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

**ANS:-**

1. The two values of the **Boolean data type** are true and false. They are written as such: True and False.

2. The three **Boolean operators** are AND, OR and NOT.

3. Truth tables of each Boolean operator is:

**AND**

A | B | A AND B

T | T | T

T | F | F

F | T | F

F | F | F

**OR**

A | B | A OR B

T | T | T

T | F | T

F | T | T

F | F | F

**NOT**

A | NOT A

T | F

F | T

4. The following **expressions evaluate** to:

1. (True AND True) OR False = True

2. True OR (False AND True) = True

3. NOT (True AND False) = True

**Boolean operators** are logical operators that are used to compare values and return a true or false value. They are used in programming, math, and logical reasoning. The three main Boolean operators are AND, OR, and NOT.

The AND **operator returns** true if both values are true, OR returns true if either value is true, and NOT returns true if the value is false. Boolean operators are very useful for controlling the flow of a program, making decisions, and performing **calculations**.

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

**ANS:-**

**AND**—requires both terms to be in each item returned. If one term is contained in the document and the other is not, the item is not included in the resulting list. (Narrows the search)

**Example:** A search on stock market AND trading includes results contains: stock market trading; trading on the stock market; and trading on the late afternoon stock market

**OR**—either term (or both) will be in the returned document. (Broadens the search)

**Example:** A search on ecology OR pollution includes results contains: documents containing the world ecology (but not pollution) and other documents containing the word pollution (but not ecology) as well as documents with ecology and pollution in either order or number of uses.

**NOT or AND NOT** ( dependent upon the coding of the database's search engine)—the first term is searched, then any records containing the term after the operators are subtracted from the results. (Be careful with use as the attempt to narrow the search may be too exclusive and eliminate good records). If you need to search the word not, that can usually be done by placing double quotes (<< >>) around it.

**Example:** A search on Mexico AND NOT city includes results contains: New Mexico; the nation of Mexico; US-Mexico trade; but does not return Mexico City or This city's trade relationships with Mexico.

**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?**

**ANS:-**

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

**not (5 > 4) = False**

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

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

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

**(not False) or (not True) = 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)

**False**

**c = 1**

**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:-**

**spam = 0**

**if spam == 10:**

**print('eggs') #Block 1**

**if spam > 5:**

**print('bacon') #Block 2**

**else:**

**print('ham') #Block 3**

**print('spam')**

**print('spam')**

Output:-

**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!")

**Output:-**

Input a no.2

Howdy

**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)

**Output:-**

0

1

2

3

4

5

6

Breaked

0

1

2

3

4

5

6

8

9

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:-**

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

print(i)

print("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx")

**for** i **in** range(0,10):

print(i)

print("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx")

**for** i **in** range(0,10,1):

print(i)

**Output :-**

0

1

2

3

4

5

6

7

8

9

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

0

1

2

3

4

5

6

7

8

9

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

0

1

2

3

4

5

6

7

8

9

From the above output we can conclude that they all do the same thing. 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

**Output :-**

For Loop

1

2

3

4

5

6

7

8

9

10

While Loop

1

2

3

4

5

6

7

8

9

10

**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().