**1.What are the two values of the Boolean data type? How do you write them?**

The two Boolean data types are *True* and *False*

These need to be written exactly as True or False with the first letter capitalized and the rest in lower case, if not then we will get a “NameError”. Also we cannot simply abbreviate it as T or F – as this too will result in a “NameError”.

**2. What are the three different types of Boolean operators?**

The Boolean operators are (all small case):

* and
* or
* not

**3. Make a list of each Boolean operator's truth tables (i.e. every possible combination of Boolean values for the operator and what it evaluate ).**

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated with medium confidence

**4. What are the values of the following expressions?**

(5 > 4) and (3 == 5)

Graphical user interface, text, application, chat or text message

Description automatically generated

not (5 > 4)

Graphical user interface, application, Word

Description automatically generated

(5 > 4) or (3 == 5)

Graphical user interface, text, application, chat or text message

Description automatically generated

not ((5 > 4) or (3 == 5))

Graphical user interface, text, application, chat or text message

Description automatically generated

(True and True) and (True == False)

Graphical user interface, text

Description automatically generated with medium confidence

(not False) or (not True)

Graphical user interface, text, application

Description automatically generated

**5. What are the six comparison operators?**

The six comparison operators are:

* Equals: ==
* Not Equals: !=
* Less Than: <
* Greater Than: >
* Less Than or Equal To: <=
* Greater Than or Equal To: >=

**6. How do you tell the difference between the equal to and assignment operators? Describe a condition and when you would use one.**

The “=” is an assignment operator is used to assign the value on the right to the variable on the left. This is used to assign a value to a variable.

The '==' is a comparison operator and checks whether the two given operands are equal or not and returns a Boolean as the outcome.

When I need to store a value in a variable, I can use the assignment operator. For example, if I want to store the value ‘5’, I can create a variable ‘x’ and store the value in it as follows: *x = 5*

Instead, when I want to check if two values are equal then I can use an equal to operator. For example, to check if 2 + 3 is equal to the value of ‘x’ as store in the example above,

I can write: *x == 2 + 3*

**7. Identify the three blocks in this code:**

Text

Description automatically generatedspam = 0

if spam == 10:

print('eggs')

if spam > 5:

print('bacon')

else:

print('ham')

print('spam')

print('spam')

**8. Write code that prints Hello if 1 is stored in spam, prints Howdy if 2 is stored in spam, and prints Greetings! if anything else is stored in spam.**

Text

Description automatically generated

**9.If your programme is stuck in an endless loop, what keys you’ll press?**

To exit out of infinite loops on the command line, press ***CTRL + C***

**10. How can you tell the difference between break and continue?**

A break statement, when used inside the loop, will terminate the loop and exit from the loop. If used inside nested loops, it will break out from the current loop.

A continue statement will stop the execution of the current iteration when used inside a loop, and the control will go back to the start of the loop. Thus it does not terminate the loop but merely terminates the current iteration of the loop. A continue statement does not exit the loop.

**11. In a for loop, what is the difference between range(10), range(0, 10), and range(0, 10, 1)?**

The range function syntax is ***range(start, stop, step)***

In case of *range(10)*

* the range start is implied as 0
* the step increment is implied as +1

in case of *range(0,10)*

* the range start is expressed as ‘0’.
* the step increment continues to be implied as +1
* Since the ‘step’ is a positive integer, the ‘start’ can also be expressed as any number that is less than the ‘stop’ number. If it equal to or greater than the stop number then no iterations of the loop will run.

in case of *range(0,10,1)*

* The range start is expressed as ‘0’ and the range stop is expressed as ‘10’
* Also here the step is expressed as ‘+1’
* The step can be any integer except ‘0’. The relationship between start, stop and step is that variable that is used for iterating the for loop will get assigned the value mentioned in ‘start’ for the first assignment and the value of ‘ += step ‘ in case of subsequent assignments. Then this variable is compared with the ‘stop’ to decide whether the loop goes into the next iteration.

**12. Write a short program that prints the numbers 1 to 10 using a for loop. Then write an equivalent program that prints the numbers 1 to 10 using a while loop**.

Graphical user interface, text, application

Description automatically generated A picture containing text

Description automatically generated

13. If you had a function named bacon() inside a module named spam, how would you call it after importing spam?

**import spam**

**spam.bacon()**