1. The two values of Boolean data type are True and False. They are written in such a way that their first letter is a capital letter because python is case-sensitive language.
2. AND, OR, NOT are three Boolean operators.
3. AND Truth Table:

Input1	Input2	Output
False	False	False
False	True	False
True	False	False
True	True	True

OR Truth Table:

Input1	Input2	Output
False	False	False
False	True	True
True	False	True
True	True	True

NOT Truth Table:

Input	Output
False	True
True	False

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. Six comparison operators are < (Less Than), > (Greater Than), <= (Less Than or Equal To), >= (Greater Than or Equal To), == (Equal To), != (Not Equal To).
6. The difference between the equal to and assignment operator is that for equal to operator '=' equal sign is used twice '==' where as in the equal to operator equal sign is used only once '='.
   Equal to operator '==' is used to check whether the two operands are equal in value or not. If they are equal then it returns True otherwise it returns False.
   Assignment operator '=' is used to assign a value to any variable.
7. spam = 0

```
---------------------------------
if spam == 10:
   print('eggs')      →Block 1
---------------------------------
---------------------------------
if spam > 5:
   print('bacon')    →Block 2
---------------------------------
---------------------------------
else:
```

```
    print('ham')
    print('spam')    →Block 3
-------------------------------------
print('spam')
```

8.
```
if spam==1:
    print('Hello')
if spam==2:
    print('Howdy')
else:
    print('Greetings!')
```

9. If the program is stuck in an endless loop we can press Ctrl+C keys to stop the loop.
10. The difference between the break and continue is that 'break' is used to stop the loop and take the cursor to the end of the loop to execute next set of statements. Whereas 'continue' is used the stop the current iteration and takes us to the next consecutive iteration.
11. In a for loop, there is no difference between range(10), range(0,10), and range(0,10,1). All these statements are just the same and gives us the same output. range() function generates a range of value between the provided arguments.
    range() function takes 3 arguments (starting inclusive value, ending exclusive value, Jump/step value).
12. Program that prints the numbers 1 to 10 using a for loop:

```
for i in range(10):
    print(i+1)
```

    Program that prints the numbers 1 to 10 using a while loop:

```
c=1
while a<=10:
    print(c)
    c+=1
```

13. If a function named bacon() is inside a module named spam, we can call it after importing spam as follows:
    spam.bacon()