What are the 2 value of the Boolean data type? How do you write them?

Ans> the 2 values are true and false which are also written as True and False.

Example: flag = True

Flag = False

What are the 3 different types of Boolean operators?

Ans> the 3 different types of Boolean operators are

1. And
2. Or
3. Not

* Example a==0 and b==1 this will evaluate to true when both the expressions turn out to be true,
* A==0 or b==0 will evaluate to true when either or both of them are true.
* Not(a<10) this is evaluate to true when the condition a<10 is false

Make a list of Boolean operator’s truth table.

Ans>

1. For And

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **A and B** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

1. For or

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **A and B** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

1. For not

|  |  |
| --- | --- |
| **A** | **Not A** |
| 1 | 0 |
| 0 | 1 |

What are the values of the following expressions?

Ans>

1. (5 > 4) and (3 == 5) this will give result False
2. Not (5 > 4) this will result in False
3. (5 > 4) or (3 == 5) this will result in True
4. Not((5 > 4) or (3 == 5)) this will result in False
5. (True and True) and (True == False) will result in False
6. (not False) or (not True) will result in True

What are the 6 comparison operators?

Ans> the 6 comparison operators are >, <, <=, >=, !=, ==

* < this is used to compare if first value is less than second value
* > this is used to compare if first values is more than 2nd value
* <= is used to check if the first value is less or equal to 2nd value
* >= is used to check if the first value is greater than or equal to 2nd value
* != is used to check if the first value is not equal to 2nd value
* == is used to check if the first value is equal to 2nd value

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

Ans> the equal to operator is used to check if 2 values are equal whereas assignment operator is used to assign values to the variable. Equal to is == whereas assignment is just =.

* Suppose we need to check if 5 is equal to 4 or not we use 5 == 4 but here = cannot be used as it will throw and error.
* But if we want to assign a value to some variable a = 5 is used but if we use == here it will either evaluate to True or False meaning it will produce undesired results.

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’)

Ans> the statements inside the first if, the 2nd if and the else statement are the 3 blocks in the provided code.

Write a 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 stores in spam.

Ans>

if(spam == 1):

print(‘Hello’)

elif(spam == 2):

print(‘Howdy’)

else:

print(“Greetings!”)

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

Ans> we can use CTRL + C to stop the infinite or endless loop. And Infinite loops run indefinitely and only stops when external command is given or break is encountered.

How can you tell the difference between break and continue?

Ans> if we have a condition which when becomes True we do not need to loop any further, for example when we are searching and desired value is found we do not need to further continue the loop so we use the keyword break to break out of the loop.

And if we are in a situation where we do not have anything decided for the loop or function we place continue inside it so that what ever is going on will continue going on, for example putting continue means the loop automatically moves to next iteration when continue is encountered and no other statements after continue are processed.

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

Ans> the values generated by all three is same 0,1,2,3,4,5,6,7,8,9 but yet they are different, the

* range(10) this function is invoked with default starting value of 0 and difference of 1 and will generate numbers until higher limit 10 is obtained (i.e. until including 9)
* range(0,10) here the lower and upper limit are already mentioned and only the difference between successive generated values will be by default 1.
* Range(0,10,1) here the lower, upper limit as well as the difference in succession values are all provided.

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.

Ans>1. for the for loop

for i in range(1,11):

print(i)

for the while loop

j = 1

while( j <= 10):

print(j)

j+=1

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

Ans> to access a function inside a module all we need a “.” Operator.

* To call bacon() we need to write spam.bacon()