

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

Answer:-

The Boolean data type represents two values: true and false.

These values are used in computer programming to represent binary decisions, where true represents a condition that is considered to be true or valid, and false represents a condition that is considered to be false or invalid.

we write them in python as

True

False

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

Answer:-

Boolean operators are used to perform logical operations on Boolean values (true and false) or expressions that evaluate to Boolean values.

There are three primary Boolean operators:

1) AND (&&): The AND operator returns true if both operands are true. Otherwise, it returns false.

2) OR (||): The OR operator returns true if at least one of the operands is true. If both operands are false, it returns false.

3) NOT (!): The NOT operator, also known as the "logical negation," takes a single Boolean operand and returns the opposite Boolean value.

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 evaluates to).

Answer:-

1) AND

True && True - result True

True && False - result False

False && True - result False

False && False - result False

2) OR

True || True - result True

True || False - result True

False || True - result True

False || False - result False

3) Not

!True - result False

!False - result True

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)) = True

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

(not False) or (not True) = True

5. What are the six comparison operators?

Answer:-

Comparison operators are used to compare values or expressions and return a Boolean result (true or false) based on the comparison.

There are six common comparison operators:

- 1) Equal to (==): Checks if the values on both sides of the operator are equal.
- 2) Not equal to (!=): Checks if the values on both sides of the operator are not equal.
- 3) Greater than (>): Checks if the value on the left side is greater than the value on the right side.
- 4) Less than (<): Checks if the value on the left side is less than the value on the right side.
- 5) Greater than or equal to (>=): Checks if the value on the left side is greater than or equal to the value on the right side.
- 6) Less than or equal to (<=): Checks if the value on the left side is less than or equal to the value on the right side.

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

Answer:-

The equal to operator (==) and the assignment operator (=) serve different purposes and are used in different contexts.

Equal to operator (==):

The equal to operator (==) is a comparison operator used to check if two values are equal. It returns true if the values on both sides of the operator are equal; otherwise, it returns false.

Assignment operator (=):

The assignment operator (=) is used to assign a value to a variable. It takes the value on the right side of the operator and stores it in the variable on the left side.

```
x = 5 #here we are assgning a value to variable  
y = 10
```

```
if x == y: # here we are comparing whether x is equal to why or not.  
    print("x is equal to y")  
else:  
    print("x is not equal to y")
```

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")
```

output:
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.

Answer:
spam = int(input("Enter a numbere : "))
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?

Answer:-
ctrl + c

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

Answer:-

break is used to exit the loop entirely.
continue is used to skip 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)?

Answer:
range(10) means loop start with index 0 to index 9.
range(0,10) is same as range(10)
range(0, 10, 1) means loop will start from index 0 to index 9 with interval of on index.

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.

Answer :
for loop
for i in range (1,11):
 print (i)

for while loop
num = 1
while num <= 10:
 print(num)
 num += 1

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

Answer:-
`import spam`

`spam.bacon()`