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()
```

1. The code defines a variable X and a function `func`. When the `func` function is called, it prints the value of X. Since X is defined outside the function, it will print the value of X which is 'iNeuron'. Output:

`iNeuron`

2. The code defines a variable X and a function `func`. When the `func` function is called, it creates a new local variable X with the value 'NI!'. The `print(X)` statement outside the function will print the original value of X which is 'iNeuron'. Output:

`iNeuron`

3. The code defines a variable X and a function `func`. When the `func` function is called, it creates a new local variable X with the value 'NI' and prints its value. The `print(X)` statement outside the function will print the original value of X which is 'iNeuron'. Output:

`NI`
`iNeuron`

4. The code defines a variable X and a function `func`. When the `func` function is called, it declares X as a global variable and assigns it a new value 'NI'. The `print(X)` statement outside the function will print the new value of X which is 'NI'. Output:

`NI`

5. The code defines a variable X and a function `func`. When the `func` function is called, it creates a new local variable X with the value 'NI' and defines a nested

function `nested` which prints the value of `X`. When the `nested` function is called, it will print the value of the local variable `X` which is 'NI'. Finally, the value of `X` is returned but not assigned to anything. Output:

```
NI
```

6. The code defines a function `func` which creates a new local variable `X` with the value 'NI' and defines a nested function `nested`. When the `nested` function is called, it uses the `nonlocal` keyword to indicate that `X` is not a local variable but a variable from the enclosing scope. It assigns `X` a new value 'Spam'. Finally, `func` prints the value of `X`, which was modified by the `nested` function, so it will print 'Spam'. Output:

```
Spam
```