Assignment 22

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

>>> X = 'iNeuron'

>>> def func():

print(X)

>>> func()

Output will be iNeuron because the function intially looks for the variable X in its local scope,But since there is no local variable X, its returns the value of global variable X.

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

>>> X = 'iNeuron'

>>> def func():

X = 'NI!'

>>> func()

>>> print(X)

Output will be iNeuron because X = 'iNeuron' is global scope variable whereas X = 'NI!' is defined inside a function so its local scope variable. When we called the function it return the value of global variable X i.e, ineuron.

3. What does this code print, and why?

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

print(X)

>>> func()

>>> print(X)

Output will be NI and iNeuron. X=NI is in the local scope of the function func() hence the function prints the x value as  NI.  X = 'iNeuron' is in the global scope. hence print(X) prints output as iNeuron

4. What output does this code produce? Why?

>>> X = 'iNeuron'

>>> def func():

global X

X = 'NI'

>>> func()

>>> print(X)

Output will be NI because we are using global keyword inside the function func() so it directly access the variable in X in global scope and changes its value to 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

Output will be NI and ‘ineuron’ because nested function are defined other function func() so it has direct access to the variable and global variable X is 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()

Output will be Spam. nonlocal keyword in python is used to declare a variable as not local. Hence the statement X = "Spam" is modified in the global scope.