



**70-505(VB)**

**TS: Microsoft .NET Framework 3.5, Windows Forms Application  
Development**

Q&A

DEMO Version

Copyright (c) 2009 Chinatag LLC. All rights reserved.

## **Important Note Please Read Carefully**

For demonstration purpose only, this free version Chinatag study guide contains **10** full length questions selected from our full version products which have more than **200** questions each.

This Study guide has been carefully written and compiled by Chinatag certification experts. It is designed to help you learn the concepts behind the questions rather than be a strict memorization tool. Repeated readings will increase your comprehension.

For promotion purposes, all PDF files are **not** encrypted. Feel free to distribute copies among your friends and let them know Chinatag website.

## **Study Tips**

This product will provide you questions and answers along with detailed explanations carefully compiled and written by our experts. Try to understand the concepts behind the questions instead of cramming the questions. Go through the entire document at least twice so that you make sure that you are not missing anything.

## **Latest Version**

We are constantly reviewing our products. New material is added and old material is revised. Free updates are available for 90 days after the purchase. You should check the products page on the <http://www.chinatag.com> website for an update 3-4 days before the scheduled exam date.

Please tell us what you think of our products. We appreciate both positive and critical comments as your feedback helps us improve future versions. Feedback on specific questions should be send to [feedback@chinatag.com](mailto:feedback@chinatag.com).

Thanks for purchasing our products and look forward to supplying you with all your Certification training needs.

Good studying!

Technical and Support Team  
Chinatag LLC.

## Microsoft 70-505 (VB)

### Question: 1.

You are creating a Windows Forms application by using the .NET Framework 3.5. The application requires a form to display a clock.

You need to create a circular form to display the clock. Which code segment should you use?

- A. `Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None`  
`Dim path As New System.Drawing.Drawing2D.GraphicsPath()`  
`path.AddEllipse(0, 0, Me.Width, Me.Height)`  
`Dim reg As New Region()`  
`Me.Region = reg`
- B. `Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedSingle`  
`Dim path As New System.Drawing.Drawing2D.GraphicsPath()`  
`path.AddEllipse(0, 0, Me.Width, Me.Height)`  
`Dim reg As New Region(path)`  
`Me.Region = reg`
- C. `Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.None`  
`Dim path As New System.Drawing.Drawing2D.GraphicsPath()`  
`path.AddEllipse(0, 0, Me.Width, Me.Height)`  
`Dim reg As New Region(path)`  
`Me.Region = reg`
- D. `Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedSingle`  
`Dim path As New System.Drawing.Drawing2D.GraphicsPath()`  
`path.AddEllipse(0, 0, Me.Width, Me.Height)`  
`Dim reg As New Region()`  
`Me.Region = reg`

### Answer: C

### Question: 2.

You are creating a Windows Forms application by using the .NET Framework 3.5. You create a new form in your application. You add a SplitContainer control named `spcFrame` to the form.

The SplitContainer control has two SplitterPanel controls named `Panel1` and `Panel2`. You are configuring the SplitContainer control to define the layout of the form.

You need to ensure that the following requirements are met:

The initial distance from the left edge of the `spcFrame` splitter is set to 200 pixels.

The size of the `Panel2` SplitterPanel remains unchanged when the form is resized.

Which code segment should you use?

- A. `spcFrame.Panel1MinSize = 200`  
`spcFrame.FixedPanel = FixedPanel.Panel1`
- B. `spcFrame.IsSplitterFixed = True`  
`spcFrame.SplitterWidth = 200`
- C. `spcFrame.SplitterDistance = 200`  
`spcFrame.FixedPanel = FixedPanel.Panel2`

D. spcFrame.Panel2MinSize = 0  
 spcFrame.SplitterIncrement = 200

**Answer: C**

**Question: 3.**

You are creating a Windows Forms application by using the .NET Framework 3.5. You create a new form in the application. You add a ContextMenuStrip control named ctxMenu to the form.

You have a user-defined class named CustomControl.

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

```
01 Dim myControl As New CustomControl()
02
```

You need to ensure that an instance of CustomControl is displayed on the form as a top-level item of the ctxMenu control.

Which code segment should you add at line 02?

- A. Dim host As New ToolStripControlHost(myControl)  
 ctxMenu.Items.Add(host)
- B. Dim panel As New ToolStripPanel()  
 panel.Controls.Add(myControl)  
 ctxMenu.Controls.Add(panel)
- C. Dim panel As New ToolStripContentPanel()  
 panel.Controls.Add(myControl)  
 ctxMenu.Controls.Add(panel)
- D. Dim menuItem As New ToolStripMenuItem()  
 Dim host As New ToolStripControlHost(myControl)  
 menuItem.DropDownItems.Add(host)  
 ctxMenu.Items.Add(menuItem)

**Answer: A**

**Question: 4.**

You are creating a Windows Forms application by using the .NET Framework 3.5.

You create a new form in your application. You add a PrintDocument control named pntDoc to the form.

To support the print functionality, you write the following code segment in the application. (Line numbers are included for reference only.)

```
01 AddHandler pntDoc.BeginPrint, _
  AddressOf PrintDoc_BeginPrint
02 ...
03 Dim canPrint As Boolean = CheckPrintAccessControl()
04 If canPrint = False Then
05
06 End If
07
```

You need to ensure that the following requirements are met:

When the user has no print access, font and file stream initializations are not executed and the print operation is cancelled.

Print operations are logged whether or not the user has print access.

What should you do?

- A. Add the following code segment at line 05.  
 RemoveHandler pntDoc.BeginPrint, AddressOf PrintDoc\_BeginPrint  
 AddHandler pntDoc.BeginPrint, \_  
 function(obj1, args1) args1.Cancel = True  
 Add the following code segment at line 07.  
 AddHandler pntDoc.BeginPrint, AddressOf  
 LogPrintOperation
- B. Add the following code segment at line 05.  
 AddHandler pntDoc.BeginPrint, AddressOf EmptyEventHandler  
 Add the following code segment at line 07.  
 RemoveHandler pntDoc.BeginPrint, AddressOf PrintDoc\_BeginPrint  
 AddHandler pntDoc.BeginPrint, AddressOf  
 LogPrintOperation
- C. Add the following code segment at line 05.  
 RemoveHandler pntDoc.BeginPrint, AddressOf PrintDoc\_BeginPrint  
 RemoveHandler pntDoc.BeginPrint, AddressOf EmptyEventHandler  
 Add the following code segment at line 07.  
 RemoveHandler pntDoc.BeginPrint, AddressOf  
 LogPrintOperation
- D. Add the following code segment at line 05.  
 AddHandler pntDoc.BeginPrint, \_  
 function(obj1, args1) args1.Cancel = True  
 Add the following code segment at line 07.  
 AddHandler pntDoc.BeginPrint, AddressOf PrintDoc\_BeginPrint  
 RemoveHandler pntDoc.BeginPrint, AddressOf  
 LogPrintOperation

**Answer: A**

**Question: 5.**

You are creating a Windows Forms application by using the .NET Framework 3.5. You plan to modify a list of orders within a DataGridView control in the application. You need to ensure that a value is required in the first column of the grid control. Which code segment should you use?

- A. Private Sub dataGridOrders\_CellValidated( \_  
 ByVal sender As Object, \_  
 ByVal e As DataGridViewCellEventArgs) \_  
 Handles dataGridOrders.CellValidated  
 If e.ColumnIndex = 0 Then  
 Dim cellValue = dataGridOrders(e.ColumnIndex, e.RowIndex).Value  
 If cellValue = Nothing \_  
 Or String.IsNullOrEmpty(cellValue.ToString()) Then  
 dataGridOrders.EndEdit()  
 End if  
 End If  
 End Sub
- B. Private Sub dataGridOrders\_Validated( \_  
 ByVal sender As Object, \_  
 ByVal e As EventArgs) \_  
 Handles dataGridOrders.Validated  
 If dataGridOrders.CurrentCell.ColumnIndex = 0 Then  
 Dim cellValue = dataGridOrders.Text  
 If cellValue = Nothing Or \_

```

String.IsNullOrEmpty(cellValue.ToString()) Then
dataGridOrders.EndEdit()
End If
End If
End Sub
C. Private Sub dataGridOrders_Validating( _
ByVal sender As Object, _
ByVal e As CancelEventArgs) _
Handles dataGridOrders.Validating
If dataGridOrders.CurrentCell.ColumnIndex = 0 Then
Dim cellValue = dataGridOrders.Text
If cellValue = Nothing Or _
String.IsNullOrEmpty(cellValue.ToString()) Then
e.Cancel = True
End If
End If
End Sub
D. Private Sub dataGridOrders_CellValidating( _
ByVal sender As Object, _
ByVal e As DataGridViewCellValidatingEventArgs) _
Handles dataGridOrders.CellValidating
If e.ColumnIndex = 0 Then
If e.FormattedValue = Nothing _
Or String.IsNullOrEmpty(e.FormattedValue.ToString()) Then
e.Cancel = True
End If
End If
End Sub

```

**Answer: D**

**Question: 6.**

You are creating a Windows Forms application by using the .NET Framework 3.5. You write the following code segment to bind a list of categories to a drop-down list. (Line numbers are included for reference only.)

```

01 Dim cnnNorthwind As OleDbConnection = _
New OleDbConnection(connectionString)
02 Dim cmdCategory As OleDbCommand = New OleDbCommand( _
"SELECT CategoryID, CategoryName FROM Categories ORDER BY
CategoryName", cnnNorthwind)
03 Dim daCategory As OleDbDataAdapter = _
New OleDbDataAdapter(cmdCategory)
04 Dim dsCategory As DataSet = New DataSet()
05 daCategory.Fill(dsCategory)
06

```

You need to ensure that the drop-down list meets the following requirements:  
 Displays all category names.  
 Uses the category ID as the selected item value.  
 Which code segment should you add at line 06?

A. ddlCategory.DataSource = dsCategory  
 ddlCategory.DisplayMember = "CategoryName"  
 ddlCategory.ValueMember = "CategoryID"

- B. `ddlCategory.DataSource = dsCategory.Tables(0)`  
`ddlCategory.DisplayMember = "CategoryName"`  
`ddlCategory.ValueMember = "CategoryID"`
- C. `ddlCategory.DataBindings.Add("DisplayMember", _`  
`dsCategory, "CategoryName")`  
`ddlCategory.DataBindings.Add("ValueMember", _`  
`dsCategory, "CategoryID")`
- D. `ddlCategory.DataBindings.Add("DisplayMember", _`  
`dsCategory.Tables(0), "CategoryName")`  
`ddlCategory.DataBindings.Add("ValueMember", _`  
`dsCategory.Tables(0), "CategoryID")`

**Answer: B**

**Question: 7.**

You are creating a Windows Forms application by using the .NET Framework 3.5. You write a code segment to connect to a Microsoft Access database and populate a DataSet.

You need to ensure that the application meets the following requirements:

It displays all database exceptions.

It logs all other exceptions by using the `LogExceptionToFile`.

Which code segment should you use?

- A. Try  
`categoryDataAdapter.Fill(dsCategory)`  
`Catch ex As SqlException`  
`MessageBox.Show(ex.Message, "Exception")`  
`LogExceptionToFile(ex.Message)`  
`End Try`
- B. Try  
`categoryDataAdapter.Fill(dsCategory)`  
`Catch ex As SqlException`  
`MessageBox.Show(ex.Message, "Exception")`  
`Catch ex As Exception`  
`LogExceptionToFile(ex.Message)`  
`End Try`
- C. Try  
`categoryDataAdapter.Fill(dsCategory)`  
`Catch ex As OleDbException`  
`MessageBox.Show(ex.Message, "Exception")`  
`Catch ex As Exception`  
`LogExceptionToFile(ex.Message)`  
`End Try`
- D. Try  
`categoryDataAdapter.Fill(dsCategory)`  
`Catch ex As OleDbException`  
`MessageBox.Show(ex.Message, "Exception")`  
`LogExceptionToFile(ex.Message)`  
`End Try`

**Answer: C**

**Question: 8.**

You are creating a Windows Forms application by using the .NET Framework 3.5.

You need to populate a list box control along with category names by using a DataReader control.

Which code segment should you use?

- A. Dim reader As OleDbDataReader  
 Dim cnnNorthwind As OleDbConnection = New \_  
 OleDbConnection(connectionString)  
 cnnNorthwind.Open()  
 Dim cmdCategory As OleDbCommand = New \_  
 OleDbCommand("SELECT \* FROM Categories", cnnNorthwind)  
 reader = cmdCategory.ExecuteReader()  
 while reader.Read()  
 lbCategories.Items.Add(reader("CategoryName"))  
 End While  
 cnnNorthwind.Close()
- B. Dim reader As OleDbDataReader  
 Dim cnnNorthwind As OleDbConnection = New \_  
 OleDbConnection(connectionString)  
 cnnNorthwind.Open()  
 Dim cmdCategory As OleDbCommand = New \_  
 OleDbCommand("SELECT \* FROM Orders", cnnNorthwind)  
 reader = cmdCategory.ExecuteReader()  
 while reader.NextResult()  
 lbCategories.Items.Add(reader("CategoryName"))  
 End While  
 cnnNorthwind.Close()
- C. Dim reader As OleDbDataReader  
 Dim cnnNorthwind As OleDbConnection = New \_  
 OleDbConnection(connectionString)  
 cnnNorthwind.Open()  
 Dim cmdCategory As OleDbCommand = New \_  
 OleDbCommand("SELECT \* FROM Orders", cnnNorthwind)  
 reader = cmdCategory.ExecuteReader()  
 cnnNorthwind.Close()  
 while reader.Read()  
 lbCategories.Items.Add(reader("CategoryName"))  
 End While  
 cnnNorthwind.Close()
- D. Dim reader As OleDbDataReader  
 Using cnnNorthwind As OleDbConnection = New \_  
 OleDbConnection(connectionString)  
 cnnNorthwind.Open()  
 Dim cmdCategory As OleDbCommand = New \_  
 OleDbCommand("SELECT \* FROM Orders", cnnNorthwind)  
 reader = cmdCategory.ExecuteReader()  
 End Using  
 while reader.Read()  
 lbCategories.Items.Add(reader("CategoryName"))  
 End While  
 cnnNorthwind.Close()

**Answer: A**

**Question: 9.**

You are creating a Windows Forms application by using the .NET Framework 3.5. The application stores a list of part numbers in an integer-based array as shown in the following code segment. (Line numbers are included for reference only.)

```

01 Dim parts() As Integer = _
{105, 110, 110, 235, 105, _ 135, 137, 205, 105, 100, 100}
02
03 For Each item In results
04 tbResults.Text += item.ToString() & vbCrLf
05 Next

```

You need to use a LINQ to Objects query to perform the following tasks:  
 Obtain a list of duplicate part numbers.  
 Order the list by part numbers.  
 Provide the part numbers and the total count of part numbers in the results.  
 Which code segment should you insert at line 02?

- A. Dim results = (From n In parts \_  
 Order By n \_  
 Group n By n Into n1 = Group \_  
 Select Key = n, Count = n1.Count()).Distinct()
- B. Dim results = (From n In parts \_  
 Group n By n Into n1 = Group \_  
 Where n1.Count() > 1 \_  
 Order By n1 \_  
 Select Key = n, Count = n1.Count())
- C. Dim results = (From n In parts \_  
 Order By n \_  
 Group n By n Into n1 = Group \_  
 Where n1.Count() > 1 \_  
 Select n1)
- D. Dim results = (From n In parts \_  
 Order By n \_  
 Group n By n Into n1 = Group \_  
 Where n1.Count() > 1 \_  
 Select Key = n, Count = n1.Count())

**Answer: D**

**Question: 10.**

You are creating a Windows Forms application by using the .NET Framework 3.5. You use LINQ expressions to read a list of customers from the following XML file.

```

<customers>
<customer id="135" birthDate="4/1/1968"> Paul Koch </customer>
<customer id="122" birthDate="7/5/1988"> Bob Kelly </customer>
<customer id="044" birthDate="3/24/1990"> Joe Healy </customer>
<customer id="982" birthDate="9/15/1974"> Matt Hink </customer>
<customer id="325" birthDate="1/7/2004"> Tom Perham </customer>
<customer id="134" birthDate="9/23/1946"> Jeff Hay </customer>
<customer id="653" birthDate="5/15/1947"> Kim Shane </customer>
<customer id="235" birthDate="4/24/1979"> Mike Ray </customer>
</customers>

```

You need to obtain a list of names of customers who are 21 years of age or older.

Which code segment should you use?

- A. Dim customers As XDocument = XDocument.Load("Customers.xml")  
Dim results = From c In customers.Descendants(XName.Get("customer")) \_  
Where DateTime.Parse(c.Attribute("birthDate")).AddYears(21) < \_  
DateTime.Now \_  
Select c.Attribute("Name")
- B. Dim customers As XDocument = XDocument.Load("Customers.xml")  
Dim results = From c In customers.Descendants(XName.Get("customer")) \_  
Where DateTime.Parse(c.Attribute("birthDate")).AddYears(21) < \_  
DateTime.Now \_  
Select FullName = c.Value
- C. Dim customers As XDocument = XDocument.Load("Customers.xml")  
Dim results = From c In customers.Descendants(XName.Get("customer")) \_  
Where DateTime.Parse(c.Attribute("birthDate")).AddYears(21) < \_  
DateTime.Now \_  
Select c.Element("customer")
- D. Dim customers As XDocument = XDocument.Load("Customers.xml")  
Dim results = From c In customers.Descendants() \_  
Where DateTime.Parse(c.Attribute("birthDate")).AddYears(21) < \_  
DateTime.Now \_  
Select FullName = c.Value

**Answer: B**