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

Boolean data types represent only two values: **true** and **false**. These values can be used in programming to make decisions, evaluate conditions, and control the flow of a program.

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

Boolean operators are fundamental in programming and logic, and they help create more complex conditions. The three primary types of Boolean operators are AND, OR and 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 ).

### AND Operator (&& or and)

| **Operand 1** | **Operand 2** | **Result** |
| --- | --- | --- |
| true | true | true |
| true | false | false |
| false | true | false |
| false | false | false |

### OR Operator (|| or or)

| **Operand 1** | **Operand 2** | **Result** |
| --- | --- | --- |
| true | true | true |
| true | false | true |
| false | true | true |
| false | false | false |

### NOT Operator (! or not)

| **Operand** | **Result** |
| --- | --- |
| true | false |
| false | true |

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)

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

* 5 > 4 is true
* 3 == 5 is false
* true and false is false

So, the value is false.

**NOT (5 > 4)**

* 5 > 4 is true
* not true is false

So, the value is false.

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

* 5 > 4 is true
* 3 == 5 is false
* true or false is true

So, the value is true.

**NOT ((5 > 4) or (3 == 5))**

* 5 > 4 is true
* 3 == 5 is false
* true or false is true
* NOT true is false

So, the value is false.

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

* True and True is true
* True == False is false
* true and false is false

So, the value is false.

**(NOT False) or (NOT True)**

* NOT False is true
* NOT True is false
* True or false is true

So, the value is true.

5. What are the six comparison operators?

(1) Equal to (2) Not equal to (3) Greater than (4) Less than (5) Less than or equal to (6) Greater than or equal to

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 (==)

The == operator is a comparison operator used to check if two values are equal. It returns a Boolean value: true if the values are equal and false if they are not.

### Assignment (=)

The = operator is used to assign a value to a variable. It does not compare values but rather sets the value of a variable to the right-hand side value.

# User's age

age = 15

# Check if the user is a teenager (ages 13-19)

is\_teenager = (age >= 13) and (age <= 19) # Using comparison operators (>=, <=, and ==)

if is\_teenager == True:

print("The user is a teenager.")

else:

print("The user is not a teenager.")

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')

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.

Assignment block

First if block

Second if-else block

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

If your program is stuck in an endless loop, you can interrupt its execution by pressing **Ctrl + C** on your keyboard. This keyboard shortcut sends an interrupt signal to the running program, causing it to stop.

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

Use break to exit the loop.

Use continue to skip the current iteration and move to the next one.

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

In summary, while range(10), range(0, 10), and range(0, 10, 1) all generate the same sequence of numbers from 0 to 9, their syntax differs in how explicitly they define the start, end, and step of the sequence.

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 to print numbers 1 to 10

for i in range(1, 11):

print(i)

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

**Import the module**: import spam

**Call the function**: spam.bacon()