Ans 1 – the following code x = ‘hello’

Def new():

Print(x)

new()

will give the output = hello as we are assigning a value to a variable and then with the help of a function we are printing the value of the variable.

Ans 2 - – the following code x = ‘hello’

Def new():

x = ‘no’

new()

print(x)

will give the output as hello as x =hello is still a global variable

ans 3 - the following code x = ‘hello’

Def new():

x = ‘no’

print(x)

new()

print(x)

will give 2 outputs no and hello as in func new() we are calling the local assigned value of x and in the print(x) statement we are calling the globally assigned value of x.

ans 4 - x = 'hello'

def new():

global x

x = 'no'

print(x)

new()

print(x)

will give the result as no and no 2 times as we have assigned the value of x as no globally in the func which will affect the value of x even if we are calling/printing it outside the function new().

Ans 5

x = 'hello'

def new():

x = 'no'

def nested():

print(x)

nested()

print(x)

the above code will give the output as hello, hello 2 times as nested is just picking the global value of x and printing it.

Ans6 –

def func():

X = 'NI'

def nested():

nonlocal X

X = 'spam'

nested()

print(X)

func()

*#Nonlocal variables are used in nested functions whose local scope is not defined.*

*#This means that the variable can be neither in the local nor the global scope. it print the updated value from nested*

*#function*

It will just print spam.