Questions

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

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

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).

4. What are the values of the following expressions?

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

not (5 > 4)

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

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

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

(not False) or (not True)

5. What are the six comparison operators?

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

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')
```

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.

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

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

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

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.

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

Answers

1) The values of Boolean data types are "True" and "False". Every expression in python gives a value which is either True or False. Ex: a>b is True is say a= 10 and b = 8 but it would be False if it was vice-versa.
2) The three Boolean operators are AND, OR and NOT, represented by and , or and not keywords respectively.
3) OR –

x	**y**	**x or y / x +y**
0	0	0
1	0	1
0	1	1
1	1	1

AND-

x	**y**	**x and y**
0	0	0
1	0	0
0	1	0
1	1	1

NOT-

x	**Not x**
0	1
1	0

4)

(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) The six comparison operators are:
==, >= , <= , != , < , >

6) The equal to operator is "==" and assignment operator is "=" and hence they can be differentiated by looking at the operator. Assignment operator is used when we need to assign a value to an operand/variable and it stores that value at a memory location pointed to by that variable, while equal to operator is used to check if two values are same or if two variables have the same value or not.

E.g. If I want to assign the value 3.14 to the variable "pie" in order to use it to calculate the area and perimeter of a circle at multiple points in the source code, I would have to use the assignment operators in the following way:

```
pie = 3.14
```

while in another section of the code I have to check whether pie's values I actually equal to 3.14 or not. Then I would have to use the equal to operator.

```
if pie == 3.14:
    print("This is the correct value of pie")
```

7)

Block 1: spam = 0

Block 2: if spam == 10:

```
    print('eggs')
```

Block 3: if spam > 5:

print('bacon')

else:

print('ham')

print('spam')

print('spam')

8) spam = input(“Enter whatever you want : ”)
if spam == 1:
print(“Hello”)
elif spam == 2:
print(“Howdy”)
else:
print(“Greetings!”)

9) control + c (mac) , ctrl +c(windows)

10) Break stops the complete execution of the loop and break out of it and moves on to the next statement after the loop, while continue just skips the current iteration and rest of the code in the loop (under the continue statement) and moves onto the next iteration.

11) All are same.

12)

USING FOR LOOP

for i in range (1,11):

print(i)

Using WHILE LOPP

i = 1

while i <= 10:

print(i)

```
    i +=1
```

13) spam.bacon()