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

The Boolean data type is used to represent logical values, typically denoting either true or false.

True: This represents a logical true value.

False: This represents a logical false value.

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

A. Logical AND (and): The ‘AND’ operator returns True if both operands are True, and False otherwise.

B. Logical OR (or): The ‘OR’ operator returns True if at least one of the operands is True, and False if both operands are False.

C. Logical NOT (not): The ‘NOT’ operator is a unary operator that negates the value of the operand. If the operand is True, not returns False, and if the operand is False, not returns 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 ).**

values for the operator and what it evaluate ).

**Here are the truth tables for each Boolean operator:**

**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

These truth tables show the possible combinations of Boolean values for the operands of each

operator and the resulting Boolean value after applying the operator.

**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?**

**Operator Name Example**

**== Equal x == y**

**!= Not equal x != y**

**> Greater than x > y**

**< Less than x < y**

**>= Greater than or equal to x >= y**

**<= Less than or equal to x <= y**

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

The equal to operator (==) is used to compare whether two values are equal, whereas the assignment operator (=) is used to assign a value to a variable.

x = 5

if x == 5:

print("x is equal to 5")

and ,

x = 10

**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')

Here,

1. spam has value 0

2. if condition checks smap is equal to 10 then print ‘eggs’ otherwise leave

3. Another if condition checks spam is greater then 5 if yes print ‘bacon’ otherwise ‘ham’ , ‘spam’ , ‘spam’ so it will print last print command ,

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.**

spam = int(input("Enter a number: "))

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?**

**If program is stuck in an endless loop and to interrupt its execution, you can press the "Ctrl" and "C" keys simultaneously.**

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

The **break** statement is used to immediately exit the innermost loop (where it is encountered).

The **continue** statement is used to skip the rest of the code block in the current iteration and move to the next iteration of the loop.

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

These are the same codes start from 0 and end at 9 step by 1.

0,1,2,3,4,5,6,7,8,9

**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=0

while i<=9:

i=i+1

print(i)

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