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

**Answer** – The two values of the Boolean data type are True and False. They are written as *“True”* and *“False”*

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

**Answer** – Three different types of boolean operators are as under

* AND operator. It results in *True* if both the operands are *True*.
* OR operator. It results in *True* if one of the operands is *True*.
* Not operator. The NOT operator flips the value of the operand

**Q 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 ).

**Answer** –

* AND operator

| Operand 1 | Operand 2 | Result |
| --- | --- | --- |
| TRUE | TRUE | TRUE |
| TRUE | FALSE | FALSE |
| FALSE | TRUE | FALSE |
| FALSE | FALSE | FALSE |

* OR operator

| Operand 1 | Operand 2 | Result |
| --- | --- | --- |
| TRUE | TRUE | TRUE |
| TRUE | FALSE | TRUE |
| FALSE | TRUE | TRUE |
| FALSE | FALSE | FALSE |

* NOT operator

| Operand | Result |
| --- | --- |
| FALSE | TRUE |
| TRUE | FALSE |

**Q 4.** What are the values of the following expressions?

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

**Answer -** TRUE

not (5 > 4)

**Answer -** FALSE

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

**Answer -** TRUE

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

**Answer -** FALSE

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

**Answer -** FALSE

(not False) or (not True)

**Answer -** TRUE

**Q 5.** What are the six comparison operators?

**Answer** – The comparison operators are as under

1. Equal to ( == )

This compares the 2 operands. It results in true if both are equal else returns false.

1. Not equal to ( != )

This results in true if the 2 operands are unequal

1. Less than ( < )

This results in true if LHS operand is less than RHS operand

1. Greater than ( > )

This results in true if LHS operand is greater than RHS operand

1. Less than equal to ( <= )

This results in true if LHS operand is less than or equal to RHS operand

1. Greater than equal to ( >= )

This results in true if LHS operand is greater than or equal to RHS operand

**Q 6.** How do you tell the difference between the equal to and assignment operators?Describe a

condition and when you would use one.

**Answer** –

* Equal to operator uses 2 equal signs ( == ). It its used to compare values of the operands.  
  Example. Following we compare 2 variables x and y.   
  x= 1  
  y=2  
  Result = (x==y)  
  print(Result)  
  **Output** - False
* Assignment operator is used to assign the value of the RHS to variable on LHS.  
  Example. Below value 5 is assigned to variable x  
  X=5

**Q 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’)

**Answer** –

1. Variable named spam is assigned with value 0
2. If statement checks if the value of spam is equal to 10
3. If statement checks if the value of spam is greater than 5. If the value is greater than 5 then *print(‘bacon’)* statement is executed. If not, then *print('ham') print('spam') print('spam')* statements are executed

**Q 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.

**Answer** –

spam = 1

if spam == 1:

print('Hello')

elif spam == 2:

print('Howdy')

else:

print('Greetings!')

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

**Answer** – we can press CTRL+C keys to stop execution.

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

**Answer** –

*Break* helps us to control the loop execution. It is used to stop the loop execution and come out of the loop. *Continue* is also used to control the loop, its used to skip to the next iteration of the loop.

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

**Answer** –

range(10) - This creates a sequence of numbers starting from 0 upto 10. The value given in brackets (10 in this case) is excluded. In this case the sequence of numbers is created from 0 to 9.  
0, 1, 2, 3, 4, 5, 6, 7, 8, 9 is the sequence generated

range(0, 10) - This creates a sequence of numbers starting from 0 upto 10. The second value given in brackets (10 in this case) is excluded and the first value ( 0 in this case) is included in the sequence.  
0, 1, 2, 3, 4, 5, 6, 7, 8, 9 is the sequence generated

range(0, 10,1) - This creates a sequence of numbers starting from 0 upto 10, increment by 1.   
0, 1, 2, 3, 4, 5, 6, 7, 8, 9 is the sequence generated

**Q 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.

**Answer** –

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

i=1  
 while i <=10:  
 print(i)  
 i=i+1

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

**Answer** –

import spam   
 spam.bacon()