Name:_____

1. Before comparing two strings, why would you want to convert both of them to all upper-case letters?

2. String variables or expressions have several methods, functions, or properties to manipulate them. Several are:

| **Sample call** | **Description** |
|---|---|
| `strName = "John Doe"`<br>`strUpperName = strName.ToUpper()` | Returns the string with all capital letters. The original string is unchanged. Here `strUpperName` is assigned "JOHN DOE". |
| `strName = "John Doe"`<br>`strLowerName = strName.ToLower()` | Returns the string with all lower-case letters. The original string is unchanged. Here `strLowerName` is assigned "john doe". |
| `strName = "    John Doe    "`<br>`strTrimmedName = strName.Trim()`<br>`' Also .TrimStart() and .TrimEnd()` | Returns the string without leading or trailing spaces. The original string is unchanged. Here `strTrimmedName` is assigned `"John Doe"` |
| `strName = "John Doe"`<br>`intNumChars = strName.Length` | Returns the **string property** which contains the number of characters in the string. The original string is unchanged. Here `int`NumChars is assigned `8`. |
| `If IsNumeric(txtAge.Text) Then`<br>`  lblMessage = "Valid age"`<br>`Else`<br>`  lblMessage = "Invalid numeric"`<br>`End If` | A (non-method) function that returns whether or not a string parameter represents a valid number. If the user entered, the string "five" for the txtAge.Text, then it would return False. |

Predict the value returned by each of the following:

```
Dim str1 As String = "  This is a string   "
Dim str2 As String = "ABCxyz123"
Dim str3 As String = "987"

a)  str1.ToUpper()

b)  str2.ToLower()

c)  str1.TrimStart()

d)  str1.Trim()

e)  str2.length
    (a property)

f)  IsNumeric(str2)
 (a non-method function)

g)  IsNumeric(str3)
```

Name:_____

3. String variables or expressions have several methods that use the *position* within the string. The *position* of the first character is 0, the second character is at position 1, etc.

```
Position: 01234567
String:  "John Doe"
```

Several of these methods that use the position are:

| **Sample call** | **Description** |
|---|---|
| `strName = "John Doe"strPartial = strName.Substring(3)` | Returns the substring from the specified position to the end of the string. The original string is unchanged. Here `strPartial` is assigned "n Doe". |
| `strName strName = "John Doe" strPartial = strName.Substring(3,6)` | Returns the substring from the first specified position up to but not including the second specified position. The original string is unchanged. Here `strPartial` is assigned "n D". |
| `strName strName = "John Doe" intPos = strName.IndexOf("o")` | Returns the position in the string where the first occurrence of the specified substring (or -1 if not found). The original string is unchanged. Here `intPos` is assigned 1. |
| `strName = "John Doe" intPos = strName.IndexOf("o", 3)` | Returns the position in the string where the first occurrence of the specified substring occurs, but the search starts from the position specified by the second parameter (or -1 if not found). The original string is unchanged. Here `intPos` is assigned 6. |
| `strName = "John Doe" intPos = strName.IndexOf("o", 3, 3)` | Returns the position in the string where the first occurrence of the specified substring occurs, but the search starts from the position specified by the second parameter, and only searches the number of character specified by the third parameter (or -1 if not found). The original string is unchanged. Here `intPos` is assigned -1, since only `"n D"` where searched. |

Predict the value returned by each of the following:

```
Dim str1 As String = "This is a string"
Dim str2 As String = "ABCxyz123"
a)  str2.Substring(2)
b)  str2.Substring(2, 6)
c)  str1.IndexOf("is")
d)  str1.IndexOf("is", 6)
e)  str1.IndexOf("is", 3, 50)
f)  str1.SubString(4).IndexOf("ring")
```

4. What would the MessageBox look like that gets generated by the following statement:

```
MessageBox.Show("Are you sure?", "Confirm", MessageBoxButtons.YesNo, _
                MessageBoxIcon.Question)
```