

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

**Solution:-** Two values of Boolean data type are True and False

Syntax:- True and False

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

**Solution:-** Boolean Operator are:- AND OR ANDNOT

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 ).

Solution:-

OR

Input	Input	Output
<b>True</b>	<b>True</b>	<b>True</b>
<b>False</b>	<b>False</b>	<b>False</b>
<b>True</b>	<b>False</b>	True
False	True	True

AND

Input	Input	Output
<b>True</b>	<b>True</b>	<b>True</b>
<b>False</b>	<b>False</b>	<b>False</b>
<b>True</b>	<b>False</b>	False

False	True	False
-------	------	-------

#### ANDNOT

<b>True</b>	False
False	True

4. What are the values of the following expressions?

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

**Solution:-** (True) and (False) = True

not (5 > 4)

**Solution:-** not(True) = False

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

**Solution:-** (True) or (False) = True

not ((5 > 4) or (3 == 5))

**Solution:-** ((True) or (False)) = (True) = True

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

(not False) or (not True)

**Solution:-** (True) and (False) (True) or (False)

5. What are the six comparison operators?

**Solution:-** Six comparison operator are :-

1:- >

2:- <

3:- ==

4:- >=

5:- <=

6:-!=

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

**Solution:-** Assignment operator assign the specific value to the variable but the equal operator will compare that the value is equal or not with the given value

**Example of assignment operator:-**

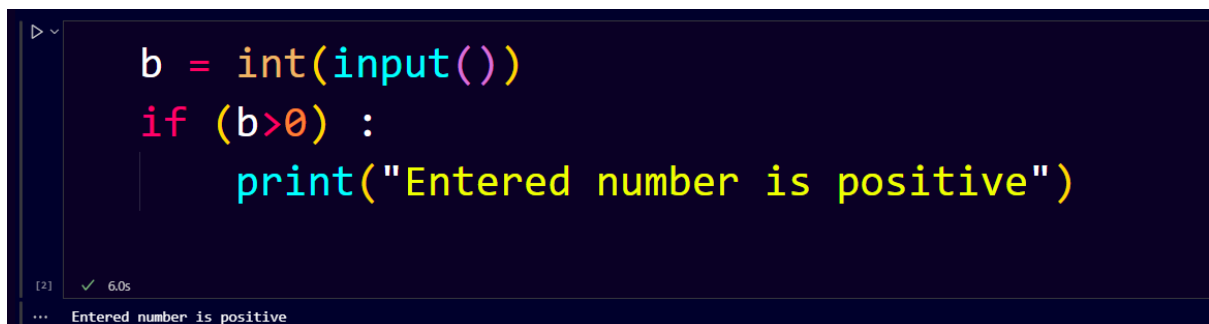


```
a=5
print(a)
```

[1] ✓ 0.0s

... 5

**Example of equal operator**



```
b = int(input())
if (b>0) :
    print("Entered number is positive")
```

[2] ✓ 6.0s

... Entered number is positive

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')
```

## Solution:-

```
spam=0;
if (spam<10) :
    print('eggs')
if (spam>5) :
    print('bacon')
else :
    print('ham')
    print('spam')
    print(['spam'])
```

[1] ✓ 0.0s

... 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.

## Solution:-

```
spam = int(input())
if (spam==1) :
    print("Hello")
elif (spam==2) :
    print("Howdy")
else :
    print("Greetings!")
```

[3] ✓ 4.9s

... Howdy

▶ [ ]

```
spam = int(input())
if (spam==1) :
    print("Hello")
elif (spam==2) :
    print("Howdy")
else :
    print("Greetings!")
```

[4] ✓ 1.3s

... Hello

▶ [ ]

```
▶ spam = int(input())
if (spam==1) :
    print("Hello")
elif (spam==2) :
    print("Howdy")
else :
    print("Greetings!")
```

[5] ✓ 1.0s

... Greetings!

```
▶ 
```

[ ]

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

**Solution:-** Ctrl+ C

```
▶ a=0
while(True) :
    a+=4
    print(a)
```

[2] ⏻ 10.5s

... 4  
8  
12  
16  
20  
24  
28  
32  
36  
40  
44  
48  
52  
56  
60  
64  
68  
72  
76  
80  
84  
88  
92  
96  
100  
...  
16112388  
16112392  
16112396  
16112400

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

**Solution:-** Break-> The break statement will break the loop on the particular condition and will come out of the break statement

```
for x in range (0,5) :  
    if (x==4) :  
        break  
    print(x)
```

[14] ✓ 0.0s

... 0  
1  
2  
3

Continue-> The Continue statement will skip the iteration of the particular condition and will print the rest of the elements

```
for x in range (0,5) :  
    if (x==2) :  
        continue  
    print(x)
```

✓ 0.0s

0  
1  
3  
4

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

**Solution:-** All will give the same output

range(0,10)

```
for x in range (10):  
    print(x)
```

9] ✓ 0.1s

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

range(0,10)

```
▶ for x in range (0,10) :  
    print(x)  
[20] ✓ 0.0s  
... 0  
    1  
    2  
    3  
    4  
    5  
    6  
    7  
    8  
    9
```

range(0,10,1)

```
for x in range (0,10,1) :  
    print(x)  
[21] ✓ 0.0s  
... 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.

**Solution:-**

```
for x in range (1,11) :  
    print(x)  
✓ 0.0s  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

```
x=0
while (x<10) :
    x=x+1
    print(x)
```

[25] ✓ 0.0s

... 1  
2  
3  
4  
5  
6  
7  
8  
9  
10

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

**Solution:-** `spam.bacon()`