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

Ans- The Python Boolean type has only two possible values:

1. True.
2. False

A Boolean data type is declared with the bool keyword and can only take the values true or false . When the value is returned, true = 1 and false = 0

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

Ans- There are three basic Boolean search commands: AND, OR and NOT.

* AND searches find all of the search terms.  For example, searching on dengue AND malaria AND zika  returns only results that contain all three search terms.  Very limited results.
* OR searches find one term or the other.  Searching on dengue OR malaria OR zika returns all items that contain any of the three search terms.  Returns a large number of results.
* NOT eliminates items that contain the specified term.  Searching on malaria NOT zika returns items that are about malaria, but will specifically NOT return items that contain the word zika.  This is a way to fine-tune results. Note:  sometimes AND NOT is used;  serves the same function as 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- True and True is True.

True and False is False.

False and True is False.

False and False is False.

True or True is True.

True or False is True.

False or True is True.

False or False is False.

not True is False.

not False is True.

# True is 1 and False is 0

Truth Table for AND  
A B output  
0 0 0  
0 1 0  
1 0 0  
1 1 1  
Truth Table for OR  
A B output  
0 0 0  
0 1 1  
1 0 1  
1 1 1  
Truth Table for NOT  
A output  
0 1  
1 0

4. What are the values of the following expressions?

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

not (5 > 4)

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

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

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

(not False) or (not True)

Ans- False

False

True

False

False

True

5. What are the six comparison operators?

Ans-==, !=, <, >, <=, and >=

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

Ans- == is the equal to operator that compares two values and evaluates to a Boolean, while = is the assignment operator that stores a value in a variable.

*#eg.*

*# Equal To Operator*

**if**(2**==**3):

print("True")

**else**:

print("False")

*#Assignment operator*

c**=**1 *#here we have used assignment operator(=) to assign value of c which is 1*

print("c =",c)

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')

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.

Ans-spam **=** int(input("Input a no."))

**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-If program is stuck in endless loop we will press ctrl+c.

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

Ans-*# use of break*

**for** i **in** range(10):

**if**(i**==**7):

**break**

print(i)

print('Breaked')

*#use of continue*

**for** i **in** range(10):

**if**(i**==**7):

**continue**

print(i)

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- The range(10) call ranges from 0 up to (but not including) 10, range(0, 10) tells the loop to start at 0, and range(0, 10, 1) tells the loop to increase the variable by 1 on each iteration.

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- *#Use of For Loop*

print("For Loop")

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

print(i)

*#Use of While Loop*

print("While Loop")

a **=**1

**while** a **<=** 10:

print(a)

a**+=**1

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().