**What is 'if \_\_name\_\_ == "\_\_main\_\_"' for?**

The if \_\_name\_\_ == "\_\_main\_\_": ... trick exists in Python so that our Python files can act as either reusable modules, or as standalone programs. As a toy example, let’s say that we have two files:

$ cat mymath.py

def square(x):

return x \* x

if \_\_name\_\_ == '\_\_main\_\_':

print "test: square(42) ==", square(42)

$ cat mygame.py

import mymath

print "this is mygame."

print mymath.square(17)

In this example, we’ve written mymath.py to be both used as a utility module, as well as a standalone program. We can run mymath standalone by doing this:

$ python mymath.py

test: square(42) == 1764

But we can also use mymath.py as a module; let’s see what happens when we run mygame.py:

$ python mygame.py

this is mygame.

289

Notice that here we don’t see the ‘test’ line that mymath.py had near the bottom of its code. That’s because, in this context, mymath is not the main program. That’s what the if \_\_name\_\_ == "\_\_main\_\_": ... trick is used for.