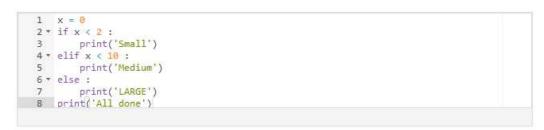1. What do we do to a Python statement that is immediately after an **if** statement to indicate that the statement is to be executed only when the **if** statement is **true**?

   ○ Begin the statement with a curly brace {

   ● Indent the line below the if statement

   ○ Start the statement with a "#" character

   ○ Underline all of the conditional code

2. Which of these operators is **not** a comparison / logical operator?

   ○ !=

   ○ ==

   ○ >=

   ● =

   ○ <

3. What is true about the following code segment:

```
1 ▾ if  x == 5 :
2        print('Is 5')
3        print('Is Still 5')
4        print('Third 5')
```

   ● Depending on the value of **x**, either all three of the print statements will execute or none of the statements will execute

   ○ The string 'Is 5' will always print out regardless of the value for **x**.

   ○ The string 'Is 5' will never print out regardless of the value for **x**.

   ○ Only two of the three print statements will print out if the value of **x** is less than zero.

4. When you have multiple lines in an **if** block, how do you indicate the end of the **if** block?

○ You omit the semicolon ; on the last line of the if block

● You de-indent the next line past the if block to the same level of indent as the original **if** statement

○ You use a curly brace { after the last line of the if block

○ You capitalize the first letter of the line following the end of the if block

5. You look at the following text:

```
1 ▾ if x == 6 :
2      print('Is 6')
3      print('Is Still 6')
4      print('Third 6')
```

It looks perfect but Python is giving you an 'Indentation Error' on the second print statement. What is the most likely reason?

○ Python thinks 'Still' is a mis-spelled word in the string

○ Python has reached its limit on the largest Python program that can be run

○ In order to make humans feel inadequate, Python randomly emits 'Indentation Errors' on perfectly good code - after about an hour the error will just go away without any changes to your program

● You have mixed tabs and spaces in the file

6. What is the Python reserved word that we use in two-way if tests to indicate the block of code that is to be executed if the logical test is false?

● else

○ iterate

○ toggle

○ otherwise

7. What will the following code print out?

```
1   x = 0
2 ▾ if x < 2 :
3       print('Small')
4 ▾ elif x < 10 :
5       print('Medium')
6 ▾ else :
7       print('LARGE')
8   print('All done')
```

🔘 Small

    All done

⚪ Small

    Medium

    LARGE

    All done

⚪ Small

    Medium

8. For the following code,

```
1 ▾ if x < 2 :
2       print('Below 2')
3 ▾ elif x >= 2 :
4       print('Two or more')
5 ▾ else :
6       print('Something else')
```

What value of 'x' will cause 'Something else' to print out?

⚪ x = 2.0

🔘 This code will never print 'Something else' regardless of the value for 'x'

⚪ x = 22

⚪ x = -2.0

9. In the following code (numbers added) - which will be the last line to execute successfully?

```
1  (1)    astr = 'Hello Bob'
2  (2)    istr = int(astr)
3  (3)    print('First', istr)
4  (4)    astr = '123'
5  (5)    istr = int(astr)
6  (6)    print('Second', istr)
```

○ 5

○ 2

◉ 1

○ 6

10. For the following code:

```
1  astr = 'Hello Bob'
2  istr = 0
3 ▾ try:
4      istr = int(astr)
5 ▾ except:
6      istr = -1
```

What will the value be for **istr after this code executes**?

○ The **istr** variable will not have a value

○ 0

○ It will be the 'Not a number' value (i.e. NaN)

◉ -1