Wyo Visual Basic
Ch. 5 Practice Test **Answers**

The underscore character is used to represent blank spaces throughout this exam.

True/False

1. Double quotes are used in VB as string delimiters. **TRUE**
2. A space at the beginning of a string is called a "leading space." **TRUE**
3. The Trim function is useful to delete all leading and trailing spaces from a string. **TRUE**
4. A loop used within another loop is called a nested loop. **TRUE**
5. VB does not allow If statements to be used within For loops. **FALSE**
6. Len("apple") is 5. **TRUE**
7. The Len function does not count spaces within a string value. **FALSE**
8. Technically a string variable and a string literal are not the same thing. **TRUE**
9. "apple" is less than "apples". **TRUE**
10. Lcase("A") is less than "A". **FALSE**
11. Right("Wyomissing", 2) is less than Left("Wyomissing", 2). **FALSE**
12. A string is part of a larger substring. **FALSE**
13. A For loop is considered to be a determinant loop. **TRUE**

Fill in the Blank - Write your answer in the space provided below the answer. For string function exercises, simplify the function. Use string delimiters where applicable in your answer.

1. Mid("Miami, Florida", 1, 5) = **"Miami"**
2. Left("Rulesville", 4) = **"Rule"**
3. Mid("SMGNFLBA ", 4, 3) = **"NFL"**
4. Right("Reading_PA", 3) = **"_PA"**
5. StrComp(Ucase("Wyo"), "wYo", 1) = **0**
6. Left("CrossCountry", 40) = **"CrossCountry"**
7. Each pass of a loop is called a(n) _____**iteration**_____ .
8. A for loop is considered to be a _____**determinant**_____ loop while a Do loop is called an indeterminant loop.
9. The & symbol is the safest _____**concatenation**_____ operator to use in Visual Basic.
10. The assignment statement **intCount = intCount + 3** is called a(n) ____**counter**_____ statement.
11. There are _____**2**_____ (i.e. how many) parameters in the statement strFirstName = Left("Robert", 3).
12. Technically, "Wyomissing" is an example of a string _____**literal**_____ while strTown is a string variable.

Trace the following code segments. Assume all variables are declared and initialized to zero.

1.

```
strName = "John_Doe"
intLength = Len(strName)

For J = 1 To intLength

     If (Mid(strName, J, 1) = "J") Then
          J = J + 1
     End If

     Beep

Next J
```

strName	intLength	J	beeps
"John_Doe"	**8**	**1**	**7**
		2	
		3	
		4	
		5	
		6	
		7	
		8	
		9	

2.

```
strCity = "Reading"

For J = 6 to 1 Step -2

     If (Mid(strCity, J, 1) = "a") Then
          intCount = intCount + 1
     Else
```

strCity	J	intCount	beeps
"Reading"	**6**	**0**	**3**
	4		
	2		
	0		

```
          Beep
     End If

Next J
```

3.

```
For intNum = -5 To 10
   intSum = intSum + 2
   Beep
Next intNum
```

intNum	intSum	beeps
-5	**0**	**16**
-4	**2**	
.	**4**	
.	**6**	
.	**.**	
11	**.**	
	32	

Write a Visual Basic statement that performs the following task.

1. As efficiently as possible and with maximum readability, write a click event procedure for a command button named **cmdBeep** that uses a Do While loop to make the computer beep 10 times.

```
Private Sub cmdBeep_Click()
      Dim intLoop As Integer

      Do While (intLoop < 10)
            Beep
            intLoop = intLoop + 1
      Loop

End Sub
```

2. Write a click event procedure for a command button named **cmdUpper** that displays a word that is typed in a textbox named **txtInput** in all uppercase letters in the caption of a label named **lblOutput**. You can assume that there is at least one letter in the textbox when the command button is clicked. However, the program should ignore any leading spaces that may be typed in the textbox. For example, if "_hello_" is typed in the textbox, you must display "HELLO" in the label lblOutput.

```
Private Sub cmdUpper_Click()
      lblOutput.Caption = Ucase(Trim(txtInput.Text))
End Sub
```

3. Write a click event procedure for a command button named **cmdCount** that compares the words that are typed into two text boxes named **txtWord1** and **txtWord2**. If the word in txtWord1 is less than the one in txtWord2, display the message "one wins" in a message box. If the word in txtWord2 is less than or equal to the word in txtWord1, display the message "two wins" in a message box. The comparison must be case-insensitive. But you can assume that no blank spaces are typed into either textbox and that at least one letter is typed in each one. For example, if the word "Zeus" was typed into the txtWord1 and if the word "alpha" is typed into txtWord2, then the message box should display the message "two wins" when the command button was clicked.

```
Private Sub cmdCount_Click()

      If (UCase(txtWord1.Text)) < UCase(txtWord2.Text)) Then
            MsgBox "one wins"
      Else
            MsgBox "two wins"
      End If

End Sub
```