

Local vs. Global Variables



CS111 Computer Programming

Department of Computer Science
Wellesley College

Local variables

Local variables exist only **within** a function's body. They cannot be referred outside of it.

Parameters are also local variables that are assigned a value when the function is invoked. They cannot be referred outside the function too.

```
def rightTrianglePerim(a, b):
    c = hypotenuse(a, b)
    return a + b + c

In [1]: rightTrianglePerim(3, 4)
Out [1]: 12.0

In [2]: c
NameError: name 'c' is not defined

In [3]: a
NameError: name 'a' is not defined

In [4]: b
NameError: name 'b' is not defined
```

Local variables in the Frame Model

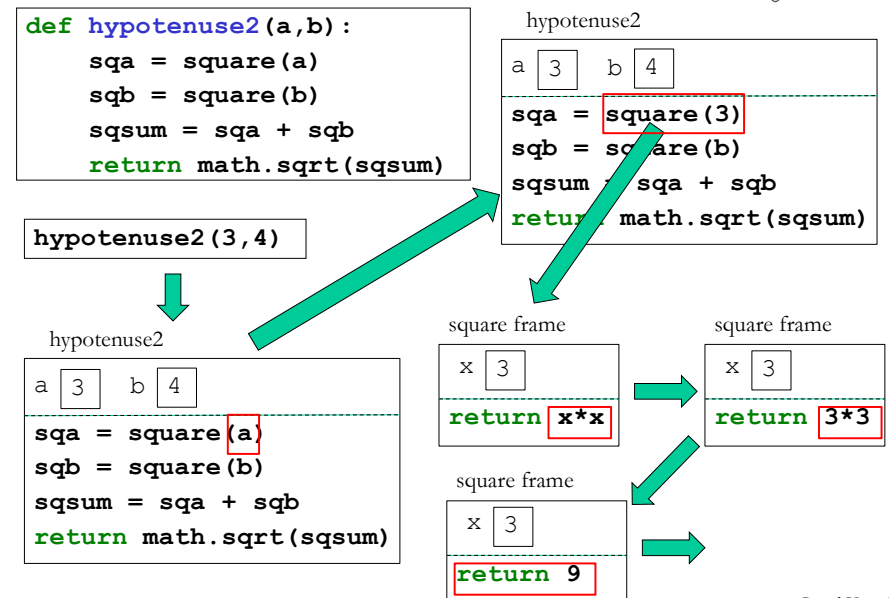


We've seen numerous examples of functions that use local variables, but we haven't explained how local variables work in the execution model with function frames.

We'll do that now with the `hypotenuse2` function:

```
def hypotenuse2(a, b):
    sqa = square(a)
    sqb = square(b)
    sqsum = sqa + sqb
    return math.sqrt(sqsum)
```

Functions w/local variables: hypotenuse2 [1]





Counter function

How can we define a zero-parameter count function that returns the number of times it has been called?

```
count() → 1  
count() → 2  
count() → 3  
count() → 4
```

```
c = 0 # global variable storing the current count  
  
def count():  
    global c # What happens if we forget this?  
    c = c + 1  
    return c
```