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

-The two values of the Boolean data types are true and false. They are written as True and False. (Case Sensitive)

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

* **And operator** (‘**and’**)– It return True value when both conditions satisfy otherwise it returns False value.
* **OR operator** (‘**or’**)– it returns True value when any of the one condition satisfy otherwise it returns False value.
* **Not operator (‘not’)** – it gives the exact opposite value of the operand.

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 evaluates).

* **and operator: -**
* if A is True & B is False then A and B will return False.
* if A is False & B is True then A and B will return False.
* if A is True & B is True then A and B will return True
* if A is False & B is False then A and B will return False.
* **or operator: -**
* if A is True & B is False then A or B will return True.
* if A is False & B is True then A or B will return True.
* if A is True & B is True then A or B will return True
* if A is False & B is False then A or B will return False.
* **not operator: -**
* if A is True, then not A will return False.
* if A is False, then not A will return 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?

* **Equal to operator: A==B**
* **Not equal to operator: A!=B**
* **Greater than operator: A>B**
* **Lesser than operator: A<B**
* **Greater than equal to operator: A>=B**
* **Lesser than equal to operator: B<=B**

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

* **The Difference between assignment operator “=” & equal to operator “==” is**
* **In assignment operator e.g. Name = “Aniket” (Aniket is assigned to variable Name)**
* **In equal to operator e.g. Name == “Aditya” (It evaluates & check if condition is true or not).**

**Refer following if and else condition statements where assignment operator & equal to operator is used.**

**Name = "Aniket"**

**if Name == "Aniket”:**

**print ("Hello", Name)**

**else:**

**print ("User denied")**

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')

**spam = 0**

**if spam == 10:**

**print('eggs') # Indentation**

**elif spam > 5: #elif**

**print('bacon') #indentation**

**else:**

**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 ("Enter the 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?

**Ctrl** + **C**

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

* Break is used to exit a loop with immediate effect without going.
* Continue is used to skip the current iteration & moves to next iteration.

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

* range (10): **1**
* range (0, 10): **2**
* range (0, 10, 1): **3**
* difference is the no of elements /value in the list

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.

* For Loop:

for n in range (1,11):

print (n)

* While loop :

n = 1

while n<=10 :

print (n)

n+=1

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

import spam

spam.bacon()