

# ***KillTest***

Higher Quality, Better Service!



## **Q&A**

<http://www.killtest.com>

We offer free update service for one year.

**Exam** : **70-505-VB**

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

**Version** : Demo

1.You are creating a Windows application by using the .NET Framework 3.5. You plan to create a form that might result in a time-consuming operation. You use the QueueUserWorkItem method and a Label control named lblResult. You need to update the users by using the lblResult control when the process has completed the operation. Which code segment should you use?

- A. Private Sub DoWork(ByVal myParameter As Object) 'thread work Invoke(New MethodInvoker (AddressOf ReportProgress))End SubPrivate Sub ReportProgress () Me.lblResult.Text = "Finished Thread"End Sub
- B. Private Sub DoWork (ByVal myParameter As Object) 'thread work Me.lblResult.Text = "Finished Thread"End Sub
- C. Private Sub DoWork (ByVal myParameter As Object)'thread work System.Threading.Monitor.Enter(Me) Me.lblResult.Text = "Finished Thread" System.Threading.Monitor.Exit(Me)End Sub
- D. Private Sub DoWork (ByVal myParameter As Object) 'thread work System.Threading.Monitor.TryEnter(Me) ReportProgress()End SubPrivate Sub ReportProgress () Me.lblResult.Text = "Finished Thread"End Sub

Answer: A

2.You are creating a Windows component by using the .NET Framework 3.5. The component will be used in Microsoft Word 2007 by using a ribbon button. The component uploads large files to a network file share. You find that Word 2007 becomes non-responsive during the upload. You plan to create your own thread to execute the upload. You need to ensure that the application completes the upload efficiently. What should you do.?

- A. Use the AsyncResult.SyncProcessMessage method.
- B. Call the BeginInvoke method, perform the upload, and then call the EndInvoke method.
- C. Retrieve a WaitHandle from an implementation of the IAsyncResult interface before the upload.
- D. Set the IsCompleted property on an implementation of the IAsyncResult interface before the upload.

Answer: B

3.You are creating a Windows Forms application by using the .NET Framework 3.5. The application requires a thread that accepts a single integer parameter. You write the following code segment. (Line numbers are included for reference only.) 01 Dim myThread As Thread = New Thread(New \_ ParameterizedThreadStart(AddressOf DoWork))02 myThread.Start(100)03 You need to declare the method signature of the DoWork method. Which method signature should you use?

- A. Public Sub DoWork()
- B. Public Sub DoWork(ByVal nCounter As Integer)
- C. Public Sub DoWork(ByVal oCounter As Object)
- D. Public Sub DoWork(ByVal oCounter As System.Delegate)

Answer: C

4.You are creating a Windows application by using the .NET Framework 3.5. The Windows application has the print functionality. You create an instance of a BackgroundWorker component named

backgroundWorker1 to process operations that take a long time. You discover that when the application attempts to report the progress, you receive a

System.InvalidOperationException exception when executing the

backgroundWorker1.ReportProgress method. You need to configure the BackgroundWorker component appropriately to prevent the application from generating exceptions. What should you do?

- A. Set the Result property of the DoWorkEventArgs instance to True before you attempt to report the progress.
- B. Set the CancellationPending property of backgroundWorker1 to True before you attempt to report the background process.
- C. Set the WorkerReportsProgress property of backgroundWorker1 to True before you attempt to report the background process.
- D. Report the progress of the background process in the backgroundWorker1\_ProgressChanged event.

Answer: C

5. You are creating a Windows application for graphical image processing by using the .NET Framework 3.5. You create an image processing function and a delegate. You plan to invoke the image processing function by using the delegate. You need to ensure that the calling thread meets the following requirements: It is not blocked when the delegate is running. It is notified when the delegate is complete. What should you do?

- A. Call the Invoke method of the delegate.
- B. Call the BeginInvoke and EndInvoke methods of the delegate in the calling thread.
- C. Call the BeginInvoke method by specifying a callback method to be executed when the delegate is complete. Call the EndInvoke method in the callback method.
- D. Call the BeginInvoke method by specifying a callback method to be executed when the delegate is complete. Call the EndInvoke method of the delegate in the calling thread.

Answer: C