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

**Ans:** Python Boolean data type is one of the built-in data types provided by Python, which represents one of the two values i.e., True or False. Generally it takes the value ‘1’ for ‘True’ and the value ‘0’ for ‘False’.

**Eg:** bool(1) = True

bool(0) = False

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

**Ans:** The three different types of Boolean operators are **‘and’, ‘or’, ‘not’**. While **‘and’** and **‘or’** operator needs two operands, which may evaluate to True or False, **‘not’** operator needs one operand evaluating to True or False.

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 TRUTH TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **X** | **and** | **Y** | **Returns** |
| True | and | True | True |
| True | and | False | False |
| False | and | True | False |
| False | And | False | False |

**OR TRUTH TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **X** | **or** | **Y** | **Returns** |
| True | **or** | True | True |
| True | **or** | False | True |
| False | **or** | True | True |
| False | **or** | False | False |

**‘NOT’ TRUTH TABLE**

|  |  |  |
| --- | --- | --- |
| **Not** | **X** | **Returns** |
| **Not** | True | False |
| **Not** | False | True |

**== TRUTH TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **X** | **==** | **Y** | **Returns** |
| True | **==** | True | True |
| True | **==** | False | False |
| False | **==** | True | False |
| 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?

**Ans:**

**COMPARISION OPERATORS**

|  |  |  |
| --- | --- | --- |
| **Operator** | **Name** | **Example** |
| == | Equal | X == Y |
| != | Not Equal | X! = Y |
| > | Greater than | X > Y |
| < | Less than | X < Y |
| >= | Greater than equal to | X >= Y |
| <= | Less than equal to | X <= Y |

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 ‘=’ is an assignment operator is used to assign the value on the right to the variable on the left. The ‘==’ operator checks whether the two given operands are equal or not. If so, it returns true.

**Condition for using Assignment operator ( ‘=’ ) :**

a = 7

b = 2

c = a + b ; **c = 9**

**Condition for using Equal to operator ( ‘= =’ ) :**

X = 3

Y = 3

**X = = Y ( True )**

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: This answer is attached to jupyter notebook file.**

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: This answer is attached to jupyter notebook file.**

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

**Ans:** In order to step out from the infinite loop, we need press **Ctrl + C.**

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

**Ans:** The ‘break’ statement stops the loop in which the statement is placed. A ‘Continue’ statement skips a single iteration in a loop.

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

**Ans:** There is no difference between range(10), range(0,10), and (0,10,1). All the three indicates the same. (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.

**Ans: This answer is attached to jupyter notebook file.**

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

**Ans:** This function can be called with **spam.bacon()**