Answers to Review Questions

Advanced Visual Basic 2010, 5th Edition, by Kip Irvine and Tony Gaddis

Last update: March 4, 2011
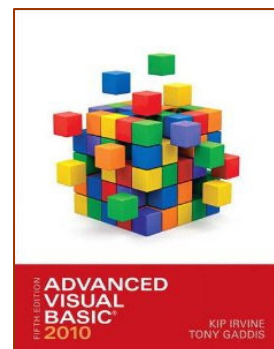
Chapter 1: Classes

True or False

1. False
2. True
3. False
4. False
5. False
6. True
7. False
8. False
9. True
10. True
11. False

Short Answer

1. No, you can write: If isFullTime Then . . .
2. Presentation tier
3. As class methods
4. Middle (business logic) tier
5. At runtime, when multiple instances of the class exist, no more than a single copy of a shared variable exists. In contrast, each instance of the class will contain its own copy of a non-shared variable.
6. Shared property:

    ```
    class Window
        Public Shared Property Color as String
    ```
7. Declaring the MyMethod method:

    ```
    Sub MyMethod(ByRef str As String)
    ```
8. Information hiding (encapsulation)
9. Create public properties that get and set the variable's value.

10. Constructor with optional parameters:

```
Class Hero
   Public Sub New(Optional ByVal pStrength As Integer = 0,
      Optional ByVal pIntelligence As Integer = 0)
      Strength = pStrength
      Intelligence = pIntelligence
   End Sub
```

11. The Set section is omitted.

12. The Clone method.

13. Right-click the project name in the Solution Explorer window, select Add, and select Class.

14. Variable declaration:

```
Private Shared smDefaultColor As String
```

15. Parameterized constructor

16. Opening a text file for input:

```
Dim reader As StreamReader = OpenText("myfile.dat")
```

17. Read from StreamReader

```
Dim line As String = reader.ReadLine()
```

18. EndOfStream property

What Do You Think?

1. You must use a subscript when inserting the items in the target array. But you can use a For Each loop to access the source array:

```
Dim index As Integer = 0
Dim target(Names.Length-1) As String
For Each nam As String in Source
   target(j) = nam
   j += 1
Next
```

2. Because the Clone method returns an array of type Object, which cannot be directly assigned to a specific array type.

3. When the different instances of the class need to access a shared value. The default background color for windows is a good example.

4. Because the middle tier might be accessed by Web applications, Windows forms applications, or other specialized application types.

5. Because the parameter names hide the class-level variables, a technique called *masking*.

6. Because it might not be appropriate to create an instance of the class without assigning it custom values (such as ID number).

7. It stores a message string in the LastError property and returns a value of False. It might be better to throw an exception, which forces the calling method to acknowledge the error.

Algorithm Workbench

1. Property

```
Public Property BirthDate() As DateTime
    Get
        return mBirthDate
    End Get
    Set(value As DataType)
        If value >= #1/1/1900# AndAlso value < Today Then
            mBirthDate = value
        End If
    End Set
End Property
```

2. Investment class constructor

```
Public Sub New(ByVal IdP as String, Optional ByVal amountP As
    Double = 0.0, Optional ByVal tickerP As String = "")
    Id = Idp
    Amount = amountP
    Ticker = tickerP
End Sub

Enum WindowColor
    red
    blue
    green
    yellow
    lightblue
End Enum
```

3. A method with no return value works best for this:

```
Public Sub SetColor( ByVal input WindowColor )
    lblWindowColor.ForeColor = input
End Function
```

4. Splitting a string:

```
scores = inputLine.Split("\"c);
```

Chapter 2: Classes

True or False

1. True
2. True
3. False
4. True
5. False
6. True
7. True
8. False

9. False

10. True

11. False

12. False

13. False

14. True

15. False

16. True

17. False (see Question 3)

18. True

19. True

20. True

Short Answer

1. Unhandled, or uncaught exception

2. ApplicationException

3. FormatException

4. StackTrace

5. TabControl

6. SetError

7. IsLetter

8. ToUpper

9. TextChanged

10. KeyPress

11. StatusStrip

12. ErrorProvider

13. ArgumentOutOfRangeException

14. CheckedIndices

15. SelectionMode

16. ToString

17. DateTimePicker control

18. AddDays

19. ToolStripButton, ToolStripSplitButton, and ToolStripDropDownButton.

20. DisplayStyle

21. SplitContainer control

22. Navigate

What Do You Think?

1. They may wish to enter the field values in a non-sequential order. Having the fields validated one at a time would prevent them from leaving some fields blank with the intention of filling them in later (before saving the form).

2. I prefer applications that prevent me from making mistakes. Telling users about their errors makes them feel incompetent.

3. It can be caught, to permit the user to be notified in a friendly and informative way. This is not always possible, because the underlying runtime system may be corrupted.

4. It can hold a statement that closes the file.

5. Quote from the chapter summary: *A software wizard is an application that leads the user through a series of predescribed steps. In each step, the user may be given choices that influence subsequent steps.*

Algorithm Workbench

1. Sprinkler wizard

   At what time should the watering begin?

   Duration of watering period?

   Which zones are affected by this timer?

   Repeat this timer daily (y/n)?

   Create additional timers (y/n)?

2. Checking the txtZip TextBox:
   ```
   e.Cancel = False
   If txtZip.Text.Length = 5 Then
      For ch As Char in txtZip.Text.ToCharArray
         If Not Char.IsDigit(ch)
            e.Cancel = True
         End If
      Next
   Else
      e.Cancel = True
   End If
   ```

3. Passing an error message:
   ```
   If Not txtName.Contains(" ") Then
      errProvider.SetError(txtName, "txtName must contain two words")
   End If
   ```

4. TimeSpan object -- hours, minutes, seconds:
   ```
   Dim ts As New Timespan(3,10,0)
   ```

5. DateTime example:
   ```
   Dim dt As DateTime = Now
   dt = dt.Add(New Timespan(3,10,0))
   ```