1. What are the Boolean data type's two values? How do you go about writing them?

Answer:

The Boolean data type is a data type that has one of two possible values usually denoted **True** and **False**.

We go writing them by:

 True and False

T & F Upper case and rest lowercase.

And

We use bool( ) as a boolean function

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

Answer:

The three basic boolean operators are: **AND**, **OR**, and **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).

    Answer:

AND Operator			OR Operator			NOR Operator		
A	B	C	A	B	C	A	B	C
1	1	1	1	1	1	1	1	0
1	0	0	1	0	1	1	0	0
0	1	0	0	1	1	0	1	0
0	0	0	0	0	0	0	0	1

4. What are the values of the following expressions?

Answers:

(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 different types of reference operators?

Answer:

Six different types of reference operators are:

i. : (colon)
ii. , (comma)
iii. (space)
iv. #(pound)
v. @(at)
vi. " "

6. How do you tell the difference between the equal to and assignment operators?

Answer:

Assignment Operator (=)	Equal to Operator (==)
It is an assignment operator	It is a relational or comparison operator
It is used for assigning the value to a variable.	It is used for comparing two values. It returns 1 if both the values are equal otherwise returns 0.
Constant term cannot be placed on left hand side.	Constant term can be placed in the left hand side.
Example: 1=x; is invalid.	Example: 1==1 is valid and returns 1.

7. Describe a condition and when you would use one.

Answer:

The Boolean expression in a conditional statement that determines which branch is executed. Conditional statement. A statement that controls the flow of execution depending on some condition. In Python the keywords if, elif, and else are used for conditional statements.

It is usually **used** when you have to judge one statement on the basis of other. If one **condition** goes wrong, then there should be another **condition** that should justify the statement or logic.

8. Recognize the following three blocks in this code:

```
spam = 0
if spam == 10:
    print('eggs')
    if spam > 5:
        print('bacon')
    else:
        print('ham')
    print('spam')
print('spam')
```
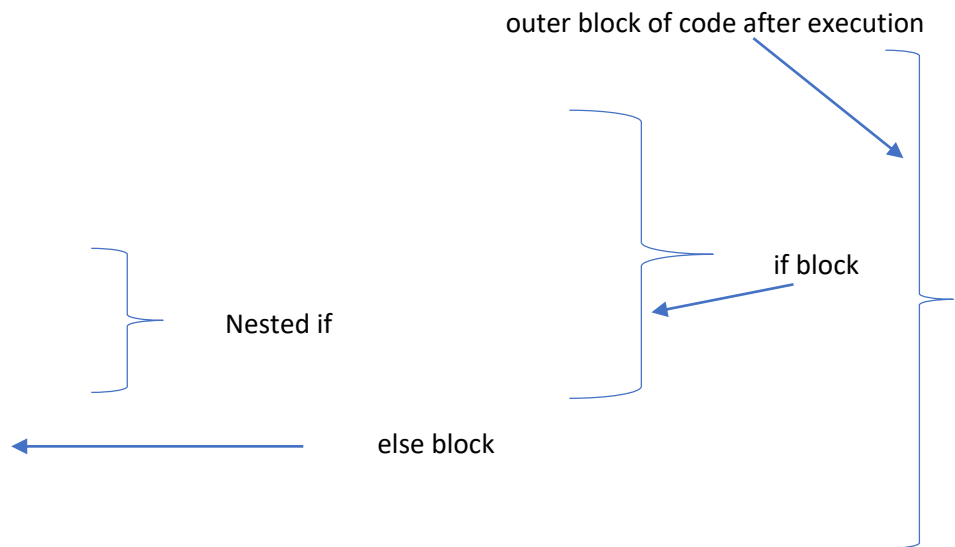
Answer:

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

Nested if

else block

if block

outer block of code after execution

9. Create a programme that prints. If 1 is stored in spam, prints Hello; if 2 is stored in spam, prints Howdy; and if 3 is stored in spam, prints Salutations! if there's something else in spam.

Answer:

```
spam = int(input("Enter the value options:  1, 2 or 3:    The options you entered is:  "))
if spam ==1:
    print("Hello")
elif spam ==2:
    print("Howdy")
```

elif spam ==3:

   print("Salutations!")

else:

   print("You have entered wrong option please re enter")

10. If your programme is stuck in an endless loop, what keys can you press?

Answer:

To exit out of infinite loops on the command line, press CTRL + C .

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

Answer:

Difference between break and continue is that :

'break'- is used for immediate termination of loop.

'continue'- used to terminate the current iteration and resumes the control to the next iteration of the loop.

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

Answer:

There is no difference between the range(10), range(0,10), and range(0,10,1)

The meaning of all are same and all returns the same output.

As if we consider range(0,10,1) then 0 is the starting point, 10 is the end point and 1 is the jump point.

All will print the same output: if we print in the list value by writing code**: list (range(10))**

Output will be: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

13. Using a for loop, write a short programme that prints the numbers 1 to 10 Then, using a while loop, create an identical programme that prints the numbers 1 to 10.

Answer:

Using for loop

a=int(input("input number till you print :   You want to print the counting till "))

```
for i in range(1,a+1):

    print(i)
```

here a value will be 10 and we will get output from 1 to 10.

Using While loop

```
a=int(input("input number till you print :   You want to print the counting till "))

i=1

while i <= a:

    print(i)

    i=i+1
```

here a value will be 10 and we will get output from 1 to 10.

14. If you had a bacon() function within a spam module, what would you call it after importing spam?

Answer:

Spam.bacon( )