**Assignment 22**

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

X = ‘iNeuron’

def func():

print(X)

func()

Ans: The output will be ‘iNeuron’ because X is a global variable and func function will print X.

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

X = ‘iNeuron’

def func():

X = ‘NI!’

func()

print(X)

Ans: The result of the code is ‘iNeuron’ because we are printing the value of global X variable. X in func() is local to the function hence it is not changing the value of X by calling the function func().

3. What does this code print, and why?

X = 'iNeuron'

>>> def func():

X = 'NI'

print(X)

>>> func()

>>> print(X)

Ans: The code will print ‘NI’ and ‘iNeuron’ because first its calling func() which will print local X variable and the code will print global X variable which is iNeuron.

4. What output does this code produce? Why?

>>> X = 'iNeuron'

>>> def func():

global X

X = 'NI'

>>> func()

>>> print(X)

Ans: The output will be ‘NI’ because we are changing the global value of X to ‘NI’ by using the global keyword in the func().

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

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

def nested():

print(X)

nested()

>>> func()

>>> X

Ans: The output of the program is ‘NI’ and ‘iNeuron’ because nested() function is defined in func(), therefore, ‘NI’ value of X is getting printed first and when we get out the function global value of X is getting printed which 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()

Ans: The output of the program is ‘Spam’ because by using nonlocal keyword for X we are taking variable X of func() function instead of creating a local variable of nested() function. Therefore, we are changing the value of X from ‘NI’ to ‘Spam’