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

**Ans.** True and False

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

**Ans.** AND ,OR AND 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 ).

**Ans.**

**AND**

|  |  |  |
| --- | --- | --- |
| **0** | **0** | **0** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |
| **1** | **1** | **1** |

**OR**

|  |  |  |
| --- | --- | --- |
| **0** | **0** | **0** |
| **0** | **1** | **1** |
| **1** | **0** | **1** |
| **1** | **1** | **1** |

**NOT**

|  |  |
| --- | --- |
| **0** | **1** |
| **1** | **0** |

4. What are the values of the following expressions?

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

**Ans.** 0

not (5 > 4)

**Ans.** 0

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

**Ans**. 1

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

**Ans**. 0

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

**Ans**. 0

(not False) or (not True)

**Ans**. 1

5. What are the six comparison operators?

**Ans.** equal to ==

Not equal to !=

Greater than >

Less than <

Greater than or equal to >=

Less 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.

**Ans**. The equal to operator compares the values and checks if it is same or not.

Eg. A=4,B=5

**A==B ?**

The assignment operator assigns values to the variable.

**A=4**

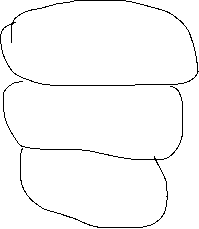
7. Identify the three blocks in this code:

spam = 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.

**Ans:**

If spam==1:

Print(“Hello”)

If spam ==2:

Print(“howdy’)

else:

print(‘greetings)

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

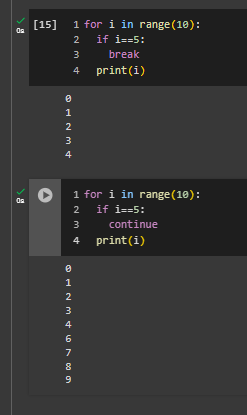
**Ans:** If your Python program is stuck in an endless loop and you're running it in a terminal or command prompt, you can typically stop it by pressing Ctrl+C

If Ctrl+C doesn't work, you might need to resort to more forceful methods like closing the terminal window or killing the Python process from the task manager or terminal commands

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

**Ans:**  When the **break** statement is encountered within a loop, it immediately terminates the loop, regardless of whether the loop's condition evaluates to True or not.

When the **continue** statement is encountered within a loop, it immediately skips the remaining code inside the loop for the current iteration and jumps to the next iteration of the loop



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

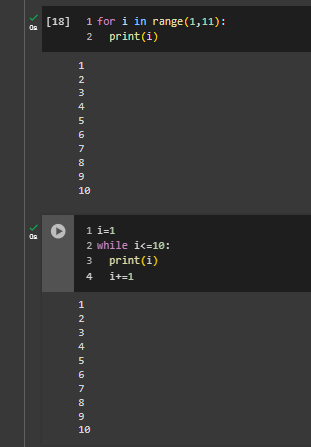
**Ans:**

Range(10) – represents sequence of numbers from 0 to 9.

Range(0,10) – same as above generates sequence of nos from 0 to 9

Range (0,10,1) – genetates nos from 0 to 9 with a step size of 1.

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.



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

**Ans**. After importing the module **spam**, you can call the function **bacon()** by prefixing it with the module name followed by a dot (**.**). Here's how you would do it:

Import spam

Spam.bacon()