1. Functions help to use the same piece of code to run many times in the program, so it helps in reusability. Secondly it helps to make huge program into small blocks of code, so it helps in debugging. Thirdly functions can be shared & used by other programmers.
2. When the function is called, code inside the function runs.
3. Function statements contains the name of the function followed by list of arguments enclosed in the parenthesis.
4. Function is a block of code to perform some task. It contains opening & closing parenthesis (). Function call means call the function name followed by () & if any value is given inside the () to get the results.
5. There is only 1 global scope available per program execution.
6. def mul():

i=10

j=20

print(i\*j)

mul() #function call

200 #Variables in the local scope gets multiplied to give result.

1. Return statement stops a function from executing & returns a value to the main programme. Yes, it is possible to have return value in an expression.

def mul():

return(10\*20)

mul()

200

1. If a function does not have a return statement, it returns None when the function is called.

def mul():

a= 10

b=20

a\*b

mul() #when the function is called, it does not return any value, hence returns None.

1. By using the keyword global, any variable inside the function which is local to that function can be made global.

def mul():

a=10

b=20

global c # c is made global

return (a\*b)

for c in range(10,0,-1):

print(c)

1. Nonetype
2. Import keyword imports the code of “areallyourpetsnamederic” module available to use in your coding. If the code is module is not installed, it can be installed using pip command
3. This function can be called with spam. bacon()
4. When errors are encountered, system crash can be avoided using exception handling.
5. Try statement contains the block of code to be tested for errors while it is being executed. In the except block error is handled