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

The two types of Boolean data types are

* True, which is written as **T**
* False, Which is written as **F**

In both cases, the first letter is capitalized and rest of the word is in lowercase

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

The three types of Boolean operators are

* 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 ).

|  |  |  |  |
| --- | --- | --- | --- |
| Truth Table of AND and OR | | | |
| Condition | **Condition** | **And** | **OR** |
| True | True | True | True |
| True | False | False | True |
| False | True | False | True |
| False | False | False | False |

|  |  |
| --- | --- |
| Condition | NOT |
| True | False |
| False | True |

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?

The six comparison operators are

1. Less Than Operator (<) – E.g., 5 < 10
2. Greater Than Operator (>) - E.g. 10 > 5
3. Less Than or Equal to Operator (<=) – E.g. 5 <= 10
4. Greater Than or Equal to Operator ( >= ) – E.g. 10 >= 5
5. Equal to Operator (==) – E.g. 5 == 5
6. Not Equal to Operator ( != ) – E.g. 10 != 5

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

The Equal to Operator (==), is used to compare values of two and evaluates to a Boolean.

* Ex – 5 == 5

The Assignment Operator (=) is used to store values in a Variable

* Ex – a = 5

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')

Solution -

spam = 0

if spam == 10:

print('eggs') # block 1

if spam > 5: # still block 1

print('bacon') # still block 1, indent increased, block 2 inside block 1

else: # still block 1, indent decreased, block 2 ended in line above

print('ham') # still block 1, indent increased, block 3 inside block 1

print('spam') # still block 1, indent decreased, block 3 ended in line above

print('spam') # indent decreased, block 1 ended in line above

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.

Solution –

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 - Press CTRL-C to stop a program stuck in an infinite loop.

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

Ans –

* **The break statement** will move the execution outside and just after a loop.
* **The continue statement** will move the execution to the start of the loop.

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

Ans –

Generally, they all perform the same. However,

* The range (10) call ranges from 0 up to (but not including) 10.
  + Output – [0,1,2,3,4,5,6,7,8,9]
* range (0, 10) explicitly tells the loop to start at 0.
  + Output – [0,1,2,3,4,5,6,7,8,9]
* range (0, 10, 1) explicitly tells the loop to increase the variable by 1 on each iteration.
  + Output – [0,1,2,3,4,5,6,7,8,9]

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.

Solution –

* Using for loop

for i in range(1,11):

print(i)

* Using while loop
  + i = 1

While i <= 10:

print (i)

i = i+1

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

Answer –

We would call the function as **spam.bacon()**