1. What is the result of the code, and explain?

>>> X = 'iNeuron'

>>> def func():

print(X)

>>> func()

The output would be

iNeuron

When you call function, func(), the print message inside it would be printed with value of X

in this case X value is iNeuron

2. What is the result of the code, and explain?

>>> X = 'iNeuron'

>>> def func():

X = 'NI!'

>>> func()

>>> print(X)

The output would be

iNeuron

func() does not print any value as it does not have any print statement inside it.

print(X) prints the value of X, here value of X is iNeuron

3. What does this code print, and why?

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

print(X)

>>> func()

>>> print(X)

The output would be

NI

The value of X is assigned to NI inside the function.

hence, func() gives output NI

4. What output does this code produce? Why?

>>> X = 'iNeuron'

>>> def func():

global X

X = 'NI'

>>> func()

>>> print(X)

The output would be

NI

func() does not print anything as it does not have any print statement inside it.

print(X) prints the value of X which is NI

5. What about this code—what’s the output, and why?

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

def nested():

print(X)

nested()

>>> func()

>>> X

nested()

this gives output :

It gives error message "name 'nested' is not defined."

nested function cannot be called directly, hence, it gives an error message.

>>> func()

>>> X

This gives output

'iNeuron'

6. How about this code: what is its output in Python 3, and explain?

>>> def func():

X = 'NI'

def nested():

nonlocal X

X = 'Spam'

nested()

print(X)

>>> func()

nested()

It gives error message "name 'nested' is not defined"

nested function cannot be called directly

print(X)

It gives output

Spam

Since X is assigned the value 'Spam' print(X) gives output Spam

>>> func()

func() does not give any output as it does not have any print statement inside it