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

Boolean data type has two logical value that can be either true or false and this values are written as  
**True**

**False**

Note: its also a keyword which are case sensitive

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

The three different types of Boolean operators are and, or , & not  
  
1. Logical ‘and’ : The logical and operator return True if both operands are condition are True. Otherwise, False  
2. Logical ‘or’ : The logical or operator return True if at least one of operands or condition is True.  
it return False only if both operands are False.

3. Logical ‘not’ : The logical not operator is a unary operator that negates the value of a Boolean expression. It returns True if the operand or condition is false and False if the operand or condition is true.

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 ).

Logical AND (and):

Operand 1 Operand 2 Result

True True True

True False False

False True False

False False False

Logical OR (or):

Operand 1 Operand 2 Result

True True True

True False True

False True True

False False False

Logical NOT (not):

Operand Result

True False

False 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?

1 Equal to (**==**): Compares if two values are equal and returns True if they are, otherwise False.

2 Not equal to (**!=**): Checks if two values are not equal and returns True if they are not, False otherwise.

3 Greater than (**>**): Determines if the left operand is greater than the right operand. It returns True if it is, False otherwise

4 Less than (<): Determines if the left operand is less than the right operand. It returns True if it is, False otherwise. For example: 3 < 5 evaluates to True.

5 Greater than or equal to (>=): Checks if the left operand is greater than or equal to the right operand. It returns True if it is, False otherwise.

6 Less than or equal to (<=): Checks if the left operand is less than or equal to the right operand. It returns True if it is, False otherwise.

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

Equal to operator ‘==’ is used to compare if two values are equal or not.

Assignment (**=**) operator is used to assign values to variables, storing a value in a variable for future use or manipulation.

For example : x=5 and y = 10, now I want check if x is equals to y or not   
If x==y:  
Print(x and y are equal)

else:  
 Print(X and Y are not equal)   
and we have assigned a value to x and y where assignment operator is used.

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')

Block 1 :   
if spam == 10:

print('eggs')

In this block statement we have condition if spam == 10 then print eggs otherwise nothing

Block 2:   
if spam > 5:

print('bacon')

else:

print('ham')

In this block statement we have another if else statement

Block 3:  
print('spam')

print('spam')

This block consists of two consecutive print statements.

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.

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?

Depends on system : Ctrl+C or ctrl+break

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

break is used to exit a loop entirely, terminating the loop's execution and moving to the next statement after the loop.

continue is used to skip the rest of the current iteration in a loop and move to the next iteration, without executing the remaining statements in the loop for that specific iteration.

The difference is break is used to exit the loop completely, while continue is used to skip the current iteration and proceed to the next iteration within the loop.

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

1. range(10): This form of the range function generates a sequence of numbers starting from 0 (by default) up to, but not including, the specified end value (10 in this case). It increments by 1 at each step. In other words, it generates the numbers 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9.

2. range(0, 10): This form of the range function specifies both the start and end values of the sequence explicitly. It generates numbers starting from the specified start value (0 in this case) up to, but not including, the specified end value (10 in this case).

3. range(0, 10, 1): This form of the range function includes both the start, end, and step values. It generates numbers starting from the specified start value (0 in this case) up to, but not including, the specified end value (10 in this case).

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

i=1

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