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?

AND OR 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

|  |  |  |
| --- | --- | --- |
| Input 1 | Input 2 | output |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

OR

|  |  |  |
| --- | --- | --- |
| Input 1 | Input 2 | output |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

NOT

|  |  |
| --- | --- |
| Input | Output |
| 1 | 0 |
| 0 | 1 |

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?

< less than

> greater than

== is equal

<= less or equal

>= greater or equal

!= not equal

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

Assignment operator: denoted as single ‘=’ sign, used to assign or update the value of a variable, returns nothing.

X=2, y=3

Is equal to : denoted as double ‘=’ sign, used to compare the value of a variable to others, returns Boolean.

X==2, y==3

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')

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.

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?

**CTRL + C**

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

**break** : used to **jump out of the loop once execute**

for i in range(5):

print(i)

break

print(“hello”)

output = 0

loop run for single time after that it jumped out of the loop and didn’t print hello

**continue** : doesn’t allow the next statements of a loop to execute and **starts executing from the beginning of the loop**

for i in range(5):

print(i)

continue

print(‘hello’)

output = 0

to

5

Loop run for five times, didn’t print hello

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

range(10) : it returns a sequence of integers starting from 0 to 9, starting from zero by default, stepping by 1 by default.

range(0,10) : it returns a sequence of integers starting from 0 to 9, starting from zero( given argument), stepping by 1 by default.

range(0,10,1): it will return the same as above but here starting and stepping is specified

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 loop

for i in range(1,11):

print(i)

#using while loop

i=1

while (i <=11):

print(i)

i++

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()