**#  A variable can be used in the area where it is created. This area is called the “scope” of the variable.**

**# Local Scope**

**# A variable made inside a function can only be used inside that same function.**

**# You can’tre stu use it outside the function.**

def myfunc():

  x = 300

  print(x)

myfunc()

print(x)

**# Function Inside Function**

**# the variable x is not available outside the function, but it is available for any function inside the function:**

def myfunc():

  x = 300

  def myinnerfunc():

    print(x)

  myinnerfunc()

myfunc()

**# Global Scope**

**# A variable created in the main body of the Python code is a global variable and belongs to the global scope.**

x = 300

def myfunc():

  print(x)

myfunc()

print(x)

**# Naming Variables**

**# If you operate with the same variable name inside and outside of a function, Python will treat them as two separate variables**

x = 10  **# Global variable**

def my\_function():

    x = 5  **# Local variable**

    print("Inside function, x =", x)

my\_function()

print("Outside function, x =", x)

**# Global Keyword**

**# If you need to create a global variable, but ack in the local scope, you can use the global keyword**

def myfunc():

  global x

  x = 300

myfunc()

print(x)

**# Also, use the global keyword if you want to make a change to a global variable inside a function.**

x = 300

def myfunc():

  global x

  x = 200

myfunc()

print(x)

**# The nonlocal keyword is used inside a nested function (a function inside another function) to change a variable from the outer function.**

def outer():

    x = 5

    def inner():

        nonlocal x

        x = 10

    inner()

    print(x)

outer()