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

**Ans:** The two Boolean data types are True and False, in this they evaluate condition and give us answer in Boolean form just like true and false example:

print(45>13) # gives True as condition is true

print(3>4) # gives True as condition is true

print(2>1) # gives false as condition is false

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

- 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).**

- Not operator

|  |  |
| --- | --- |
| Input | output |
| 1 | 0 |
| 0 | 1 |

And Operator

|  |  |  |
| --- | --- | --- |
| Input | Input | output |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |
| 0 | 0 | 0 |

OR Operator

|  |  |  |
| --- | --- | --- |
| Input | Input | output |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |
| 0 | 0 | 0 |

XOR Operator

|  |  |  |
| --- | --- | --- |
| Input | Input | output |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 0 | 0 | 0 |

4. What are the values of the following expressions?

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

False

**not (5 > 4)**

False

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

True

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

False

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

False

**(not False) or (not True)**

True

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

<, >, = =, <= , >= , !=

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

**Ans:** Equal to ( = = ) is a comparison operator, which is used when we need to compare a situation.

while assignment operator is used where you are assigning some value to a variable, and we can use these variables depending on situation.

For example: if I want to compare two values if they are equal or not I’ll use equal to operator

Example 1:

a = 10 #Here assignment operator( = ) is used.

b = 10

a == b #Here comparison operator equal to ( == ) is used to compare if two values are equal or not

---------------------

**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')

**Ans:**

spam = 0

if spam == 10:               #Block 1

    print('eggs')

if spam > 5:                 #Block 2

    print('bacon')

else:                        #Block 3

    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.**

spam = int(input("spam value: "))

if spam == 1:

  print("hello")

elif spam == 2:

  print("Howdy")

else:

  print("Greetings!")

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

**Ans:** CTRL + C

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

Ans: Both break and continue are used to alter the normal flow of loops,

**Break** is used when we want to come out of a loop breaking its usual flow, when break is encountered loop brake immediately.

For ex:

for i in range(0,10):

  if i == 5:

    break

  print(i)

**Output**

0

1

2

3

4

**Continue:** Continue statements skips the remaining part of the program after continue statement and move to the next iteration immediately.

Ex:

for i in range(0,5):

  if i == 3:

    continue #below code will not execute, when condition is true

  print(i)

**Output**

0

1

2

4

In above example the when i == 3, the print statement is not executed, so in output we will not get 3.

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

**Ans:** while using in program above all three would give us same the same sequence of numbers, which are having a range from 0 to 10(exclusive), in which the last number is exclusive

In all three above examples we will get these sequence of numbers 0,1,2,3,4,5,6,7,8,9.

**range(10)** generates sequence of numbers from 0 to 10(exclusive) here step size is 1 by default, in this we do not give any starting point, by default starting point is 0

**range(0,10)** in this we have given a starting point as 0 and goes upto 9, here 10 is exclusive it could be changed to some other starting point as well, it has default step size of 1, it will generate a sequence of numbers 0,1,2,3,4,5,6,7,8,9.

**range(0,10,1)** in this case the starting point and ending points are also similar to range(0,10) but in this we are giving a the step size of 1, we can also change the step size as well

**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.**

**Ans:**

**for loop**

for i in range(1,11):

  print(i)

**while loop**

i =1

while i <11:

  print(i)

  i+=1

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

**Ans:**

Import spam()

Spam.bacon()