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

• True and False, using capital T and F, with the rest of the word in lowercase.

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

• 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 ).

* True and True is True.
* True and False is False.
* False and True is False.
* False and False is False.
* True or True is True.
* True or False is True.
* False or True is True.
* False or False is False.
* not True is False.
* not False is 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)) : False

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

(not False) or (not True) : True

5. What are the six comparison operators?

* less than (<)
* less than or equal to (<=)
* greater than (>)
* greater than or equal to (>=)
* equal to (==)
* not equal to (!=)

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

* == is the equal to operator that compares two values and evaluates to a Boolean
* = is the assignment operator that stores a value in a variable.

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')

Answer:

if spam == 10:

print('eggs')

if spam > 5:

print('bacon')

else:

print('ham')

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.

* The code:

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?

• Press CTRL-C to stop a program stuck in an infinite loop.

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

The Python break statement stops the loop in which the statement is placed. A Python continue statement skips a single iteration in a loop. Both break and continue statements can be used in a for or a while loop.

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

* range(10): range(stop) takes one argument. When user call range() with one argument, user will get a series of numbers that starts at 0 and includes every whole number up to, but not including, the number that user have provided as the stop.

Output : 0 1 2 3 4 5 6 7 8 9

* range(0,10) : range(start, stop) takes two arguments. When user call range() with two arguments, user get to decide not only where the series of numbers stops but also where it starts, so user don’t have to start at 0 all the time.
* Range(0,10,1) : range(start, stop, step) takes three arguments. When the user call range() with three arguments, the user can choose not only where the series of numbers will start and stop but also how big the difference will be between one number and the next. If the user doesn’t provide a step, then range() will automatically behave as if the step is 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.

Using for loop:

for i in range(1, 11):

print(i)

Using while loop:

i = 1

while(i<=10):

print(i)

i += 1

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

• spam.bacon()