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

**Ans:** Boolean data type returns **True** when the argument x is true, **False** otherwise.  
We write them as **True** and **False** and these are the only two instances of the class bool.

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

**Ans:** 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 ).

## **Ans:** Truth Table for Python OR Operator

|  |  |  |
| --- | --- | --- |
| A | B | Result (A or B) |
| True | True | True |
| True | False | True |
| False | True | True |
| False | False | False |

## Truth Table for Python AND Operator

|  |  |  |
| --- | --- | --- |
| A | B | Result (A & B) |
| True | True | True |
| True | False | False |
| False | True | False |
| False | False | False |

## Truth Table for Python NOT Operator

|  |  |  |
| --- | --- | --- |
| A | B | NOT (A & B) NOT (A or B) |
| True | True | False True |
| True | False | True False |
| False | True | True False |
| False | False | True False |

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:**

|  |  |
| --- | --- |
| **Operator** | **Name** |
| == | Equal |
| != | Not equal |
| > | Greater than |
| >= | Greater than or equal to |
| < | Less than |
| <= | 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 “=” 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. Otherwise, it returns false.

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:** It prints

ham  
spam  
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.

**Code:**  spam = int(input())

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 (it will interrupt the kernel)

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

|  |  |  |
| --- | --- | --- |
| **Basis for comparison** | **break** | **continue** |
| Task | It eliminates the execution of remaining iteration of loop | It will terminate only the current iteration of loop. |
| Control after break/continue | ‘break’ will resume control of program to the end of loop enclosing that ‘break’. | The ‘continue’ will resume the control of the program to next iteration of that loop enclosing ‘continue' |
| causes | It early terminates the loop. | It causes the early execution of the next iteration. |
| continuation | The ‘break ‘stop the continuation of the loop. | The ‘continue’ does not stop the continuation of loop and it stops the current. |
| Other | It used with the ‘switch’, ‘label’ | Cannot be executed with switch and the labels. |

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

**Ans:** range(10) - it prints 0 to 9.

range(0,10) - it prints 0 to 9.

range(0,10,1) - it prints 0 to 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:** Printing numbers from 1 to 10 using **For** loop.

**for x in range(1,11):**

**print(x)**

Printing numbers from 1 to 10 using **While** loop.

**x = 1**

**while x < 11:**

**print(x)**

**x += 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**  
**a = spam.bacon()**