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

>>> X = 'iNeuron'
>>> def func():
print(X)
>>> func()

```
x = 'iNeuron'  # Defining a string x
def func():   # Creating a function with the name func
    print(x)  # Printing the string x within the function body
func()      # Calling the function
```

```
iNeuron
```

2. What is the result of the code, and explain?
>>> X = 'iNeuron'
>>> def func():
X = 'NI!'
>>> func()
>>> print(X)

```
x = 'iNeuron'  # Defining a string x
def func():    # Creating a function with the name func
    x = 'NI!'  # assigning x = 'NI!'
func()         # Calling function without any return
print(x)       # Printing the global variable x = 'iNeuron'
```

```
iNeuron
```

3. What does this code print, and why?
>>> X = 'iNeuron'
>>> def func():
X = 'NI'
print(X)
>>> func()
>>> print(X)

```
x = 'iNeuron'  # Defining a string x
def func():    # Creating a function with the name func
    x = 'NI'   # assigning x = 'NI!'
    print(x)   # printing the local variable x ='NI'
func()         # Calling function without any return
print(x)       # Printing the global variable x = 'iNeuron'
```

```
NI
iNeuron
```

4. What output does this code produce? Why?
>>> X = 'iNeuron'
>>> def func():
global X
X = 'NI'
>>> func()
>>> print(X)

```
x = 'iNeuron'   # Defining a string x
def func():     # Creating a function with the name func
    global x    # Creating a global variable x
    x = 'NI'    # Assigning variable x with 'NI'
func()          # Calling function without return
print(x)        # Printing the global variable x = 'NI'
```

```
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

```
X = 'iNeuron'       # Defining a string x
def func():         # Creating a function with the name func
    X = 'NI'        # assigning x = 'NI'
    def nested():   # Creating another nested function with the name nested
        print(X)    # This will print x = 'NI'
        nested()
func()              # Calling main function func. But there is no return or print
X                   # This will show the global variable string 'iNeuron'
```

```
'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()

```
def func():
    X = 'NI'
    def nested():
        nonlocal X
        X = 'Spam'
        nested()
        print(X)
func()
 # This function call will not produce any output as there is no return or print statement in main function func
```