PYTHON ASSIGNMENT 22

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

>>> X = 'iNeuron'

>>> def func():

print(X)

>>> func()

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

>>> X = 'iNeuron'

>>> def func():

X = 'NI!'

>>> func()

>>> print(X)

3. What does this code print, and why?

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

print(X)

>>> func()

>>> print(X)

4. What output does this code produce? Why?

>>> X = 'iNeuron'

>>> def func():

global X

X = 'NI'

>>> func()

>>> print(X)

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

>>> X = 'iNeuron'

>>> def func():

X = 'NI'

def nested():

print(X)

nested()

>>> func()

>>> X

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()

SOLUTIONS

```
# 1. Result and Explanation

X = 'iNeuron'

def func():

    print(X)


func()  # Output: iNeuron

# Explanation: The function `func` prints the value of the global variable X, which is 'iNeuron'.


# 2. Result and Explanation

X = 'iNeuron'

def func():

    X = 'NI!'

func()

print(X)  # Output: iNeuron

# Explanation: The function `func` creates a local variable X with the value 'NI!', but it doesn't affect the global X.


# 3. Result and Explanation

X = 'iNeuron'

def func():

    X = 'NI'
```

```
    print(X)


func()  # Output: NI
print(X)  # Output: iNeuron
# Explanation: The function `func` creates a local variable X with the value 'NI' and prints it. The global X remains unchanged.


# 4. Result and Explanation
X = 'iNeuron'
def func():
    global X
    X = 'NI'

func()
print(X)  # Output: NI
# Explanation: The function `func` uses the `global` keyword to modify the global variable X, changing its value to 'NI'.


# 5. Result and Explanation
X = 'iNeuron'
def func():
    X = 'NI'
    def nested():
        print(X)
    nested()
```

```
func()  # Output: NI
print(X)  # Output: iNeuron
# Explanation: The nested function `nested` prints the value of the local X within the scope of the `func` function.


# 6. Result and Explanation (Python 3)
def func():
    X = 'NI'
    def nested():
        nonlocal X
        X = 'Spam'
    nested()
    print(X)


func()  # Output: Spam
# Explanation: The `nonlocal` keyword is used to indicate that the variable X is a non-local variable, and its value is changed in the nested function.
```