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

True

False

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

The three different types of Boolean operators are:

and – this operator results in true when all the condition is satisfied, if any one of the condition is not satisfied it gives false

or – this operator results in true when any one of the condition is satisfied if all the condition do not satisfy it gives false value.

not – this operator just inverts the result true becomes false and vice versa

eg

a=5

b=10

print(a>4 and b>5)

o/p True

print(a>4 or b<5)

o/p True

print(not(a>4 and b>5))

o/p False

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

0 0 🡪 0

0 1 🡪 0

1 0 🡪 0

1 1 🡪 1

or

0 0 🡪 0

0 1 🡪 1

1 0 🡪 1

1 1 🡪 1

not

0 🡪 1

1 🡪 0

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?

== 🡪 Equal to

!= 🡪 Not Equal to

> = Greater than

< = less than

<= 🡪 Less than equal to

>= 🡪 Greater than 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 equal to operator is used for comparison and checks whether two values are equal or not.

It returns a Boolean value based on whether the comparison is true or false.

Eg.

A=10

if A == 10: # checking if the value of A is equal to the value of 10

print(“Both are same”)

Assignment (=)

The assignment operator is used to assign a value to a variable.

The value on the right side is stored in the variable on the left side.

Eg.

A=10 # assigning the values 10 and 20 to A and B variables respectively

B=20

Print(A)

Print(B)

7. Identify the three blocks in this code:

1st block

spam = 0

if spam == 10:

print('eggs')

2nd block

if spam > 5:

print('bacon')

else:

print('ham')

3rd block

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.

spam = int(input("Enter any value for 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?

An infinite loop can be terminated by using the keys Ctrl + C

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

The difference between break and continue are in following ways:

When break is found it terminates the loop and jumps to the first statement after the loop whereas when continue is found it skips the current iteration of the loop and jumps to the next iteration

Both break and continue are executed based on some previous conditions.

Break

for i in range(10):

if i == 4:

break # when the value of i becomes 4 then the break gets triggered and go to the next line

#after the loop

print(i)

Continue

for i in range(10):

if i == 4:

continue # when the value of i becomes 4 then the continue gets triggered and skips the current

# loop and starts the next iteration

print(i)

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

There is no difference between all the three, all of them results a sequence of 0 – 9 integers incrementing them 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.

Using for

for i in range(0,10):

print(i)

Using while

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?

We can perform the operation in following 2 ways:

import spam

spam.bacon()

from spam import bacon

bacon()