the .vb or .cs filename extension added

Option B, C:
The CreateObject function creates and returns a reference to a COM object. CreateObject cannot be used to create instances of classes in Visual Basic unless those classes are explicitly exposed as COM components.

Option D:
The Type.GetTypeFromProgID method is provided for COM support. Program IDs are not used in Microsoft .NET Framework because they have been superceded by the concept of namespace.

Question 5.
You create an ASP.NET page that display customer order information. This information is displayed in two separate DataGrid controls on the Page. The first DataGrid control displays the current year orders, and the second DataGrid control displays all orders from previous years. The page uses both the System.Data.SqlClient namespace and the System.Data namespace.
The information is stored in a Microsoft SQL Server Database. A customer's complete order history information is obtained from the database by calling a stored procedure named GetOrders and passing the customer's identification number as a parameter.
The Page.Load event handler populates a DataView object named EXDataView with the result of calling the GetOrders stored procedure. The following code segment in the Page.Load event handler is then used to bind the two DataGrid controls to EXDataView:

```
dataGridCurrentYear.DataSource = EXDataView
EXDataView.RowFilter = "OrderDate >= #01/01/" & _
Now.Year & "#"
dataGridCurrentYear.DataBind()
dataGridPreviousYears.DataSource = EXDataView
EXDataView.RowFilter = "OrderDate < #01/01" & _
Now.Year & "#"
dataGridPreviousYears.DataBind().
Page.DataBind
```

During testing, you discover that both DataGrid controls are displaying order information for the previous years only.
What should you do to correct this problem?

A. Remove the Page.DataBind() statement.
B. Remove the dataGridPreviousYears.DataBind() statement.
C. Add a Response.Flush() statement immediately before the Page.DataBind() statement.
D. Add a Response.Flush() statement immediately before the dataGridPreviousYears.DataBind() statement.

Answer: A

Explanation:
Both datagrids use the same DataView. The Page.Databind method binds a data source to the invoked server control and all its child controls. We should remove this statement.

Option B:
We must bind each data grid control to its data source

Option C, D:

The HttpResponse Flush method sends all currently buffered output to the client. It is not useful in this scenario.

Question 6
You are creating an ASP.NET page for ITCertKeys. You create a DataGrid control that displays past purchases made by the user. The DataGrid control is populated from an existing database when the page is created.

The page contains TextBox controls that allow users to update their personal information, such as address and telephone number.

You need to ensure that the page is refreshed as quickly as possible when users update their contact information.

What should you do?

A. Set the Enable property of the DataGrid control to **false.**
B. Set the EnableViewState property of the DataGrid to **false**.
C. Write code in the Page.Load event handler that populates the DataGrid control only when the IsPostBack property of the page is **false.**
D. Write in the Page.Load event handler that populates the DataGrid control only when the IsPostBack property of the page is **true.**

Answer: D

Explanation:
The Page.IsPostBack property gets a value indicating whether the page is being loaded in response to a client postback, or if it is being loaded and accessed for the first time. The value is true if the page is being loaded in response to a client postback; otherwise, false. By adding code in the Page Load event handler that populates the Data Grid control when the IsPostBack property is true we ensure that the page is refreshed as quickly as possible.

OptionA:
The DataGrid control has an Enabled property, but no Enable property. Furthermore, the Enable property only indicates if the control is enabled or not.

Option B:
The Control.EnableViewState property indicates whether the server control persists its view state, and the view state of any child controls it contains, to the requesting client.

Option C:
The DataGrid should only be populated when the user updates the contact information. This occurs when the IsPostBack property is true, not false. This suggested solution would only populate the DataGrid when the page is loaded the first time.

Question 7.
You create an ASP.NET page named Location.aspx. Location.aspx contains a Web user control that displays a drop-down list box of counties. The Web user control is named CountyList, and it is defined in a file named CountyList.ascx. The name of the DropDownList control in CountyList.ascx is EXCounty.

You try to add code to the Page.Load event handler for Location.aspx, but you discover that you cannot access EXCounty from code in Location.aspx. You want to ensure that code within Location.aspx can access properties of EXCounty.

What should you do?

A. **In the code-behind file for CountyList.ascx, add the following line of code:**
Protected EXCounty As DropDownList
B. **In the code-behind file for CountyList.ascx, add the following line of code:**
Public EXCounty As DropDownList
C. **In the code-behind file for LocationList.aspx, add the following line of code:**
Protected EXCounty As DropDownList
D. **In the code-behind file for Location.aspx, add the following line of code:**
Public EXCounty As DropDownList

Answer: B

Explanation:
We must declare the EXCounty as public in the file in which it is defined (CountyList.ascx).

The **Public** keyword in the Dim statement declares elements to be accessible from anywhere within the same project, from other projects that reference the project, and from an assembly built from the project.

Option A, C:
The Protected keyword in the Dim statement declares elements to be accessible only from within the same class, or from a class derived from this class. However, do not want to protect MyCount, at the contrary we must make it public.

Option D:
We must declare it public in the files in which it is defined, not Location.aspx where it is only used.

Question 8.
You create an ASP.NET application for ITCertKeys Inc. to sell toys online. One of the requirements is that every page must display the company name at the top. You create a Web custom control that encapsulate the company name in a heading element. Your control class named CompanyName inherits from the Control class.
The following HTML code displays the company name:

<h2>Tailspin Toys<h2>
You need to write code in the CompanyName class to display the company header.
Which code should you use?

A. Protected Overrides Sub Render(ByVal output As _
System.Web.UI.HtmlTextWriter)
output.Write("<h2>ITCertKeys Inc</h2>") End Sub
B. Protected Overrides Sub OnPreRender(ByVal e As _
System.EventArgs)
Me.Controls.Add _
(New LiteralControl("<h2> ITCertKeys Inc</h2>")) End Sub
C. Protected Overrides Sub RenderChildren(writer As _
System.Web.UI.HtmlTextWriter)
writer.Write("<h2> ITCertKeys Inc</h2>")
End Sub
D. Protected Overrides Sub OnInit(e As EventArgs)
Me.Controls.Add _
(New LiteralControl("<h2> ITCertKeys Inc</h2>"))
End Sub

Answer: A

Explanation:
You create a rendered custom control' s appearance by overriding the base class' s Render method and writing to the method' s output argument using the HtmlTextWriter utility methods. The most direct approach is to use the Write methods to add the HTML directly to the HtmlTextWriter.
The Control.RenderChildren method outputs the content of a server control's children to a provided HtmlTextWriter object, which writes the content to be rendered on the client. This method notifies ASP.NET to render any Active Server Pages (ASP) code on the page. If no ASP code exists on the page, this method renders any child controls for the server control..

Option B, D:
We should not add controls to the web page, just a header.

Option C:
We should override the render method, not the RenderChildren method, as we want to add content to the page itself, not the controls of the page.

Question 9.
You are creating an ASP.NET application that delivers customized news content over the Internet. Users make selections from and ASP.NET page. Your code creates a DataSet object named EXNewsItems, which contains the news items that meet the criteria selected by the user.

You create a style sheet named NewsStyle.xsl that renders the data in EXnNwsItems in HTML format. You write the following code segment:

Dim doc As XmlDataDocument = new XmlDataDocument(EXNewsItems) Dim tran As XslTransform = New XslTransform() tran.Load(“NewsStyle.xsl”)

You want to display the transformed data as HTML text. Which line of code should you add to the end of the code segment?

A. tran.Transform(doc, Nothing, Response.OutputStream)
B. tran.Transform(doc, Nothing, Request.InputStream)
C. EXNewsItems.WriteXml(Response.OutputStream)
D. EXNewsItems.WriteXml(tran.ToString())

Answer: A

Explanation:
The XslTransform.Transform method transforms the XML data in the XpathNavigator using the specified args and outputs the result to a Stream. We should use the Response.OutputStream to enable output of text to the outgoing HTTP response stream.

Option B:
We want to display data, not read data, so we must use Response.OutputStream not Request.InputStream.

Option C, D:
We want to generate HTML, not XML data. We should use the XslTransform.Transform method, not the DataSet.WriteXml method..

Question 10.
You are creating an ASP.NET page that enables users to select a country and view information on tourist attractions in that country. Users select a country from a list box named EXcountryList. The list box displays country names. The list box also contains hidden country codes.

Your code retrieves a cached DataTable object that contains tourist attraction description and a numeric country code named CountryID. The DataTable object is named attractionsTable.
You want to extract an array of DataRow objects from the DataTable object. You want to include tourist attractions for only the selected county.
Which code segment should you use?

A. Dim result As DataRow() = _
attractionsTable.Select(_
"CountryID = " & EXcountryList.SelectedItem.Text)
B. Dim result As DataRow() = _
attractionsTable.Select(_
"CountryID = " & EXcountryList.SelectedItem.Value)
C. Dim result As DataRow = _
attractionsTable.Rows.Find(_
"CountryID = " & EXcountryList.SelectedItem.Value)
D. Dim result As DataRow = _
attractionsTable.Rows.Find(_
EXcountryList.SelectedItem.Value)

Answer: B

Explanation:
The DataTable.Select method gets an array of all DataRow objects that match the filter criteria in order of primary key (or lacking one, order of addition.). The filter will compare CountryID values. We should use Country codes and not country names. We should therefore use the Value of the selected item, not the Text.

Option A:
The ListBox.TextBox property gets or searches for the text of the currently selected item in the ListBox. However, this would retrieve names of countries, but the filter use comparison to a CountryID column. We must use the country code, not the country name.

Option C, D:
The DataRowCollection.Find method is not appropriate in this scenario. It retrieves only a single row, not an array of rows.

Part 2 Creating and Managing Components and .NET Assemblies

Question 1.
You create an ASP.NET application named EXApp. You create an assembly named EXApp.dll in a directory named EXDir. The assembly includes a default resource file named strings.resources that adequately support English-speaking users of the application. You create an additional resource file named strings.ja.resources to enable support for Japanese-speaking users. The resource file is located in the EXDir/ja subdirectory. You want to create a satellite assembly for EXApp.dll that will use the new resource file.
What should you do?

A. Run the Assembly Linker (Al.exe) to embed strings.ja.resources in the output assembly. Place the output assembly in EXDir.
B. Run the Assembly Linker (Al.exe) to embed strings.ja.resources in the output assembly. Place the output assembly in EXDir/ja.
C. Run the Assembly Linker (Al.exe) to link strings.ja.resources to the output assembly. Place the output assembly in EXDir.
D. Run the Assembly Linker (Al.exe) to link strings.ja.resources to the output assembly. Place the output assembly in EXDir/ja.

Answer: B

Explanation:
Assemblies contain resources. We embed the strings.ja.resources in the assembly.
After you have compiled your satellite assemblies, they all have the same name. The runtime differentiates between them based upon the culture specified at compile time with Al.exe's /culture option and by each assembly's directory location.

Option A:
We must put the Japanese assembly into a separate folder.

Option C, D:
We must embed the resource file within the assembly, not link it.

Question 2.
You are creating an ASP.NET application that will display facts about the solar system. This application will support localization for users from France, Germany, Japan, and the United States. To see information about a particular planet, the user will select the planet from a drop-down list box on SolarSystem.aspx.

You want to display the planet names in the drop-down list box in the language appropriate to the individual who is using the application.
What should you do?

A. Create a database table named Planets.
Create three column named PlanetID, LocaleID, and Description.
Use SqlCommand.ExecuteReader to query the table for the locale specified in the request.
Using the locale specified in the request, trsnslate the values by using the TextInfo.OEMCodePage property.
Populate the drop-down list box with the trsnslated text.
B. Create a DataTable object named Planets.

Populate the Planets DataTable object by using string constants.
Using the locale specified in the request, trsnslate the values by using a UnicodeEncoding object.
Bind the DataSource property of the drop-down list box to the DataTable object.

C. Create a database table named Planets.
Create two columns named PlanetID and Description.
Use a SqlDataAdapter to load the planet information into a DataSet object.
Using the locale specified in the request, use the String format provider to trsnslate the values.
Bind the DataSource property of the drop-down list box to the DataSet.DefaultView object.

D. Create string resources assemblies for each locale.
Using the locale specified in the request, use a ResourceManager to load the appropriate assembly.
Populate an array with the string values from the assembly.
Bind the DataSource property of the drop-down list box to the array.

Answer: D

Explanation:
The ResourceManager class provides convenient access to culture-correct resources at run time.

Question 3.
You are creating an ASP.NET application that will be published in several languages. You develop a satellite assembly that will include the localized resources for one of the other languages. The satellite assembly will also contain code that accesses Enterprise Services. Your company has a build team that is responsible for compiling and publishing all software applications created by your group. The build team is also responsible for digitally signing the software with a public/private key pair.

The build team permits you to have access to your company's public key, but not the private key. In order to test your localized satellite assembly, you need to digitally sign the assembly.

What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two)

A. Create a test certificate for your satellite assembly by using the Software Publisher Certificate Test tool (Cert2spc.exe).
B. Compile the satellite assembly by using the Resource File Generator (Resgen.exe) with the /compile switch.
C. Compile the satellite assembly by using the Assembly Linker (Al.exe) with the /delay+ switch..
D. Use the Global Assembly Cache tool (Gacutil.exe) to install the assembly in the global assembly cache.
E. Generate a new public/private key pair by using the Strong Name tool (Sn.exe).
Use the new key pair to sign the assembly temporarily for testing purposes.

Answer: C & E

Explanation:
The **/delay** switch specifies whether the assembly will be fully or partially signed. When an assembly is delay signed, Al.exe does not compute and store the signature, but just reserves space in the file so the signature can be added later.

The Strong Name tool helps sign assemblies with strong names. Sn.exe provides options for key management, signature generation, and signature verification.

The -R and –Rc options are useful with assemblies that have been delay signed. In this scenario, only the public key has been set at compile time and signing is performed later when the private key is known.

Option A:
The Software Publisher Certificate Test tool creates a Software Publisher's Certificate (SPC) from one or more X.509 certificates. Cert2spc.exe is for test purposes only. However, there is no need of a SPC since we already have access to the company's public key.

Option B:
Resgen is not useful for signing assemblies.

The Resource File Generator converts .txt files and .resx (XML-based resource format) files to common language runtime binary .resources files that can be embedded in a runtime binary executable or compiled into satellite assemblies.

Option D:
The Global Assembly Cache tool allows you to view and manipulate the contents of the global assembly cache and download cache. However, it cannot be used to digitally sign an assembly.

Question 4.
You create an assembly to access data in a relational database. This assembly will be used by several ASP.NET applications on your Web server.

You need to ensure that all your applications can access the assembly. Which two actions should you take? (Each correct answer presents part of the solution. Choose two)

A. Run the Assembly Registration tool (Regasm.exe).
B. Run the Strong Name tool (Sn.exe).
C. Run the Installer tool (Intallutil.exe).
D. Run the Global Assembly Cache tool (Gacutil.exe).

Answer: B & D

Explanation:
The Strong Name tool helps sign assemblies with strong names.

There are two ways to install an assembly into the global assembly cache:
• Using Microsoft Windows Installer 2.0. This is not an option here.
• Using the Global Assembly Cache tool (Gacutil.exe).

Option A:
The Assembly Registration tool reads the metadata within an assembly and adds the necessary entries to the registry, which allows COM clients to create .NET Framework classes transparently.

Option C:
The Installer tool allows you to install and uninstall server resources by executing the installer components in a specified assembly.

Question 5.
You are a member of a team of developers creating several ASP.NET applications for ITCertKeys. You want to create a reusable toolbar that will be used in each of the applications. The toolbar will be displayed at the top of each page viewed by the user.
The contents of the toolbar will vary depending on options each user selects when creating a profile.

You want to be able to add the toolbar to the ASP.NET toolbox for each of the developers on your team.

What should you do?

A. Create a new Web Control Library project.
   Create the toolbar within a Web custom control.
B. Add a new Web user control to your ASP.NET project.
   Create the toolbar within the Web user control.
C. Add a new Web Form to your ASP.NET project.
   Design the toolbar within the Web Form and save the Web Form with an .ascx extension.
D. Add a new component class to your ASP.NET project.
   Design the toolbar within the designer of the component class.

Answer: A

Explanation:
Web custom controls are compiled code, which makes them easier to use but more difficult to create. You can add a Web custom control to the Toolbox and display it in a visual designer with full Properties window support and all the other design-time features of ASP.NET server controls.

Option B:
Web user controls are easy to make, but they can be less convenient to use in advanced scenarios such as this. Because Web user controls are compiled dynamically at run time they cannot be added to the Toolbox

Option C:
A Web form would be inadequate.

Option D:
The Component class Provides the base implementation for the IComponent interface and enables object-sharing between applications. It does not fit in this scenario.

Part 3 Consuming and Manipulating Data

Question 1.
You are a Web developer for ITCertKeys. You create an ASP.NET application that accesses sales and marketing data. The data is stored in a Microsoft SQL Server 2000 database on a server named ITCertKeys01.

The company purchases a factory automation software application. The application is installed on ITCertKeys01, where it creates a second instance of SQL Server 2000 named Factory and a database named FactoryDB. You connect to FactoryDB by using Windows Integrated authentication.

You want to add a page to your ASP.NET application to display inventory data from FactoryDB. You use a SqlConnection object to connect to the database. You need to create a connection string to FactoryDB in the instance of SQL Server named Factory on ITCertKeys01.

Which string should you use?

A. "Server=ITCertKeys01;Data Source=Factory;
Initial Catalog=FactoryDB;Integrated Security=SSPI"
B. "Server= ITCertKeys01;Data Source=Factory;
Database=FactoryDB;Integrated Security=SSP1"
C. "Data Source= ITCertKeys01\Factory;Initial Category=Factory;
Integrated Security=SSP1"
D. "Data Source= ITCertKeys01\Factory;Database=FactoryDB;
Integrated Security=SSP1"

Answer: D

Explanation:
The Data Source attribute of the connection string contains the name, instance or network address of the instance of SQL Server to which to connect. In this scenario we are to connect to the Factory Instance on ITCertKeys01 so we use ITCertKeys01\Factory as data source.

To specify the database we should either use the Database or the Initial Catalog attribute. Here we use **Database=FactoryDB**.

The SQL Server .NET Data Provider provides connectivity to Microsoft SQL Server version 7.0 or later using the SqlConnection object. The connection string includes the source database name, and other parameters needed to establish the initial connection.

Option A, B:
There is no **Server** attribute in the connection string. Instead we should use the Data Source attribute to specify the server and the instance.

Option C:
There is no **Initial Category** attribute in the connection string. We can use Database or the Initial Catalog attribute to specify the database.

Question 2.

You create an ASP.NET page that retrieves product information from a Microsoft SQL Server database named ITCertKeysDB. You want to display the list of products in a Repeater control named repeaterProducts. Your code uses the System.Data namespace and the System.Data.SqlClient namespace.

You write the following procedure to retrieve the data:

```
Private Sub RepeaterBind( _
ByVal ConnectionString As String, _
ByVal SQL As String)
Dim da As SqlDataAdapter
Dim dt As DataTable
da = New SqlDataAdapter(SQL, ConnectionString)
dt = New DataTable()
```

You need to add code that will fill repeaterProducts with data retrieved from the database. Which code segment should you use?

A.
```
repeaterProducts.DataSource = dt
repeaterProducts.DataBind()
da.Fill(dt)
```
B.
```
da.Fill(dt)
repeaterProducts.DataBind()
repeaterProducts.DataSource = dt
```
C.
```
repeaterProducts.DataBind()
da.Fill(dt)
repeaterProducts.DataSource = dt
```
D.
```
da.Fill(dt)
repeaterProducts.DataSource = dt
repeaterProducts.DataBind()
```

Answer: D

Explanation:
First we must fill the data set. Then we specify the data source, and finally we bind the data to the control.

Using data-access objects in code follows the sequence:
1. Create the data connection object.
2. Create a data adapter object.
3. Create a data set object.
4. Invoke methods on the adapter object to fill or update the data set.
This scenario: da.Fill(dt) 5. Use data binding or another technique to display the data from the data set.
This scenario:
repeaterProducts.DataSource = dt repeaterProducts.DataBind()

Option A:
We must start by filling the data set.

Option B:
We must specify the data source before we bind the control to the data..

Option C:
We must start by filling the data set.

Question 3.

Your ASP.NET application enables customers to create new sales orders. The sales orders are stored in a Microsoft SQL Server database table named ITCertKeysOrders. The table has an IDENTITY column named OrderID.

Your code uses a DataTable object to manage the order data. The DataTable object contains a column named OrderNumber. You use the Update method of a SqlDataAdapter object to call a stored procedure that inserts each new order into the database. The stored procedure uses a parameter to return the new OrderID value for each order.

You assign a SqlCommand object to the InsertCommand property of the SqlDataAdapter object. You add a SqlParameter object to the Parameters collection of the SqlDataAdapter object, specifying the name and data type of the parameter.

You need to set properties of the SqlParameter object to retrieve new OrderID values from the database into the OrderNumber column of your DataTable object.
What should you do?

A. Set the Direction property to ParameterDirection.ReturnValue.
   Set the SourceColumn property to “OrderID”.
B. Set the Direction property to ParameterDirection.ReturnValue.
   Set the SourceColumn property to “OrderNumber”.
C. Set the Direction property to ParameterDirection.Output.
   Set the SourceColumn property to “OrderID”.
D. Set the Direction property to ParameterDirection.Output.
   Set the SourceColumn property to “OrderNumber”.

Answer: D

Explanation:
As the stored procedure uses a parameter to return the new OrderID value we need to use an output paramater. This is accomplished by setting the Direction property to ParameterDirection.Output.
The SqlParameter.SourceColumn property gets or sets the name of the source column that is mapped to the DataSet and used for loading or returning the Value. The source column, where the value will be stored, is the OrderNumber column.
SqlParameter.Direction property gets or sets a value indicating whether the parameter is input-only, output-only, bidirectional, or a stored procedure return value parameter.

Option A, B:
The scenario clearly states that the stored procedure uses a parameter, not a return value, to return the new OrderID value. We should not set the Direction property to ParameterDirection.ReturnValue

Option C:
The output parameter should be stored in the OrderNumber column. We must set the SourceColumn property to the OrderNumber column.

Question 4.
You create an ASP.NET application. The application uses integrated security to retrieve information from a Microsoft SQL Server database named ITCertKeysOrder. You need to provide a connection string for the application to use to connect to ITCertKeysOrder.

You decide to store the connection string in the Web.config file.

How should you set up the Web.config file?

A. In the configuration section, create an element named appSettings.
Create and **add** element that has a **key** attribute set to SQLConnection, and a **value** attribute set to the connection string.
B. In the configuration section, create an element named SQLConnection.
Create a **key** element that has a **value** attribute set to the connection string.
C. In the authorization section, create an element named SQLConnection.
Create a **key** element that has a **value** attribute set to the connection string.
D. In the authentication section, create an element named appSettings.
Create an element named SQLConnection that has a **value** attribute set to the connection string.

Answer: A

Explanation:
The appSettings element contains custom application settings. The appSetting element is placed in the configuration section. Each element added to the appSettings element has a key attribute and a value attribute. For example SQLConnection and a connection string respectively.

Option B, C:
SQLConnection is a class, and cannot be used as an element in the configuration section of a Web.config file.

Option D:
The appSetting element is placed in the configuration section, not in the authentication section.

Question 5.
You are creating an ASP.NET page for ITCertKeys. Employees at the company will use the page to enter suggested names for new products. Each suggestion is saved in a Microsoft SQL Server database. The table in the database for suggestion includes the following three columns.

Column name	Content
EmployeeID	identification number of employee making a suggestion
ProductID	identification number for the product being named
Suggestion	suggested name for product

To add a suggestion to the ASP.NET page, an employee logs on by entering the appropriate EmployeeID and password. The employee then uses a drop-down list box to select a ProductID and uses a grid to enter suggested names for that product. The employee can enter multiple suggestions for a single products before submitting the page.

The database table has a unique index that includes the EmployeeID, ProductID, and Suggestion columns. The unique index does not allow the same suggested name to be recorded twice for the same product by the same employee.

You are using a SqlDataAdapter object to insert the suggestions into the database. If one of the suggested names for a product is a duplicate, the database returns an error to your code. You do not want such errors to interrupts processing. You want your code to continue inserting any remaining suggestions entered by the employee. You also want to be able to access a list of any suggested names that were skipped due to errors.

What should you do?

A. Set the SqlDataAdapter object's ContinueUpdateOnError property to **True** before calling the object's Update method.
B. Enclose your call to the SqlDataAdapter object's Update method in a try/catch block.
In the Catch code, set the object's ContinueUpdateOnError property to **Tr**ue.

C. Create an event handler for the SqlDataAdapter object's RowUpdated event.
In the event handler, if the SqlRowUpdatedEventArgs object's UpdateStatus property has a value of UpdateStatus.ErrorsOccured, then set the SqlDataAdapter object's ContinueUpdateOnErrorProperty to **True**.
D. Create an event handler for the SqlDataAdapter object's RowUpdated event.
In the event handler, if the SqlRowUpdatedEventArgs object's Errors property returns a non-null value, then set the SqlDataAdapter object's ContinueUpdateOnError property to **True**.

Answer: A

Explanation:
The SqlDataAdapter.ContinueUpdateOnError property gets or sets a value that specifies whether to generate an exception, or the row in error when an error is encountered during a row update. If ContinueUpdateOnError is set to true, no exception is thrown when an error occurs during the update of a row. The update of the row is skipped and the error information is placed in the RowError property of the row in error.

Option B:
We should set the ContinueUpdateOnError property to true beforehand, not the Catch code.

Option C, D:
An event handler is not needed. The required functionality is inherent in the SqlDataAdapter class.

Question 6.
You are creating an ASP.NET accounting application that stores and manipulates data in a Microsoft SQL Server database ITCertKeysiDB. One of the pages in the application will be used for performing month-end operations to calculate the balance of all accounts.

When a user clicks a button on the page, you want your code to run several stored procedures to calculate the month-end balances. These procedures must all succeed before the calculated balances can be stored in the database. If any of the procedures fail, then you do not want to store any of the month-end calculated balances. While the procedures are running, you do not want any users to be able to edit, add, or delete the tables affected by the procedures.

What should you do?

A. Create a class derived from System.EnterpriseServices.ServicesComponent to run the stored procedures.
Annotate the class by using a TransactionAttribute type of attribute.
Set the Value property of the attribute to **TransactionOption.RequiresNe**w.
B. Create a master stored procedure. Use this master stored procedure to call the other stored procedures that perform the month-end operations.
Add WITH REPEATABLEREAD to the master stored procedure.
C. Use structured exception handling to catch a SqlException if one of the stored procedures fails.
Use the Procedure property of the SqlException to identify which stored procedure generated the exception, and call a stored procedure to reserve the previous calculations.
D. Set the IsolationLevel property of a SqlTransaction object to **IsolationLevel.Serializabl**e.
Assign the SqlTransaction object to the Transaction property of the SqlCommand object.
Use a SqlCommand object to run the stored procedures.

Answer: D

Explanation:

We should use an Transaction to ensure that either all stored procedures will succeed or if one stored procedure fails, the whole transaction will be backtracked. Furthermore, in order to protect the data in tables during the transaction, we should use the highest transaction isolation level of Serializable. We use a SQLCommand object to run the stored procedure. We set the Transaction property of the SqlCommand to the SqlTransaction object we created.

The transactionIsolation level of Serializable places a range lock on the DataSet, preventing other users from updating or inserting rows into the dataset until the transaction is complete.

Option A, B:
This is not the way to set up a transaction.

Option C:
Exception handling would be extremely complicated to meet the requirement of the scenario.

Question 7.
You are a software developer at ITCertKeys. You develop a contact management application that will enable users to retrieve information from a central database named ITCertKeysSales. After the data is returned to your application, users must be able to view it, edit it, add new records, and delete existing records. All user changes must then be saved in the database.

Your application design requires several ADO.NET object to work together to accomplish these requirements. You use classes from the System.Data and System.Data.OleDb namespaces.

First you write the code to connect to the database.

Which four actions should you take next? (Each correct answer presents part of the solution. Choose four).

A. Create an OleDbDataAdapter object and define the SelectCommand property.
B. Create an OleDbCommand object and use the ExecuteScalar method.
C. Create a DataTable object as container for the data.
D. Create a DataSet object as a container for the data.
E. Call the DataAdapter.Fill method to populate the DataSet object.
F. Call the DataAdapter.Update method to populate the DataSet object.
G. Call the DataAdapter.Update method to save changes to the database.
H. Call the DataSet.AcceptChanges method to save changes to the database.

Answer: A, D, E & G

Explanation:
First we need to create a DataAdapter, or more specifically an OleDbDataAdapter, object in order to access the data source. We use the SelectCommand property to define an appropriate SQL command.

The data will be stored in a DataSet.

We must populate the DataSet with the DataAdapter.Fill method.

We make updates to the DataSet and then store this changes in the database by the DataAdapter.Update method. The Update method of the DataAdapter is called to resolve changes from a DataSet back to the data source.

Option B:
The ExecuteScalar method returns a single scalar value.

Option C:
A DataTable object is not called for. DataTables are optional.

Option F:
We use the fill, not the update. method to populate the DataSet.

Option H:
The DataSet.AcceptChanges only affects the DataSet. However, we save the changes back to the data source.

Question 8.
You are creating an ASP.NET application for ITCertKeys. Your application will can an XML Web service run by Wide World Importers. The XML Web service will return an ADO.NET DataSet object containing a list of companies that purchase wine.

You need to make the XML Web service available to your application.
What should you do?

A. On the **.NET** tab of the **Reference** dialog box, select System.Web.Services.dll.
B. In the **Web References** dialog box, type the address of the XML Web service.
C. Add a using statement to your Global.asax.cs file, and specify the address of the XML Web service.
D. Write an event handler in the Global.asax.cs file to import the .wsdl and .disco files associated with the XML Web service.

Answer: B

Explanation:
Web references differ from traditional references and components in that they refer to XML Web services published on either a local intranet or the Internet.
Procedure to add a Web reference to a project

1. In **Solution Explore**r, select a project that supports adding Web references.
2. On the **Project** menu, choose **Add Web Referenc**e.
3. In the **Add Web Reference** dialog box, type the URL for the XML Web service in the **Address** text box,
4. Verify that the items in the **Available References** box are the items you want to reference in your project, and then choose **Add Referenc**e.
5. In **Solution Explore**r, expand the **Web References** folder to note the namespace for the Web reference classes that are available to the items in your project.

Question 9.
You create an ASP.NET application that produces sales reports. The sales data is stored in a Microsoft SQL Server database that is used for transaction processing. The application consists of complex Transact-SQL statements.

Many users report that the report generation is taking longer to run each day. You need to improve response times.

What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two)

A. Use an OleDbDataAdapter indexes exist on the SQL Server tables.
B. Ensure that appropriate indexes exist in the SQL Server tables.
C. Rewrite your SQL statements to use aliases for all table names.

D. Rewrite your direct SQL statements as stored procedures and call the stored procedures from your application.
E. Configure queries to run in the security context of the user who is running the query.

Answer: B & D

Explanation:
We use an index to speed access to data in a database table.

When Microsoft SQL Server executes a query, the query optimizer evaluates the costs of the available methods for retrieving the data and uses the most efficient method. SQL Server can perform a table scan, or it can use an index if one exists.

A stored procedure is a batch of SQL Statements that is located on the SQL Server. This saves network bandwidth as the SQL Statements do not have to be send from the client to the SQL Server computer. Furthermore, SQL Server compiles the stored procedures and selects an optimal execution plan. This saves time as well.

Option A:
OleDBDataAdapter can be used to access SQL Server databases. However, they introduce overhead.

Option C:
Aliasing the table names would not improve performance.

Option E:
The security context of the Stored Procedure does not effect performance.

Question 10.
You are creating an ASP.NET application for ITCertKeys. The application will be used to identify potential customers.

Your application will call an XML Web service run by Wide World Importers. The XML Web service will return an ADO.NET DataSet object containing a list of companies that purchase wine. You want to merge this DataSet object into a DataSet object containing a list of companies that are potential customers.

You specify wideWorld as the name of the DataSet object form Wide World Importers, and you specify customerProspects as the name of the DataSet object containing potential customers. After the merge, customerProspects will include the company names in wideWorld.

The two DataSet objects contain tables that have the same names and primary keys. The tables in the two DataSet objects contain columns that have the same names and data types. A table in wideWorld also contains additional columns that you do not want to add to customerProspects. If customerProspects included any tables containing rows with pending changes, you want to preserve the current values in those rows when the merge occurs.

Which lime of code should you use to merge the wideWorld DataSet object into customerProspects DataSet object?

A. customerProspects.Merge (wideWorld, true, MissingSchemaAction.Ignore)
B. customerProspects.Merge (wideWorld, true, MissingSchemaAction.AddWithKey)
C. wideWorld.Merge (customerProspects, true, MissingSchemaAction.Ignore)
D. wideWorld.Merge (customerProspects, true, MissingSchemaAction.Add)

Answer: A

Explanation:
The DataSet.Merge (DataTable, Boolean, MissingSchemaAction) method merges this DataTable with a specified DataTable preserving changes according to the specified argument, and handling an incompatible schema according to the specified argument.
As we want to merge the DataSets into the wideWorld DataSet we should apply the merge method on wideWorld.
The Ignore MissingSchemaAction ignores the extra columns. This meets the requirement not to add the extra columns from the table in wideWorld that contains additional columns.

Option B:
The AddWithKey MissingSchemaAction adds the necessary columns and primary key information to complete the schema. However, we do not want to add any extra columns.

Option C, D:
As we want to merge the DataSets into the customerProspects DataSet we should apply the merge method on customerProspects, not on wideWorld .

Part 3 Consuming and Manipulating Data

Question 1.
You are a Web developer for ITCertKeys. You create an ASP.NET application that accesses sales and marketing data. The data is stored in a Microsoft SQL Server 2000 database on a server named ITCertKeys01.

The company purchases a factory automation software application. The application is installed on ITCertKeys01, where it creates a second instance of SQL Server 2000 named Factory and a database named FactoryDB. You connect to FactoryDB by using Windows Integrated authentication.

You want to add a page to your ASP.NET application to display inventory data from FactoryDB. You use a SqlConnection object to connect to the database. You need to create a connection string to FactoryDB in the instance of SQL Server named Factory on ITCertKeys01.

Which string should you use?

A. "Server=ITCertKeys01;Data Source=Factory;
Initial Catalog=FactoryDB;Integrated Security=SSPI"
B. "Server= ITCertKeys01;Data Source=Factory;
Database=FactoryDB;Integrated Security=SSP1"
C. "Data Source= ITCertKeys01\Factory;Initial Category=Factory;
Integrated Security=SSP1"
D. "Data Source= ITCertKeys01\Factory;Database=FactoryDB;
Integrated Security=SSP1"

Answer: D

Explanation:
The Data Source attribute of the connection string contains the name, instance or network address of the instance of SQL Server to which to connect. In this scenario we are to connect to the Factory Instance on ITCertKeys01 so we use ITCertKeys01\Factory as data source.

To specify the database we should either use the Database or the Initial Catalog attribute. Here we use **Database=FactoryDB**.

The SQL Server .NET Data Provider provides connectivity to Microsoft SQL Server version 7.0 or later using the SqlConnection object. The connection string includes the source database name, and other parameters needed to establish the initial connection.

Option A, B:
There is no **Server** attribute in the connection string. Instead we should use the Data Source attribute to specify the server and the instance.

Option C:
There is no **Initial Category** attribute in the connection string. We can use Database or the Initial Catalog attribute to specify the database.

Question 2.

You create an ASP.NET page that retrieves product information from a Microsoft SQL Server database named ITCertKeysDB. You want to display the list of products in a Repeater control named repeaterProducts. Your code uses the System.Data namespace and the System.Data.SqlClient namespace.

You write the following procedure to retrieve the data:

```
Private Sub RepeaterBind( _
ByVal ConnectionString As String, _
ByVal SQL As String)
Dim da As SqlDataAdapter
Dim dt As DataTable
da = New SqlDataAdapter(SQL, ConnectionString)
dt = New DataTable()
```

You need to add code that will fill repeaterProducts with data retrieved from the database. Which code segment should you use?

A.
```
repeaterProducts.DataSource = dt
repeaterProducts.DataBind()
da.Fill(dt)
```
B.
```
da.Fill(dt)
repeaterProducts.DataBind()
repeaterProducts.DataSource = dt
```
C.
```
repeaterProducts.DataBind()
da.Fill(dt)
repeaterProducts.DataSource = dt
```
D.
```
da.Fill(dt)
repeaterProducts.DataSource = dt
repeaterProducts.DataBind()
```

Answer: D

Explanation:
First we must fill the data set. Then we specify the data source, and finally we bind the data to the control.

Using data-access objects in code follows the sequence:
1. Create the data connection object.
2. Create a data adapter object.
3. Create a data set object.
4. Invoke methods on the adapter object to fill or update the data set.
This scenario: da.Fill(dt) 5. Use data binding or another technique to display the data from the data set.
This scenario:
repeaterProducts.DataSource = dt repeaterProducts.DataBind()

Option A:
We must start by filling the data set.

Option B:
We must specify the data source before we bind the control to the data..

Option C:
We must start by filling the data set.

Question 3.

Your ASP.NET application enables customers to create new sales orders. The sales orders are stored in a Microsoft SQL Server database table named ITCertKeysOrders. The table has an IDENTITY column named OrderID.

Your code uses a DataTable object to manage the order data. The DataTable object contains a column named OrderNumber. You use the Update method of a SqlDataAdapter object to call a stored procedure that inserts each new order into the database. The stored procedure uses a parameter to return the new OrderID value for each order.

You assign a SqlCommand object to the InsertCommand property of the SqlDataAdapter object. You add a SqlParameter object to the Parameters collection of the SqlDataAdapter object, specifying the name and data type of the parameter.

You need to set properties of the SqlParameter object to retrieve new OrderID values from the database into the OrderNumber column of your DataTable object.
What should you do?

A. Set the Direction property to ParameterDirection.ReturnValue.
   Set the SourceColumn property to “OrderID”.
B. Set the Direction property to ParameterDirection.ReturnValue.
   Set the SourceColumn property to “OrderNumber”.
C. Set the Direction property to ParameterDirection.Output.
   Set the SourceColumn property to “OrderID”.
D. Set the Direction property to ParameterDirection.Output.
   Set the SourceColumn property to “OrderNumber”.

Answer: D

Explanation:
As the stored procedure uses a parameter to return the new OrderID value we need to use an output paramater. This is accomplished by setting the Direction property to ParameterDirection.Output.
The SqlParameter.SourceColumn property gets or sets the name of the source column that is mapped to the DataSet and used for loading or returning the Value. The source column, where the value will be stored, is the OrderNumber column.
SqlParameter.Direction property gets or sets a value indicating whether the parameter is input-only, output-only, bidirectional, or a stored procedure return value parameter.

Option A, B:
The scenario clearly states that the stored procedure uses a parameter, not a return value, to return the new OrderID value. We should not set the Direction property to ParameterDirection.ReturnValue

Option C:
The output parameter should be stored in the OrderNumber column. We must set the SourceColumn property to the OrderNumber column.

Question 4.
You create an ASP.NET application. The application uses integrated security to retrieve information from a Microsoft SQL Server database named ITCertKeysOrder. You need to provide a connection string for the application to use to connect to ITCertKeysOrder.

You decide to store the connection string in the Web.config file.

How should you set up the Web.config file?

A. In the configuration section, create an element named appSettings.
Create and **add** element that has a **key** attribute set to SQLConnection, and a **value** attribute set to the connection string.
B. In the configuration section, create an element named SQLConnection.
Create a **key** element that has a **value** attribute set to the connection string.
C. In the authorization section, create an element named SQLConnection.
Create a **key** element that has a **value** attribute set to the connection string.
D. In the authentication section, create an element named appSettings.
Create an element named SQLConnection that has a **value** attribute set to the connection string.

Answer: A

Explanation:
The appSettings element contains custom application settings. The appSetting element is placed in the configuration section. Each element added to the appSettings element has a key attribute and a value attribute. For example SQLConnection and a connection string respectively.

Option B, C:
SQLConnection is a class, and cannot be used as an element in the configuration section of a Web.config file.

Option D:
The appSetting element is placed in the configuration section, not in the authentication section.

Question 5.
You are creating an ASP.NET page for ITCertKeys. Employees at the company will use the page to enter suggested names for new products. Each suggestion is saved in a Microsoft SQL Server database. The table in the database for suggestion includes the following three columns.

Column name	Content
EmployeeID	identification number of employee making a suggestion
ProductID	identification number for the product being named
Suggestion	suggested name for product

To add a suggestion to the ASP.NET page, an employee logs on by entering the appropriate EmployeeID and password. The employee then uses a drop-down list box to select a ProductID and uses a grid to enter suggested names for that product. The employee can enter multiple suggestions for a single products before submitting the page.

The database table has a unique index that includes the EmployeeID, ProductID, and Suggestion columns. The unique index does not allow the same suggested name to be recorded twice for the same product by the same employee.

You are using a SqlDataAdapter object to insert the suggestions into the database. If one of the suggested names for a product is a duplicate, the database returns an error to your code. You do not want such errors to interrupts processing. You want your code to continue inserting any remaining suggestions entered by the employee. You also want to be able to access a list of any suggested names that were skipped due to errors.

What should you do?

A. Set the SqlDataAdapter object's ContinueUpdateOnError property to **True** before calling the object's Update method.
B. Enclose your call to the SqlDataAdapter object's Update method in a try/catch block.
In the Catch code, set the object's ContinueUpdateOnError property to **Tr**ue.

C. Create an event handler for the SqlDataAdapter object's RowUpdated event.
In the event handler, if the SqlRowUpdatedEventArgs object's UpdateStatus property has a value of UpdateStatus.ErrorsOccured, then set the SqlDataAdapter object's ContinueUpdateOnErrorProperty to **True**.
D. Create an event handler for the SqlDataAdapter object's RowUpdated event.
In the event handler, if the SqlRowUpdatedEventArgs object's Errors property returns a non-null value, then set the SqlDataAdapter object's ContinueUpdateOnError property to **True**.

Answer: A

Explanation:
The SqlDataAdapter.ContinueUpdateOnError property gets or sets a value that specifies whether to generate an exception, or the row in error when an error is encountered during a row update. If ContinueUpdateOnError is set to true, no exception is thrown when an error occurs during the update of a row. The update of the row is skipped and the error information is placed in the RowError property of the row in error.

Option B:
We should set the ContinueUpdateOnError property to true beforehand, not the Catch code.

Option C, D:
An event handler is not needed. The required functionality is inherent in the SqlDataAdapter class.

Question 6.
You are creating an ASP.NET accounting application that stores and manipulates data in a Microsoft SQL Server database ITCertKeysiDB. One of the pages in the application will be used for performing month-end operations to calculate the balance of all accounts.

When a user clicks a button on the page, you want your code to run several stored procedures to calculate the month-end balances. These procedures must all succeed before the calculated balances can be stored in the database. If any of the procedures fail, then you do not want to store any of the month-end calculated balances. While the procedures are running, you do not want any users to be able to edit, add, or delete the tables affected by the procedures.

What should you do?

A. Create a class derived from System.EnterpriseServices.ServicesComponent to run the stored procedures.
Annotate the class by using a TransactionAttribute type of attribute.
Set the Value property of the attribute to **TransactionOption.RequiresNe**w.
B. Create a master stored procedure. Use this master stored procedure to call the other stored procedures that perform the month-end operations.
Add WITH REPEATABLEREAD to the master stored procedure.
C. Use structured exception handling to catch a SqlException if one of the stored procedures fails.
Use the Procedure property of the SqlException to identify which stored procedure generated the exception, and call a stored procedure to reserve the previous calculations.
D. Set the IsolationLevel property of a SqlTransaction object to **IsolationLevel.Serializabl**e.
Assign the SqlTransaction object to the Transaction property of the SqlCommand object.
Use a SqlCommand object to run the stored procedures.

Answer: D

Explanation:

We should use an Transaction to ensure that either all stored procedures will succeed or if one stored procedure fails, the whole transaction will be backtracked. Furthermore, in order to protect the data in tables during the transaction, we should use the highest transaction isolation level of Serializable. We use a SQLCommand object to run the stored procedure. We set the Transaction property of the SqlCommand to the SqlTransaction object we created.

The transactionIsolation level of Serializable places a range lock on the DataSet, preventing other users from updating or inserting rows into the dataset until the transaction is complete.

Option A, B:
This is not the way to set up a transaction.

Option C:
Exception handling would be extremely complicated to meet the requirement of the scenario.

Question 7.
You are a software developer at ITCertKeys. You develop a contact management application that will enable users to retrieve information from a central database named ITCertKeysSales. After the data is returned to your application, users must be able to view it, edit it, add new records, and delete existing records. All user changes must then be saved in the database.

Your application design requires several ADO.NET object to work together to accomplish these requirements. You use classes from the System.Data and System.Data.OleDb namespaces.

First you write the code to connect to the database.

Which four actions should you take next? (Each correct answer presents part of the solution. Choose four).

A. Create an OleDbDataAdapter object and define the SelectCommand property.
B. Create an OleDbCommand object and use the ExecuteScalar method.
C. Create a DataTable object as container for the data.
D. Create a DataSet object as a container for the data.
E. Call the DataAdapter.Fill method to populate the DataSet object.
F. Call the DataAdapter.Update method to populate the DataSet object.
G. Call the DataAdapter.Update method to save changes to the database.
H. Call the DataSet.AcceptChanges method to save changes to the database.

Answer: A, D, E & G

Explanation:
First we need to create a DataAdapter, or more specifically an OleDbDataAdapter, object in order to access the data source. We use the SelectCommand property to define an appropriate SQL command.

The data will be stored in a DataSet.

We must populate the DataSet with the DataAdapter.Fill method.

We make updates to the DataSet and then store this changes in the database by the DataAdapter.Update method. The Update method of the DataAdapter is called to resolve changes from a DataSet back to the data source.

Option B:
The ExecuteScalar method returns a single scalar value.

Option C:
A DataTable object is not called for. DataTables are optional.

Option F:
We use the fill, not the update. method to populate the DataSet.

Option H:
The DataSet.AcceptChanges only affects the DataSet. However, we save the changes back to the data source.

Question 8.
You are creating an ASP.NET application for ITCertKeys. Your application will can an XML Web service run by Wide World Importers. The XML Web service will return an ADO.NET DataSet object containing a list of companies that purchase wine.

You need to make the XML Web service available to your application.
What should you do?

A. On the **.NET** tab of the **Reference** dialog box, select System.Web.Services.dll.
B. In the **Web References** dialog box, type the address of the XML Web service.
C. Add a using statement to your Global.asax.cs file, and specify the address of the XML Web service.
D. Write an event handler in the Global.asax.cs file to import the .wsdl and .disco files associated with the XML Web service.

Answer: B

Explanation:
Web references differ from traditional references and components in that they refer to XML Web services published on either a local intranet or the Internet.
Procedure to add a Web reference to a project

1. In **Solution Explore**r, select a project that supports adding Web references.
2. On the **Project** menu, choose **Add Web Referenc**e.
3. In the **Add Web Reference** dialog box, type the URL for the XML Web service in the **Address** text box,
4. Verify that the items in the **Available References** box are the items you want to reference in your project, and then choose **Add Referenc**e.
5. In **Solution Explore**r, expand the **Web References** folder to note the namespace for the Web reference classes that are available to the items in your project.

Question 9.
You create an ASP.NET application that produces sales reports. The sales data is stored in a Microsoft SQL Server database that is used for transaction processing. The application consists of complex Transact-SQL statements.

Many users report that the report generation is taking longer to run each day. You need to improve response times.

What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two)

A. Use an OleDbDataAdapter indexes exist on the SQL Server tables.
B. Ensure that appropriate indexes exist in the SQL Server tables.
C. Rewrite your SQL statements to use aliases for all table names.

D. Rewrite your direct SQL statements as stored procedures and call the stored procedures from your application.
E. Configure queries to run in the security context of the user who is running the query.

Answer: B & D

Explanation:
We use an index to speed access to data in a database table.

When Microsoft SQL Server executes a query, the query optimizer evaluates the costs of the available methods for retrieving the data and uses the most efficient method. SQL Server can perform a table scan, or it can use an index if one exists.

A stored procedure is a batch of SQL Statements that is located on the SQL Server. This saves network bandwidth as the SQL Statements do not have to be send from the client to the SQL Server computer. Furthermore, SQL Server compiles the stored procedures and selects an optimal execution plan. This saves time as well.

Option A:
OleDBDataAdapter can be used to access SQL Server databases. However, they introduce overhead.

Option C:
Aliasing the table names would not improve performance.

Option E:
The security context of the Stored Procedure does not effect performance.

Question 10.
You are creating an ASP.NET application for ITCertKeys. The application will be used to identify potential customers.

Your application will call an XML Web service run by Wide World Importers. The XML Web service will return an ADO.NET DataSet object containing a list of companies that purchase wine. You want to merge this DataSet object into a DataSet object containing a list of companies that are potential customers.

You specify wideWorld as the name of the DataSet object form Wide World Importers, and you specify customerProspects as the name of the DataSet object containing potential customers. After the merge, customerProspects will include the company names in wideWorld.

The two DataSet objects contain tables that have the same names and primary keys. The tables in the two DataSet objects contain columns that have the same names and data types. A table in wideWorld also contains additional columns that you do not want to add to customerProspects. If customerProspects included any tables containing rows with pending changes, you want to preserve the current values in those rows when the merge occurs.

Which lime of code should you use to merge the wideWorld DataSet object into customerProspects DataSet object?

A. customerProspects.Merge (wideWorld, true, MissingSchemaAction.Ignore)
B. customerProspects.Merge (wideWorld, true, MissingSchemaAction.AddWithKey)
C. wideWorld.Merge (customerProspects, true, MissingSchemaAction.Ignore)
D. wideWorld.Merge (customerProspects, true, MissingSchemaAction.Add)

Answer: A

Explanation:
The DataSet.Merge (DataTable, Boolean, MissingSchemaAction) method merges this DataTable with a specified DataTable preserving changes according to the specified argument, and handling an incompatible schema according to the specified argument.
As we want to merge the DataSets into the wideWorld DataSet we should apply the merge method on wideWorld.
The Ignore MissingSchemaAction ignores the extra columns. This meets the requirement not to add the extra columns from the table in wideWorld that contains additional columns.

Option B:
The AddWithKey MissingSchemaAction adds the necessary columns and primary key information to complete the schema. However, we do not want to add any extra columns.

Option C, D:
As we want to merge the DataSets into the customerProspects DataSet we should apply the merge method on customerProspects, not on wideWorld .

Part 4 Testing and Debugging

Question 1.
You create an ASP.NET application. You implement tracing and debugging instrumentation. The application is deployed on ITCertKeys's intranet.

After working with the application for several days, users report that some pages are displaying errors that incorrectly identify valid date values as being invalid.

You need to gather debugging information from the application while it is running in the production environment. You need to perform this task with the least impact on the performance of the application.

What should you do?

A. Enable Debug mode in the application's Web.config file on the production server.
Use Visual Studio .NET on your client computer to select **Debug Processes** from the **Tools** menu and attach to the aspnet_wp.exe process on the production server.
B. Enable Debug mode in the application's Web.config file on the production server.
Use Visual Studio .NET on your client computer to open the application project on the production server and select **Start** from the **Debug** menu.
C. Enable application tracing and disable tracing page output in the application's Web.config file on the production server.
View the debugging information on the trace.axd page.
D. Enable application tracing and disable tracing page output in the application's Web.config file on the production server.
Run the DbgClr.exe and attach to the aspnet_wp.exe process on the production server.

Answer: A

Explanation:
We should use remote debugging to minimize the impact on the product server.
Remote debugging is the scenario in which you run Visual Studio .NET on one machine (the client) and debug a Web application running on another machine (the server). For remote ASP.NET debugginf the aspnet_wp.exe process must be debugged.

Option B, C, D:
Running the debug process on the production server would unnecessarily decrease the performance of the production server.

Microsoft common language runtime Debugger (DbgCLR.exe), which is a Windows debugger.

Question 2.
You create an ASP.NET application for ITCertKeys. Your application contains a method named nextBusinessDay. This method uses a date parameter and returns the next date that is not a holiday or weekend day.
You are debugging a page named ProjectTimeline.aspx. You need the execution to break on the following line of code when the value of the dStartDate variable changes:
dStartDate = nextBusinessDay(dStartDate)
What should you do?

A. **Set a breakpoint on the line of code and open the BreakPoint Properties dialog box.**
Specify the following breakpoint condition:

dStartDate <> dStartDate.
Select the is true option.

B. **Set a breakpoint on the line of code and open the BreakPoint Properties dialog box.**
Specify the following breakpoint condition:

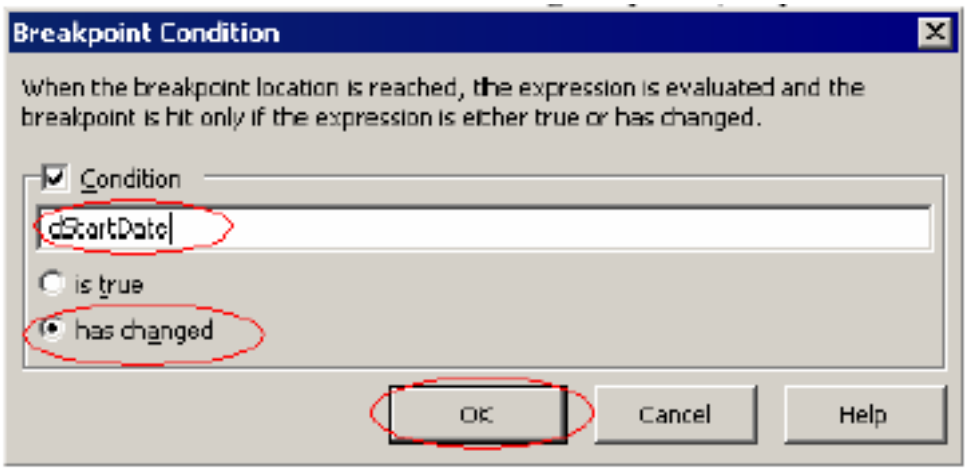
dStartDate
Select the has changed option.

C. **Add the following statement immediately after the call to nextBusinessDay:**
System.Diagnostics.Debug.Assert(_
dStartDate <> dStartDate, “dStartDate has changed.”)

D. **Add the following statement immediately after the call to nextBusinessDay:**
System.Diagnostics.Trace.Assert(_
dStartDate <> dStartDate, “dStartDate has changed.”)

Answer: B

Explanation:
Breakpoints are used to stop a project at a particular line of code. Further conditions for the breakpoint can also be set. In this scenario we specify the condition to be the name of the variable. We also select the **has changed** option (see picture below).



There are four types of breakpoints:

- A function breakpoint causes the program to break when execution reaches a specified location within a specified function.
  We need to specify a function breakpoint in this scenario.
- A file breakpoint causes the program to break when execution reaches a specified location within a specified file.
- An address breakpoint causes the program to break when execution reaches a specified memory address.
- A data breakpoint causes the program to break when the value of a variable changes.
  Visual Basic and C# do not support data breakpoints.

Option A:
The conditions dStartDate <> dStartDate is nonsense. It would always be false.

Option C, D:
As we want to break out of the code we should use breakpoints not assertions. Furthermore, the condition of the assertions, dStartDate <> dStartDate is nonsense.

The difference between the Debug and Trace classes is how they are handled in release builds. By default, Debug methods and properties are automatically stripped out of code compiled for release. Trace methods and properties are retained in release code by default.

Question 3.
You are using your computer to debug and ASP.NET application named ITCertKeysiApp. Your login account has administrative permissions for your computer. ITCertKeysiApp contains several existing ASP pages that use server-side scripts. These server-side scripts are written in Microsoft Visual Basic Scripting Edition.

You locate a line of VBScript code in an existing ASP page that might be incorrect. You add a breakpoint on the line. When you run the application, everything appears to work properly, but the breakpoint is not invoked. When you examine the breakpoint in the VBScript code, you see the following ToolTip: "The breakpoint will not currently be hit. No symbols have been loaded for this document."
You want the breakpoint to be invoked when you run the application in Debug mode.
What should you do?

A. Open the Configuration Manager and set the Active Solution Configuration option to Debug.
B. Select the ASP page in Solution Explorer.
   Set the Build Action property to Compile.
C. Open the property pages for the ASP.NET application and select the Enable ASP Debugging check box.
D. Select Options from the Tools menu.
   Select the Debugging folder.
   In the General category, select the Insert breakpoints in Active Server Pages for breakpoints in client script check box.

Answer: C

Explanation:
We need to enable debugging for the application D is false because the user has set a breakpoint in the SERVER side script and NOT the CLIENT side script.

Question 4.
You use Visual Studio .NET on your client computer to develop an ASP.NET application on a remote server. The application provides asset management functionality.
Another developer at ITCertKeys uses Visual Basic .NET to develop a custom component named EXEXAssetManagement. Your ASP.NET application uses this custom component. The EXAssetManagement component defines an Assets class that exposes a public method named DepreciateAssets(). You deploy EXAssetManagement to the remote server that hosts your ASP.NET application. You add the source files of EXAssetManagement to your ASP.NET application.

You are debugging an .aspx page in your application by using the Visual Studio .NET interactive debugger. The code in the page creates an instance of the Assets class and then calls the DepreciateAssets() method of the instance.

You attempt to step into a call to the DepreciateAssets() method. Instead of showing the first line of code in the DepreciateAssets() method, the interactive debugger moves to the next line of code in the .aspx page.

You need to enable interactive debugger to step into the code within the Assets class.
What should you do in Visual Studio .NET?

A. Configure Visual Studio .NET to enable just-in-time debugging for native programs.

B. Configure Visual Studio .NET to allow editing of Visual Basic files while debugging.
C. In the Configuration Manager, select the Debug configuration and rebuild the EXAssetManagement component.
D. In the Configuration Manager, select the Debug configuration and rebuild the ASP.NET application..

Answer: C

Explanation:
No matter how you start debugging, make sure you build the Debug version of the class library first and make sure the Debug version is in the location where the application expects to find it.

Option A:
Just-In-Time debugging is a technique for debugging a program that is started outside of Visual Studio.

Option B:
This will not help us debug the component.

Option D:
We only have to build the debug version of the class, not rebuild the entire application.

Question 5.
You create an ASP.NET application for a bank. The application provides account management functionality.
A page named AccountWithdrawal.aspx contains a method named WithdrawFunds. The WithdrawFunds method is defined in the following code segment. (Line numbers are included for reference only.)

```
1   Private Function WithdrawFunds(Amount As Double)_
    as Double
2
3   m_dAccountBalance-= Amount
4   Return m_dAccountBalance
5   End Function
```

The callers of this method need to verify that sufficient funds exist in the account before attempting to withdrawal. During unit testing, you want to receive notification when a call is made requesting a withdrawal amount for which the account does not have sufficient funds available.

You plan to build the production version of your application by using the Release Build Configuration in Visual Studio .NET. You need the testing instrumentation to be included but not enabled in the application when the application is deployed to production. You need to have the ability to enable the instrumentation after deploying it to production without requiring the application to be rebuilt.
Which code should you insert at line 2 of the code segment?

A. Debug.Assert(m_dAccountBalance – Amount >=0,_
“Insufficient funds for withdrawal.”).
B. Trace.Assert(m_dAccountBalance – Amount >=0,_
“Insufficient funds for withdrawal.”)
C. Debug.WriteLineIf(m_dAccountBalance - >=0,_
“Insufficient funds for withdrawal.”)
D. Trace.WriteLineIf(m_dAccountBalance – Amount >=0,_
Insufficient funds for withdrawal.”)

Answer: B

Explanation:
As we want to the ability to enable the instrumentation after deployment we must use tracing. The Trace.Assert statement will stop the execution and display the message when the condition is appropriate.

The term instrumentation refers to an ability to monitor or measure the level of a product's performance and to diagnose errors.

Option A, C:
Debug assertions would only enable tracing on in the development environment, not on the deployed systems.

Option D:
The Trace.WriteLineIf method writes information about the trace without stopping the execution. It is better to use an Assert statement, since we need to ensure that the end user is notified of the condition.

Question 6.
You create an ASP.NET application for an online shopping site. The application uses a Microsoft SQL Server 2000 database. The database contains a stored procedure named getProductsByCategory that returns all products that match a specified category code. The category code is supplied as a parameter named @ProdCode.

The application includes a page named ShowProducts.aspx. You are using Visual Studio .NET to debug ShowProducts.aspx.

ShowProducts.aspx uses the getProductsByCategory stored procedure to populate a DataSet object. You set a breakpoint within getProductsByCategory so that you can step through the stored procedure within the debugger.

Which you are debugging getProductsByCategory, you need to view the current value of @ProdCode.

What should you do?

A. Open the Locals debugging window.
B. Open the Modules debugging window.
C. Add the following line of code to getProductsByCategory:
   Print @ProdCode Open the Output debugging window and select **Debug** as the source from the drop-.down list box.
D. Add the following line of code to getProductsByCategory:
   SELECT @ProdCode As DebugOutput Open the Output debugging window and select **Database Output** as the source from the drop-down list box.

Answer: A

Explanation:
The Locals window displays variables local to the current context if the debugger is in break mode.

Option B:
The Modules window lists the modules (DLLs and EXEs) used by your program and shows relevant information for each.

Option C, D:
SQL statements would not provide the required functionality.

Question 7
You are debugging an ASP.NET application that was written by other developers at ITCertKeys. The developers used Visual Studio .NET to create the application. A TextBox control on one of the .aspx pages incorrectly identifies valid data values as being invalid.

You discover that the validation logic for the TextBox control is located within a method that is defined in client-side code. The client-side code is written in Visual Basic Scripting Edition. You want to verify that the validation method is receiving valid input parameters when the page is running. You need to perform this task by stepping through the client-side code as it runs.

Which four courses of action should you take? (Each correct answer presents part of the solution. Choose four)

A. In Internet Explorer, cleat the **Disable script debugging** check box in the advanced options and browse to the page that contains the client-side code.
B. In Visual Studio .NET, select **Debug Processes** from the **Tools** menu and attach to the local copy of IExplore.exe.
In the Running Document window, select the .aspx page that you want to debug.
C. Create a new active solution configuration named Client and copy the settings from the Release configuration.
Select the new configuration in the Configuration Manager.
D. Set the following attribute in the application's Web.config file:
debug="true".
E. In Solution Explorer, open the source for the .aspx file that you want to debug and select **Start** from the **Debug** menu.
F. In Visual Studio .NET, set a breakpoint or add a Stop statement in the client-side code where you want to begin interactive debugging.
G. In Internet Explorer, perform the actions that cause the client-side code to run.

Answer: A, B, F & G

Explanation:
To debug script you must enable script debugging.
To enable script debugging
1. In Internet Explorer, click the **Tools** menu and choose **Internet Option**s.
2. Click the **Advanced** tab.
3. Under the **Browsing** category, clear the **Disable Script Debugging** checkbox.

From within Visual Studio, you can use debugger commands to attach to the browser process (Iexplore.exe) and break into the script.

In Visual Studio .NET we set breakpoint, Stop statements in the client-side code.

We perform the actions in Internet Explorer that causes the client-side code to run.

Option :C, D, E:
These steps are not required.

Question 8.
You are creating an ASP.NET application for an online stock trader ITCertKeys. You need to allow customers to Transfer funds between accounts. You write a component in Visual Basic .NET to handle Transfers of founds. This component is used by the page named EXFundsTransfer.aspx.

For unit testing, you add the following code segment to the EXTransferFunds method of your component. (Line numbers are included for reference only.

1 Dim ctx As HttpContext 2 ctx =HttpContext.Current 3 ctx.Trace.Write(“Founds Transfer requested.”)

You want to be able to view the trace output on the EXFundsTransfer.aspx page. What should you do?

A. Add code to the EXFundsTransfer.aspx page that instantiates a Trace listener.
B. Enable tracing in the Page directive for the EXFundsTransfer.aspx page.
C. Add the following attribute to the Machine.config file:
<trace enabled=”true”>
D. Modify line 3 of the code segments as follows:
System.Diagnostics.Trace.WriteIf(_
ctx.IsDebuggingEnabled, “Funds Transfer requested.”)

Answer: B

Explanation:
You can control whether tracing is enabled or disabled for a page with the Trace attribute of the @ Page directive.

Option A:
This is not the procedure to configure tracing of a page.

Option C:
This would enable tracing of all application on this computer.

Option D:
This would only write the trace message if current HTTP request is in debugging mode. Furthermore, we need to enable tracing of the page.

Question 9.
You are creating an ASP.NET application for ITCertKeys’s human resources (HR) department. Users in the HR department will use the application to process new employees. The application automates several activities that include creating a network login account, creating an e-mail account, registering for insurance benefits, and other activities.

During integration testing of your application, you need to verify that the individual activities run successfully and in the proper order.

Each page in your application includes the following elements in the Page directive:
Debug=”True” Trace=”True”

You want each page to provide execution information in the Web browser immediately after the page’s normal display output. You need to add instrumentation to the code in your pages to accomplish this goal.

Which statement should you use?

A. Trace.Write
B. Debug.Print
C. System.Diagnostics.Trace.Write
D. System.Diagnostics.Debug.Write
E. System.Diagnostics.Debugger.Log

Answer: A

Explanation:
We simply use the Trace.Write method.

Option:
B, D, E: As we want to test the product during integration we need to trace the application, not only debug it.

Question 10.
ITCertKeys's project team develops an order fulfillment ASP.NET application. The application is hosted on a single server named ITCertKeys1.

You are responsible for verifying and correcting problems identified by the quality assurance team. The quality assurance team reports that freight costs are not being calculated accurately.

You need to replicate the problem in order to resolve it.

You attempt to use the interactive debugger from your client computer to step through the ASP.NET application code on ITCertKeys1. You are not able to initiate a debugging session, and the following entry is added to the Application event log on your computer: "DCOM got error 'General access denied error' from the computer ITCertKeys1 when attempting to activate the server."

You need to enable remote debugging.
What should you do?

A. Add your user account to the Power Users group on your client computer.
B. Add your user account to the Power Users group on ITCertKeys1.
C. Add your user account to the Debugger Users group on your client computer.
D. Add your user account to the Debugging Users group on ITCertKeys1.

Answer: D

Explanation:
The remote server must grant the debugger access. To grant access to a user, you must add the user to the Debugger User group on the server. This permission is required even if the debugger user is Administrator on the remote server.

Option A, B:
The Power Users group does not allow remote debugging.

Option C:
The user should be added to the Debugger Users group on the Server, not on the client computer.

Part 5 Deploying a Web Application

Question 1.
You create English, French, and German versions of your ASP.NET application. You have separate resource files for each language version. You need to deploy the appropriate resource file based on the language settings of the server.
What should you do?

A. Create an installer and set the Installer.Context property for each version of your application.
B. Create an installer that has a launch condition to verify the locale settings.
C. Create an installer that has a custom action to install only location-specific files.
D. Create an installer that has an MsiConfigureProduct function to install the appropriate version.

Answer: C

Explanation:
Custom actions are a Windows Installer feature that allows you to run code at the end of an installation to perform actions that cannot be handled during installation. This is an appropriate solution for this scenario as we only want to deploy the resource files on the server.
Resources can be composed of a wide range of elements, including interface elements that provide information to the user (for example a bitmap, icon, or cursor); custom resources that contain data an application needs; version resources that are used by setup APIs; and menu and dialog box resources.

Option A:
We just want to deploy the resource files. We do not need to set the Context property in the application.

Option B:
We don't need any launch conditions. We just want to deploy the resource files.

Option D:
We just want to deploy the resource files.

Question 2.
You are creating an ASP.NET application for an insurance company. The company will use your ASP.NET application to record insurance claims.

Another development team creates a redistributable component that will be used by your ASP.NET application. The component requires several registry entries to be created during installation so that the component will run properly. The same component might be used by other ASP.NET applications in the future.

The development team gives you the source code to the component as well as all of the project files for the component. You add the component project to your ASP.NET application.
You need to create a deployment package for your application. You want to include the redistributable component with your deployment package.
What should you do?

A. Create a setup project for the redistributable component.
Create a Web setup project for your ASP.NET application.

B. Create a merge module project for your ASP.NET application.
Create a setup project for redistributable component and add the merge module for your ASP.NET application to the project.
C. Create a merge module project for both your ASP.NET application and the redistributable component.
Create a Web setup project and add both merge modules to the project.
D. Create a merge module project for the redistributable component.
Create a Web setup project for your ASP.NET application and add the merge module for the redistributable component to the project.

Answer: D

Explanation:
We create a merge module for the redistributable component. We then integrate the merge medole into the Web setup project.
Merge module projects are used to package files or components that will be shared between multiple applications. They create a merge module (.msm) file that includes all files, resources, registry entries, and setup logic for your component. The resulting .msm file can then be merged into other deployment projects, insuring consistent installation of your component across multiple applications.

Option A:
A setup project is used for Windows applications.

Option B, C:
We should not create a merge module from the application.

Question 3.
You create an ASP.NET application for online ordering. You need to store a small amount of page-specific information on pages that are submitted to the server. This information does not need to be secured. The page must work properly for browsers that do not support cookies. You anticipate that the volume of orders on the site will be high, and you need to conserve server resources.

What should you do?

A. Store the information in application state variables.
B. Store the information in session state variables.
C. Store the information in a Microsoft SQL Server database.
D. Store the information in hidden fields on the page..

Answer: D

Explanation:
The advantages of hidden fields are
• No server resources are required. Server resources will be conserved.
• Broad support. It will work on browsers that do no support cookies.
The lack of security, a drawback of hidden fields, is not a problem, since the information do not need to be secured.

State management is the process by which you maintain state and page information over multiple requests for the same or different pages. ASP.NET provides multiple ways to maintain state between server round trips.

Option A:
We want to conserve server resources, so we should avoid using application state variables.

Application state requires server memory, which can affect the performance of the server as well as the scalability of the application.

Option B:
Session state variables stay in memory until they are either removed or replaced, and therefore can degrade server performance.

Option C:
We do not need a complex SQL Server solution.

Question 4.
You create an ASP.NET application that will be sold to ITCertKeys's corporate customers. The corporate customers will buy your application and run it on their intranets.

You create a Web setup project for your application and add it to your ASP.NET solution. You also add a file named Readme.txt to the Web setup project.

You create the deployment package and install it on a test server. You notice that the deployment package installed Readme.txt in the Web application folder. You want the deployment package to add a shortcut to Readme.txt to the desktop on the server computer.

What should you do?

A. Add Readme.txt to your solution and rebuild the deployment package.
B. Select Readme.txt in the Web setup project.
   Change the TargetName property to **DESKTOP\Readme.tx**t.
C. In the Web setup project, add the User's Desktop folder to the **File System on Target Machine** node.
   Add a shortcut to Readme.txt in the User's Desktop folder.
D. In the Web setup project, add a custom folder to the **File System on Target Machine** node.
   Name the folder Server Desktop and add a shortcut to Readme.txt in that folder..

Answer: C

Explanation:
The **User's Desktop folder** contains files and folders that appear on the desktop on a per-user basis. We should add an appropriate shortcut to this folder in the Web setup project.

Special folders are folders in the File System Editor that represent predefined Windows folders. Using special folders in a deployment project allows you to choose a destination folder on a target computer without knowing the actual path to that folder.

Option A:
We need to create a shortcut to the Readme.txt file.

Option B:
The TargetName property Specifies a name for a file when it is installed on a target computer. The file is renamed during installation if it differs from the source file name. However, it is not useful in this scenario..

Option D:
Custom folders are not required.

Custom folders are special folders that represent folders on a target computer. Unlike special folders, custom folders do not necessarily depend on existing folders on the target, but rather allow you to create new folders at install time.

Question 5.
You are planning the deployment of an ASP.NET application. The application uses a Visual Studio .NET component named DataAccess that will be shared with other applications on your Web server.

You are using Visual Studio .NET to create a Windows Installer package. You need to deploy DataAccess and the ASP.NET application so that they can be uninstalled later if necessary. What should you do?

A. Create a setup project for DataAccess.
   Add the ASP.NET application in a custom action.
B. Create a setup project for the ASP.NET application.
   Create another setup project for DataAccess.
C. Create a Web setup project for the ASP.NET application.
   Add a project output for DataAccess..
D. Create a Web setup project for the ASP.NET application.
   Add a merge module for DataAccess.

Answer: D

Explanation:
To deploy a Web application to a Web server, you create a Web Setup project. We should use a merge module in order to be able to uninstall the DataAccess component later.

Merge module projects are used to package files or components that will be shared between multiple applications. They create a merge module (.msm) file that includes all files, resources, registry entries, and setup logic for your component. The resulting .msm file can then be merged into other deployment projects, insuring consistent installation of your component across multiple applications.

Option A, B:
We cannot use a setup project to deploy an ASP.NET Web application. Setup projects are used for Windows applications.

Option C:
A project output cannot be uninstalled separately.

Part 6 Maintaining and Supporting a Web Application

Question 1.
You are a Web developer for ITCertKeys Research Inc. ITCertKeys has a Microsoft SQL Server database that contains information about science experiments. The database currently contains information on 5,000 experiments. The information is updated monthly.

You are creating an ASP.NET application for users to find information about specific experiments. The application will run on a server that is connected to the database by a low-bandwidth leases line.
You want to accomplish the following two goals:

• Minimize the number of times the application accesses the database.
• Minimize the time required for each page of the application to load.

What should you do?

A. Initialize the DataSet object in the Page.Load event handler when the IsPostBack property is false.
Store the DataSet object in an XML file on the user's computer.
B. Create a Cache object for the application and add the DataSet object to the cache.
C. Use the view state to maintain the DataSet object during postback events to the server.
D. Create a Cache object for the session and add the DataSet object to the cache.

Answer: B

Explanation:
The database is only updated monthly. Therefore we can store the dataset in the Application Cache with out the need for that extra over head of reading and writing into the XML file client computer for every session (not A).

Question 2.
You are creating an ASP.NET page for ITCertKeys. The page uses string concatenation to gather data from multiple e-mail messages and format the data for display on the page.
You want to ensure that the page displays as quickly as possible.
What should you do?

A. Write code that uses the Append method of the StringBuilder object.
B. Write code that uses the Substring method of the String object.
C. Write code that uses the Concat method of the String object.
D. Write code that uses the plus-sign (+) operator to concatenate the strings.

Answer: A

Explanation:
The StringBuilder.Append method appends the string representation of a specified object to the end of this instance.The StringBuilder class represents a string-like object whose value is a mutable sequence of characters. The value is said to be mutable because it can be modified once it has been created by appending, removing, replacing, or inserting characters.

Option B:

The Substring method is used to select a part of a string, not to concatenate multiple strings.

Option C:
The String.Concat method Concatenates one or more instances of String, or the String representations of the values of one or more instances of Object. However, compared to the Append method of the StringBuilder object, the Concat method create new instances, and is therefore not the preferred method.

Option D:
Not he best solution.

Question 3.
You are creating an ASP.NET application that uses the Microsoft SQL Server .NET Data Provider to connect to ITCertKeys's database. Your database administrator reports that, due to heavy usage of the application, data requests are being blocked while users wait for new connections to be created.
You want to improve throughput by setting a minimum connection pool size of 10.
What should you do?

A. Add a connection element under an appSettings element in the Web.config file for your application, and specify a minimum size of 10 for the connection pool.
B. Add a connection element under an appSettings element in the Machine.config file on you Web server, and specify a minimum size of 10 for the connection pool.
C.Add a Min Pool Size property to the connection string you use when opening a connection, and specify a minimum size of 10 for the connection pool
D.Add a Min Pool Size property to your ADO.NET connection objects, and assign a value of 10 to the property.

Answer: C

Explanation:
The Min Pool Size property of the connection string denotes the minimum number of connections maintained in the pool.

Option A, B:
The appSettings element contains custom application settings. However, Minimum pool size should be configured in the connection string, not in the custom application settings.

Option D:
Min Pool Size is not a property of a connection object. It is an attribute in the connection string.

Part 7 Configuring and Securing a Web Application

Question 1.
You create an ASP.NET application that will run on ITCertKeys's Internet Web site. Your application contains 100 Web pages. You want to configure your application so that it will display customized error messages to users when an HTTP code error occurs.

You want to log the error when an ASP.NET exception occurs. You want to accomplish these goals with the minimum amount of development effort.

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

A. Create an Application_Error procedure in the Global.asax file for your application to handle ASP.NET code errors.
B. Create an applicationError section in the Web.config file for your application to handle ASP.NET code errors.
C. Create a CustomErrors event in the Global.asax file for your application to handle HTTP errors.
D. Create a customErrors section in the Web.config file for your application to handle HTTP errors.
E. Add the Page directive to each page in the application to handle ASP.NET code errors.
F. Add the Page directive to each page in the application to handle HTTP errors.

Answer: A & D

Explanation:
Any public event raised by the HttpApplication class is supported using the syntax Application_EventName. For example, a handler for the Error event can be declared protected void Application_Error(Object sender, EventArgs e).

The <customErrors> element, which is used in the Web.config file, provides information about custom error messages for an ASP.NET application.

Option B:
There is no such thing as a applicationError section in the Web.config file.

Option C:
There is no such thing as CustomErros event in the Global.asax file.

Option E, F:
It is not necessary to add a Page Directive to each page.

Question 2.
You create a new ASP.NET application named ITCertKeysSalesReports on your development computer. You add code to the default WebForm1. To test the code's functionality, you copy the entire ITCertKeysSalesReports folder from the C:\inetpub\wwwroot folder on your computer to the C:\inetpub\wwwroot folder on a separate Microsoft Windows 2000 Server computer named ITCertKeys1. ITCertKeys1 hosts several ASP.NET applications

When you use the browser on your computer to open the copy of the application hosted on ITCertKeys1, you receive the following error message:

“It is an error to use a section registered as allowDefinition=”MachineToApplication’ beyond application level.”

You want to correct this error without altering the other Web sites that are hosted on ITCertKeys1. What should you do?

A. Use Internet Information Services (IIS) to create a virtual directory that points to the ITCertKeysSalesReports folder on ITCertKeys1.
B. Remove the following element from the Web.config file in C\inetpub\wwwroot\ITCertKeysSalesReports on ITCertKeys1:
<authentication mode=”Windows” /> C. Remove the following element from the Web.config file in C:\inetpub\wwwroot on ITCertKeys1:
<authentication mode=”Windows” />
D. Move the ITCertKeysSalesReports folder on ITCertKeys1 up one level, so that it is a subfolder of the inetpub folder.

Answer: B

Explanation:
You may have defined a section in your application’s web.config file that is not configurable on our shared web hosting platform. Remove or comment out any configuration sections from your web.config file that are not supported. In this scenario we edit the Web.config file in C\inetpub\wwwroot\ITCertKeysSalesReports on ITCertKeys1.

Option A:
Create a virtual directory would solve the problem.

Option C:
We must edit the application’s web.config file, not the Web.config file in the root directory.

Option D:
Moving the application directory would not solve the problem.

Question 3.
Your ASP.NET application displays sales data on a page. You want to improve performance by holding the page in memory on the server for one hour. You want to ensure that the page is flushed from memory after one hour, and that the page is re-created when the next request for the page is received.

What should you do?

A. Initialize a new instance of the Cache class in the Application.Start event handler.
B. Initialize a new instance of the Timer class in the Page.Load event handler.
C. Set the Duration attribute of the OutputCache directive in the page.
D. In the Web.config file, set the timeout attribute of the sessionState element.

Answer: C

Explanation:
ASP.NET allows you to cache the entire response content for dynamic pages on HTTP 1.1 capable mechanisms, including browsers, proxy servers, and the origin Web server where your application resides. This provides a powerful way for you to increase the performance of your Web applications. Called output caching, it allows subsequent requests for a particular page to be

satisfied from the cache so the code that initially creates the page does not have to be run upon subsequent requests.

To set output-cache expirations for a page declaratively. Include an @ OutputCache directive in the page (.aspx file) that you want to output cache. The directive must include a Duration attribute.

Question 4.
You are creating an ASP.NET application for ITCertKeys. Customers will use the application to file claim forms online.
You plan to deploy the application over multiple servers. You want to save session state information to optimize performance.

What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two)

A. Modify the Web.config file to support StateServer mode.
B. Modify the Web.config file to support SQLServer mode.
C. Modify the Web.config file to support InProc mode.
D. In the Session_Start procedure in the Global.asax file, set the EnableSession property of the WebMethod attribute to **tru**e.
E. In the Session_Start procedure in the Global.asax file, set the Description property of the WebMethod attribute to **sessionStat**e.

Answer: A & D

Explanation:
With StateServer mode session state is using an out-of-process Windows NT Server to store state information. This mode is best used when performance is important but you can't guarantee which server a user will request an application from. With out-of-process mode, you get the performance of reading from memory and the reliability of a separate process that manages the state for all servers. As this scenario requires that we should optimize performance, not reliability, StateServer mode is the preferred solution.
The EnableSession property of the WebMethod attribute enables session state for an XML Web service method. Once enabled, the XML Web service can access the session state collection directly.

Option B:
With SQLServer mode session state is using an out-of-process SQL Server to store state information. The SQL Server mode option is similar to that of the Windows NT Service, except that the information persists to SQL Server rather than being stored in memory. This mode is best used when the reliability of the data is fundamental to the stability of the application, as the database can be clustered for failure scenarios. The performance isn't as fast as out of process, but the tradeoff is the higher level of reliability.

Option C:
With InProc mode session state is in process with an ASP.NET worker process. InProc is the default. However, since we are using multiple servers, we cannot use InProc mode.

Option E:
The Description property of the WebMethod attribute supplies a description for an XML Web service method that will appear on the Service help page.

Question 5.
You create an ASP.NET application named TimeSheet for ITCertKeys's intranet. The application will be used only by employees of your company.

You want the application to recognize the user without forcing the user to enter a name and password. You write the following code to save the user's Microsoft Windows login name in the Session object:
Session.Item("User")=User.Identity.Name
When you run the TimeSheet, the Session.Item("User") variable returns an empty string.
You want to configure Internet Information Services (IIS) and your application to be able to retrieve the user name automatically.
What should you do?

A. Disable Anonymous Access for the application in IIS-
B. Enable Basic authentication for the application in IIS
C. Add the following element to the Web.config file for TimeSheet:
<identity impersonate="True" />
D. Add the following element to the Web.config file for TimeSheet:
<identity impersonate="False" />

Answer: A

Explanation:
We should configure IIS to disallow anonymous access and only use Windows authentication.

Option B:
We want to use Windows authentication, not Basic Authentication.

Option C, D:
Impersonating does not apply in this scenario.

Question 6.
You create an ASP.NET application for a hotel. The application contains a page that displays current weather conditions for the city in which the hotel is located. The application calls an XML Web service every 10 minutes to update the current weather conditions. A new page is then displayed for subsequent requests.

You want the same page to be cached and retrieved from the cache during the time between calls to the XML Web service. You decide to use a Page directive to accomplish this goal.
Which page directive should you use?

A. <%@ Cache Seconds="600" VaryByParam="page" %>
B. <%@ OutputCache Time="600" %>
C. <%@ OutputCache Duration="600" VaryByParam="None" %>
D. <%@ OutputCache Duration="600" %>

Answer: C

Explanation:
To set output-cache expirations for a page declaratively include **an @ OutputCache** directive in the page (.aspx file) that you want to output cache. The directive must include a **Duration** attribute, with a positive numeric value, and a **VaryByParam** attribute. The following @ OutputCache directive sets the page's expiration to 10 minutes (600 seconds).
<%@ OutputCache Duration="600" VaryByParam="None" %>

Option A, B:
We should use the **Duration** attribute, not the **Seconds** or the **Time** attribute, to specify the output-cache expiration for a page.

Option D:

We must use a **VaryByParam** attribute.

Question 7.
You create an ASP.NET application for tracking student examinations at ITCertKeys Boot Camp. You use Microsoft Windows authentication. Students are members of a group named Students, and teachers are members of a group named Teachers.
The root folder for your application is named ITCertKeys. The ITCertKeys folder displays information about pending examinations. The ITCertKeys folder has a subfolder named Grades. Both Students and teachers can access pages in ITCertKeys. Only teachers can access page in Grades.
You create the following entries in the Web.config file in ITCertKeys. (Line numbers are included for reference only)

1 <authentication mode="Windows" />
2 <authorization>
3 <allow roles="Students, Teachers" />
4 <deny users="*" />
5 </authorization>

You crate the following entries in the Web.config file in Grades. (Line numbers are included for reference only.)

1 <authentication mode="Windows" />
2 <authorization>
3 <allow roles="Teachers" />
4 <deny users="*" />
5 </authorization>
When teachers try to access pages in the Grades folder, they receive an error message that reads in part: "An error occurred during the processing of a configuration file required to service this request."
You need to ensure that teachers can access pages in the Grades folder.
What should you do?

A. **Remove line 1 in the Web.config file in Grades.**
B. **Modify line 4 in the Web.config file in Grades as follows:**
<allow users="*" />
C. **Add the following line between line 1 and line 2 in the Web.config file in ITCertKeys:**
<identity impersonate="true" />
D. **Add the following line between line 1 and line 2 in the Web.config file in Grades:**
<identity impersonate="true" />
E. **Add the following line between line 1 and line 2 in the Web.config file in Grades**:
<identity impersonate="false" />

Answer: A

Explanation:
The error messages indicates an incorrect line in the configuration file. The first line with the authentication mode element should be removed.
.NET Framework General Reference, <authorization> Element

Option B:
We only want Teachers to have access to the Grades folder. We cannot allow everyone access.

Option C, D, E:
Impersonate does not apply to this scenario.

Impersonation is when ASP.NET executes code in the context of an authenticated and authorized client.

Question 8.
You create an ASP.NET application named Inventory. This application will be used by customers on the Internet.

During the beta test period, you ensure that the actual ASP.NET error message is displayed whenever an error is encountered. Both developers and beta testers see the actual text of the error message.

You perform beta testing of other applications on the same beta test server during the beta testing period for Inventory. All of the other applications display ASP.NET error messages. After the beta testing period is complete, the beta test server is promoted to a production server. You want all applications to display a single, user-friendly error message.

You want to configure Inventory and the production server to meet these goals. You want to perform this task by using the minimum amount of administrative effort.

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

A. Set the mode parameter of the customErrors element in the Web.config file for Inventory to **"On"**.
B. Remove the customErrors element from the Web.config file for Inventory.
C. Set the mode parameter of the customErrors element in the Inventory.config file to **"On"**.
D. Remove the customErrors element from the Inventory.config file.
E. Set the mode parameter of the customErrors element in the Machine.config file to **"On"**.
F. Remove the customErrors element from the Machine.config file.

Answer: B & E

Explanation:
We should remove the customized error message for the Web.config file for the Inventory application.

We should define a single customized error message for all application on the server. This is done be setting the **mode** attribute to **on** in the Machine.config file.

The <customErrors> Element provides information about custom error messages for an ASP.NET application. The mode attribute specifies whether custom errors are enabled, disabled, or shown only to remote clients. The value of **on** specifies that custom errors are enabled.

Option A:
We want to remove the error messages which are specific to the application.

Option C:
We use the Web.config and the Machine.config files, not Inventory.config, to configure appropriate error messages.

Option D:
A Web.config file is used for the Inventory application. No Inventory.config is used.

Option F:
We want to add one single customized error messages for all application on the server. We should add, not remove, a customErrors element to the Machine.config file.

Question 9.
You create an ASP.NET application for ITCertKeys's purchasing department. A page in the application displays a list of products based on the supplier, the product category, or the price. The URL of the page includes this information as parameters.

You want to store multiple versions of your ASP.NET page in the cache based in the parameter values. You want each version of the page to be cached for 300 seconds.

You need to add code to the page to accomplish this goal.

Which code segment should you use?

A. Response.Cache.SetExpires(DateTime.Now.AddSeconds(300)).
Response.Cache.VaryByParams("?") = true
B. Response.Cache.SetExpires(DateTime.Now.AddSeconds(300))
Response.Cache.VaryByParams("All") = true
C. Response.Cache.SetCacheability(HttpCacheability.Public)
Response.Cache.SetLastModified(DateTime.Parse("00:05:00"))
Response.Cache.VaryByParams("All") = true
D. Response.Cache.SetCacheability(HttpCacheability.Public
Response.Cache.SetExpires(DateTime.Now.AddSeconds(300))
Response.Cache.VaryByParams("*") = true

Answer: D

Explanation:
Cachability corresponds to the Location attribute. The Public value corresponds to any location. We use the SetExpires to set the cache duration. Finally we use the "*" string to specify that all parameter values are cached.

Option A, B:
Cachability has to be set.

Option C:
We should use "*", not "all" when specify VaryByParams.

Question 10.
You are a Web developer for a ITCertKeys bookstore. You create a Web user control named EXBookTopics that is defined in a file named EXBookTopics.ascx. EXBookTopics displays a list of book topics based on an author's profile identification number. The profile identification number is stored in a public property of EXBookTopics named AuthorProfile.

You create an ASP.NET page named AuthorPage.aspx that contains an instance of the EXBookTopics Web user control. AuthorPage.aspx is opened by an HTTP-GET request that has two parameters. The parameters are named publisherID and authorProfileID. The value of authorProfileID is a profile identification number.

You want to enable output caching for the EXBookTopics Web user control. You need to ensure that the cached control is varied only by an author's profile identification number.
What should you do?

A. **Add the following element to the OutputCache directive for AuthorPage.aspx:**
VaryByParam="EXBookTopics.AuthorProfile"
B. **Add the following element to the OutputCache directive for AuthorPage.aspx:**
VaryByControl="EXBookTopics.AuthorProfile"

C. **Add the following element to the OutputCache directive for EXBookTopics.ascx:** VaryByParam=”none”
D. **Add the following element to the OutputCache directive for EXBookTopics.ascx:** VaryByControl=”authorProfileID”

Answer: D

Explanation:
You can vary user control output to the cache in two ways:

1) With the user control name and the parameter. The VaryByParam attribute of the @ OutputCache directive must be used. A) and C) are inadequate since both the control name and the parameter must be specified.

2) With the **VaryByControl** attribute just the parameter should be supplied. This is the case in D), but not in B).

Part 6 Maintaining and Supporting a Web Application

Question 1.
You are a Web developer for ITCertKeys Research Inc. ITCertKeys has a Microsoft SQL Server database that contains information about science experiments. The database currently contains information on 5,000 experiments. The information is updated monthly.

You are creating an ASP.NET application for users to find information about specific experiments. The application will run on a server that is connected to the database by a low-bandwidth leases line.
You want to accomplish the following two goals:

• Minimize the number of times the application accesses the database.
• Minimize the time required for each page of the application to load.

What should you do?

A. Initialize the DataSet object in the Page.Load event handler when the IsPostBack property is false.
Store the DataSet object in an XML file on the user's computer.
B. Create a Cache object for the application and add the DataSet object to the cache.
C. Use the view state to maintain the DataSet object during postback events to the server.
D. Create a Cache object for the session and add the DataSet object to the cache.

Answer: B

Explanation:
The database is only updated monthly. Therefore we can store the dataset in the Application Cache with out the need for that extra over head of reading and writing into the XML file client computer for every session (not A).

Question 2.
You are creating an ASP.NET page for ITCertKeys. The page uses string concatenation to gather data from multiple e-mail messages and format the data for display on the page.
You want to ensure that the page displays as quickly as possible.
What should you do?

A. Write code that uses the Append method of the StringBuilder object.
B. Write code that uses the Substring method of the String object.
C. Write code that uses the Concat method of the String object.
D. Write code that uses the plus-sign (+) operator to concatenate the strings.

Answer: A

Explanation:
The StringBuilder.Append method appends the string representation of a specified object to the end of this instance.The StringBuilder class represents a string-like object whose value is a mutable sequence of characters. The value is said to be mutable because it can be modified once it has been created by appending, removing, replacing, or inserting characters.

Option B:

The Substring method is used to select a part of a string, not to concatenate multiple strings.

Option C:
The String.Concat method Concatenates one or more instances of String, or the String representations of the values of one or more instances of Object. However, compared to the Append method of the StringBuilder object, the Concat method create new instances, and is therefore not the preferred method.

Option D:
Not he best solution.

Part 7 Configuring and Securing a Web Application

Question 1.
You create an ASP.NET application that will run on ITCertKeys's Internet Web site. Your application contains 100 Web pages. You want to configure your application so that it will display customized error messages to users when an HTTP code error occurs.

You want to log the error when an ASP.NET exception occurs. You want to accomplish these goals with the minimum amount of development effort.

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

A. Create an Application_Error procedure in the Global.asax file for your application to handle ASP.NET code errors.
B. Create an applicationError section in the Web.config file for your application to handle ASP.NET code errors.
C. Create a CustomErrors event in the Global.asax file for your application to handle HTTP errors.
D. Create a customErrors section in the Web.config file for your application to handle HTTP errors.
E. Add the Page directive to each page in the application to handle ASP.NET code errors.
F. Add the Page directive to each page in the application to handle HTTP errors.

Answer: A & D

Explanation:
Any public event raised by the HttpApplication class is supported using the syntax Application_EventName. For example, a handler for the Error event can be declared protected void Application_Error(Object sender, EventArgs e).

The <customErrors> element, which is used in the Web.config file, provides information about custom error messages for an ASP.NET application.

Option B:
There is no such thing as a applicationError section in the Web.config file.

Option C:
There is no such thing as CustomErros event in the Global.asax file.

Option E, F:
It is not necessary to add a Page Directive to each page.

Question 2.
You create a new ASP.NET application named ITCertKeysSalesReports on your development computer. You add code to the default WebForm1. To test the code's functionality, you copy the entire ITCertKeysSalesReports folder from the C:\inetpub\wwwroot folder on your computer to the C:\inetpub\wwwroot folder on a separate Microsoft Windows 2000 Server computer named ITCertKeys1. ITCertKeys1 hosts several ASP.NET applications

When you use the browser on your computer to open the copy of the application hosted on ITCertKeys1, you receive the following error message:

“It is an error to use a section registered as allowDefinition=”MachineToApplication’ beyond application level.”

You want to correct this error without altering the other Web sites that are hosted on ITCertKeys1. What should you do?

A. Use Internet Information Services (IIS) to create a virtual directory that points to the ITCertKeysSalesReports folder on ITCertKeys1.
B. Remove the following element from the Web.config file in C\inetpub\wwwroot\ITCertKeysSalesReports on ITCertKeys1:
<authentication mode=”Windows” /> C. Remove the following element from the Web.config file in C:\inetpub\wwwroot on ITCertKeys1:
<authentication mode=”Windows” />
D. Move the ITCertKeysSalesReports folder on ITCertKeys1 up one level, so that it is a subfolder of the inetpub folder.

Answer: B

Explanation:
You may have defined a section in your application’s web.config file that is not configurable on our shared web hosting platform. Remove or comment out any configuration sections from your web.config file that are not supported. In this scenario we edit the Web.config file in C\inetpub\wwwroot\ITCertKeysSalesReports on ITCertKeys1.

Option A:
Create a virtual directory would solve the problem.

Option C:
We must edit the application’s web.config file, not the Web.config file in the root directory.

Option D:
Moving the application directory would not solve the problem.

Question 3.
Your ASP.NET application displays sales data on a page. You want to improve performance by holding the page in memory on the server for one hour. You want to ensure that the page is flushed from memory after one hour, and that the page is re-created when the next request for the page is received.

What should you do?

A. Initialize a new instance of the Cache class in the Application.Start event handler.
B. Initialize a new instance of the Timer class in the Page.Load event handler.
C. Set the Duration attribute of the OutputCache directive in the page.
D. In the Web.config file, set the timeout attribute of the sessionState element.

Answer: C

Explanation:
ASP.NET allows you to cache the entire response content for dynamic pages on HTTP 1.1 capable mechanisms, including browsers, proxy servers, and the origin Web server where your application resides. This provides a powerful way for you to increase the performance of your Web applications. Called output caching, it allows subsequent requests for a particular page to be

satisfied from the cache so the code that initially creates the page does not have to be run upon subsequent requests.

To set output-cache expirations for a page declaratively. Include an @ OutputCache directive in the page (.aspx file) that you want to output cache. The directive must include a Duration attribute.

Question 4.
You are creating an ASP.NET application for ITCertKeys. Customers will use the application to file claim forms online.
You plan to deploy the application over multiple servers. You want to save session state information to optimize performance.

What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two)

A. Modify the Web.config file to support StateServer mode.
B. Modify the Web.config file to support SQLServer mode.
C. Modify the Web.config file to support InProc mode.
D. In the Session_Start procedure in the Global.asax file, set the EnableSession property of the WebMethod attribute to **tru**e.
E. In the Session_Start procedure in the Global.asax file, set the Description property of the WebMethod attribute to **sessionStat**e.

Answer: A & D

Explanation:
With StateServer mode session state is using an out-of-process Windows NT Server to store state information. This mode is best used when performance is important but you can't guarantee which server a user will request an application from. With out-of-process mode, you get the performance of reading from memory and the reliability of a separate process that manages the state for all servers. As this scenario requires that we should optimize performance, not reliability, StateServer mode is the preferred solution.
The EnableSession property of the WebMethod attribute enables session state for an XML Web service method. Once enabled, the XML Web service can access the session state collection directly.

Option B:
With SQLServer mode session state is using an out-of-process SQL Server to store state information. The SQL Server mode option is similar to that of the Windows NT Service, except that the information persists to SQL Server rather than being stored in memory. This mode is best used when the reliability of the data is fundamental to the stability of the application, as the database can be clustered for failure scenarios. The performance isn't as fast as out of process, but the tradeoff is the higher level of reliability.

Option C:
With InProc mode session state is in process with an ASP.NET worker process. InProc is the default. However, since we are using multiple servers, we cannot use InProc mode.

Option E:
The Description property of the WebMethod attribute supplies a description for an XML Web service method that will appear on the Service help page.

Question 5.
You create an ASP.NET application named TimeSheet for ITCertKeys's intranet. The application will be used only by employees of your company.

You want the application to recognize the user without forcing the user to enter a name and password. You write the following code to save the user's Microsoft Windows login name in the Session object:
Session.Item("User")=User.Identity.Name
When you run the TimeSheet, the Session.Item("User") variable returns an empty string.
You want to configure Internet Information Services (IIS) and your application to be able to retrieve the user name automatically.
What should you do?

A. Disable Anonymous Access for the application in IIS-
B. Enable Basic authentication for the application in IIS
C. Add the following element to the Web.config file for TimeSheet:
<identity impersonate="True" />
D. Add the following element to the Web.config file for TimeSheet:
<identity impersonate="False" />

Answer: A

Explanation:
We should configure IIS to disallow anonymous access and only use Windows authentication.

Option B:
We want to use Windows authentication, not Basic Authentication.

Option C, D:
Impersonating does not apply in this scenario.

Question 6.
You create an ASP.NET application for a hotel. The application contains a page that displays current weather conditions for the city in which the hotel is located. The application calls an XML Web service every 10 minutes to update the current weather conditions. A new page is then displayed for subsequent requests.

You want the same page to be cached and retrieved from the cache during the time between calls to the XML Web service. You decide to use a Page directive to accomplish this goal.
Which page directive should you use?

A. <%@ Cache Seconds="600" VaryByParam="page" %>
B. <%@ OutputCache Time="600" %>
C. <%@ OutputCache Duration="600" VaryByParam="None" %>
D. <%@ OutputCache Duration="600" %>

Answer: C

Explanation:
To set output-cache expirations for a page declaratively include **an @ OutputCache** directive in the page (.aspx file) that you want to output cache. The directive must include a **Duration** attribute, with a positive numeric value, and a **VaryByParam** attribute. The following @ OutputCache directive sets the page's expiration to 10 minutes (600 seconds).
<%@ OutputCache Duration="600" VaryByParam="None" %>

Option A, B:
We should use the **Duration** attribute, not the **Seconds** or the **Time** attribute, to specify the output-cache expiration for a page.

Option D:

We must use a **VaryByParam** attribute.

Question 7.
You create an ASP.NET application for tracking student examinations at ITCertKeys Boot Camp. You use Microsoft Windows authentication. Students are members of a group named Students, and teachers are members of a group named Teachers.
The root folder for your application is named ITCertKeys. The ITCertKeys folder displays information about pending examinations. The ITCertKeys folder has a subfolder named Grades. Both Students and teachers can access pages in ITCertKeys. Only teachers can access page in Grades.
You create the following entries in the Web.config file in ITCertKeys. (Line numbers are included for reference only)

1 <authentication mode="Windows" />
2 <authorization>
3 <allow roles="Students, Teachers" />
4 <deny users="*" />
5 </authorization>

You crate the following entries in the Web.config file in Grades. (Line numbers are included for reference only.)

1 <authentication mode="Windows" />
2 <authorization>
3 <allow roles="Teachers" />
4 <deny users="*" />
5 </authorization>
When teachers try to access pages in the Grades folder, they receive an error message that reads in part: "An error occurred during the processing of a configuration file required to service this request."
You need to ensure that teachers can access pages in the Grades folder.
What should you do?

A. **Remove line 1 in the Web.config file in Grades.**
B. **Modify line 4 in the Web.config file in Grades as follows:**
<allow users="*" />
C. **Add the following line between line 1 and line 2 in the Web.config file in ITCertKeys:**
<identity impersonate="true" />
D. **Add the following line between line 1 and line 2 in the Web.config file in Grades:**
<identity impersonate="true" />
E. **Add the following line between line 1 and line 2 in the Web.config file in Grades**:
<identity impersonate="false" />

Answer: A

Explanation:
The error messages indicates an incorrect line in the configuration file. The first line with the authentication mode element should be removed.
.NET Framework General Reference, <authorization> Element

Option B:
We only want Teachers to have access to the Grades folder. We cannot allow everyone access.

Option C, D, E:
Impersonate does not apply to this scenario.

Impersonation is when ASP.NET executes code in the context of an authenticated and authorized client.

Question 8.
You create an ASP.NET application named Inventory. This application will be used by customers on the Internet.

During the beta test period, you ensure that the actual ASP.NET error message is displayed whenever an error is encountered. Both developers and beta testers see the actual text of the error message.

You perform beta testing of other applications on the same beta test server during the beta testing period for Inventory. All of the other applications display ASP.NET error messages. After the beta testing period is complete, the beta test server is promoted to a production server. You want all applications to display a single, user-friendly error message.

You want to configure Inventory and the production server to meet these goals. You want to perform this task by using the minimum amount of administrative effort.

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

A. Set the mode parameter of the customErrors element in the Web.config file for Inventory to **"On"**.
B. Remove the customErrors element from the Web.config file for Inventory.
C. Set the mode parameter of the customErrors element in the Inventory.config file to **"On"**.
D. Remove the customErrors element from the Inventory.config file.
E. Set the mode parameter of the customErrors element in the Machine.config file to **"On"**.
F. Remove the customErrors element from the Machine.config file.

Answer: B & E

Explanation:
We should remove the customized error message for the Web.config file for the Inventory application.

We should define a single customized error message for all application on the server. This is done be setting the **mode** attribute to **on** in the Machine.config file.

The <customErrors> Element provides information about custom error messages for an ASP.NET application. The mode attribute specifies whether custom errors are enabled, disabled, or shown only to remote clients. The value of **on** specifies that custom errors are enabled.

Option A:
We want to remove the error messages which are specific to the application.

Option C:
We use the Web.config and the Machine.config files, not Inventory.config, to configure appropriate error messages.

Option D:
A Web.config file is used for the Inventory application. No Inventory.config is used.

Option F:
We want to add one single customized error messages for all application on the server. We should add, not remove, a customErrors element to the Machine.config file.

Question 9.
You create an ASP.NET application for ITCertKeys's purchasing department. A page in the application displays a list of products based on the supplier, the product category, or the price. The URL of the page includes this information as parameters.

You want to store multiple versions of your ASP.NET page in the cache based in the parameter values. You want each version of the page to be cached for 300 seconds.

You need to add code to the page to accomplish this goal.

Which code segment should you use?

A. Response.Cache.SetExpires(DateTime.Now.AddSeconds(300)).
Response.Cache.VaryByParams("?") = true
B. Response.Cache.SetExpires(DateTime.Now.AddSeconds(300))
Response.Cache.VaryByParams("All") = true
C. Response.Cache.SetCacheability(HttpCacheability.Public)
Response.Cache.SetLastModified(DateTime.Parse("00:05:00"))
Response.Cache.VaryByParams("All") = true
D. Response.Cache.SetCacheability(HttpCacheability.Public
Response.Cache.SetExpires(DateTime.Now.AddSeconds(300))
Response.Cache.VaryByParams("*") = true

Answer: D

Explanation:
Cachability corresponds to the Location attribute. The Public value corresponds to any location. We use the SetExpires to set the cache duration. Finally we use the "*" string to specify that all parameter values are cached.

Option A, B:
Cachability has to be set.

Option C:
We should use "*", not "all" when specify VaryByParams.

Question 10.
You are a Web developer for a ITCertKeys bookstore. You create a Web user control named EXBookTopics that is defined in a file named EXBookTopics.ascx. EXBookTopics displays a list of book topics based on an author's profile identification number. The profile identification number is stored in a public property of EXBookTopics named AuthorProfile.

You create an ASP.NET page named AuthorPage.aspx that contains an instance of the EXBookTopics Web user control. AuthorPage.aspx is opened by an HTTP-GET request that has two parameters. The parameters are named publisherID and authorProfileID. The value of authorProfileID is a profile identification number.

You want to enable output caching for the EXBookTopics Web user control. You need to ensure that the cached control is varied only by an author's profile identification number.
What should you do?

A. **Add the following element to the OutputCache directive for AuthorPage.aspx:**
VaryByParam="EXBookTopics.AuthorProfile"
B. **Add the following element to the OutputCache directive for AuthorPage.aspx:**
VaryByControl="EXBookTopics.AuthorProfile"

C. **Add the following element to the OutputCache directive for EXBookTopics.ascx:** VaryByParam=”none”
D. **Add the following element to the OutputCache directive for EXBookTopics.ascx:** VaryByControl=”authorProfileID”

Answer: D

Explanation:
You can vary user control output to the cache in two ways:

1) With the user control name and the parameter. The VaryByParam attribute of the @ OutputCache directive must be used. A) and C) are inadequate since both the control name and the parameter must be specified.

2) With the **VaryByControl** attribute just the parameter should be supplied. This is the case in D), but not in B).