

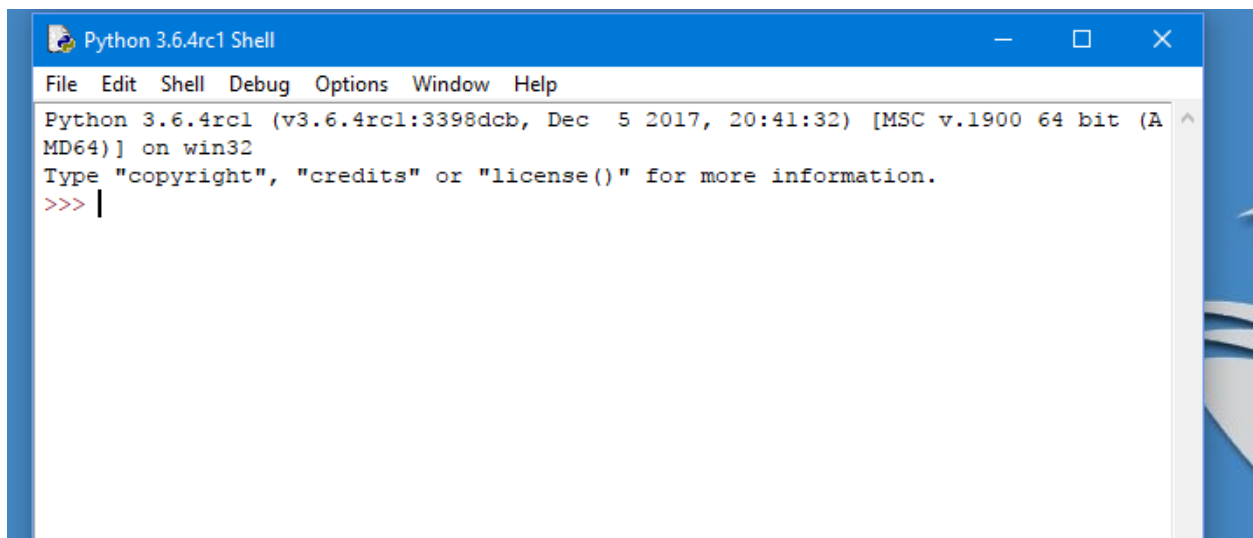
EE194/BIO196: Modeling Biological Systems: HW #0

Introduction to the PC lab, Python and Idle

The first thing to do is go to the PC Lab (it is in Halligan Hall, room 122) and log into your PC. Note that the PCs in Halligan (and, for that matter, all other computers in Halligan) are on the Electrical Engineering/Computer Science network, which is distinct from the Tufts network. Thus, your usual Tufts UTLN and password will *not* allow you to log into any of these machines. However, if you registered for this class, you should already have an account. If not, you can get one by e-mailing staff@eecs.tufts.edu or by walking in at Halligan room 231C.

Once you are logged into a machine, you can launch Idle. On the Windows Start menu, scroll down to the “P”s and click on “Python 3.6;” a submenu will pop up and you should then click on “IDLE (Python 3.6 64-bit)”. Note that the PCs also have Python 2.6 available, but you should not use that.

Once you click on “IDLE (Python 3.6 64-bit),” an IDLE window will pop up:



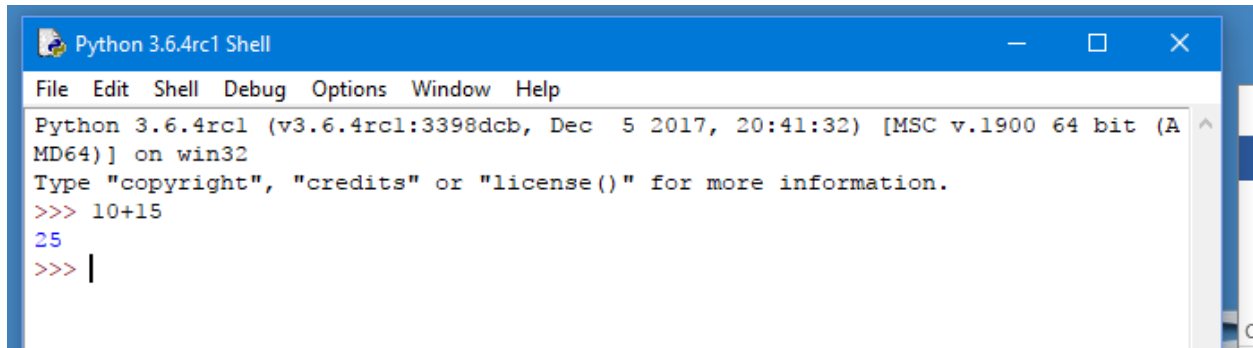
The three greater-than signs, “>>>”, are indicating that IDLE is ready for you to type there.

This is what is called the “Interpreter” (or “shell”) window. The Interpreter allows us to enter commands directly into Python, and as soon as we enter in a command, Python will execute it and display its result. We’ll be using this Interpreter window a lot when we’re exploring Python: it’s very nice because we get back our results immediately. If it helps, we can think of it as a very powerful calculator.

Let’s start out by doing just that: using Python as a calculator.

In general, you type a command into the command window at the “>>>” prompt, and Python invokes the command and computes the results.

At the prompt, type $10+15$ and “Enter”:

A screenshot of a Python 3.6.4rc1 Shell window. The window title is "Python 3.6.4rc1 Shell". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The main text area shows the following content:

```
Python 3.6.4rc1 (v3.6.4rc1:3398dcb, Dec 5 2017, 20:41:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> 10+15
25
>>> |
```

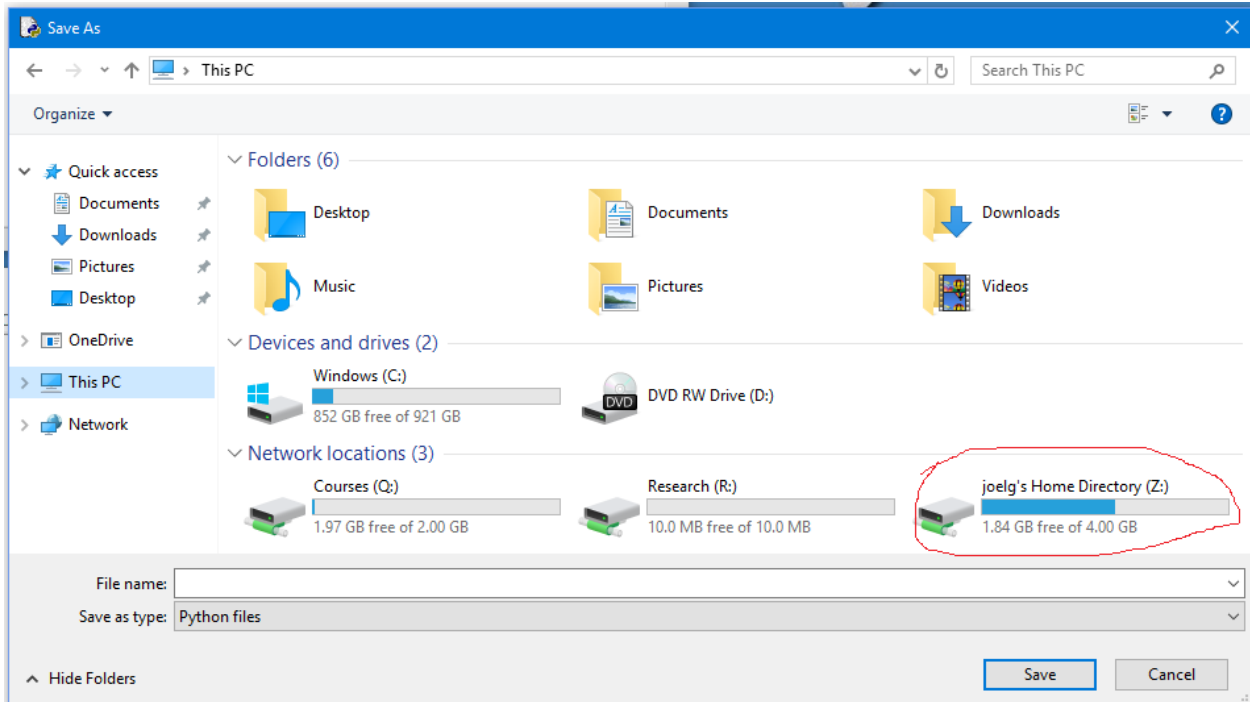
Now do the same thing for a few more calculations; e.g., $20-5$ and $10*10$. Note that you can also use parentheses; e.g., try out both $3+4*5$ and $(3+4)*5$. The order of operations of Python is just what you learned in high school; multiplication and division first, and then addition and subtraction. In Python, multiplication uses “ $*$ ” and division uses “ $/$ ”.

Try something illegal, like $1/0$. What happens? Python’s error messages can sometimes be a bit difficult to decipher; with practice, you’ll get good at it.

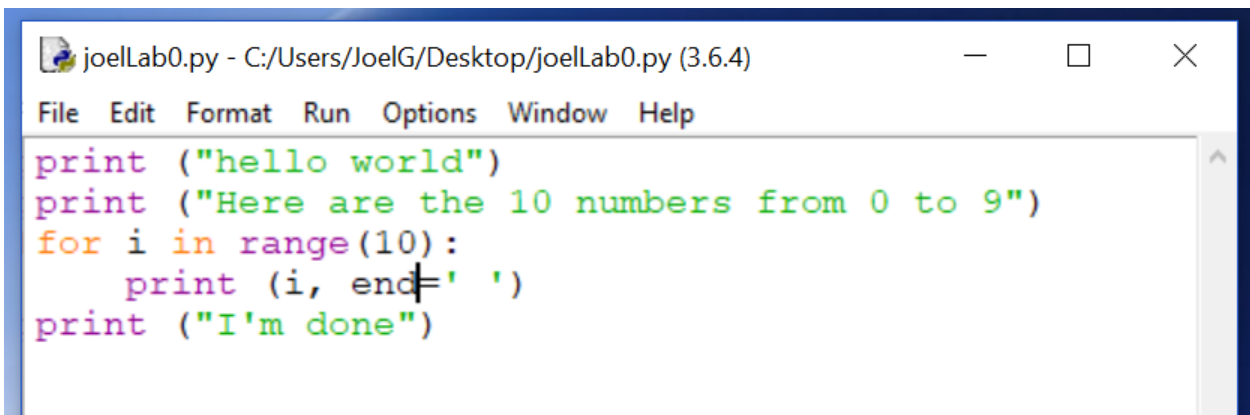
Python has plenty of operators other than $+$, $-$, $*$ and $/$. Try typing $\text{sqrt}(9)$ and also, on a different line, $\text{sqrt}(25)$. What do you think “ sqrt ” is doing?

Similar to sqrt , Python also has round , abs and log10 . Try these out; what do you think they do?

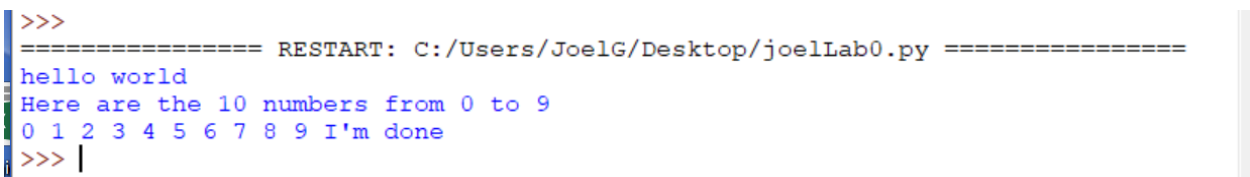
Next, please try the tutorial at <https://www.cs.uky.edu/~keen/help/python-tutorial/Lab0.html> . It’s the more reasonable one for our purposes that I could find on the web. However, their machine setup is not quite identical to ours, so be warned. For example, on our systems, your own personal folder for files is on the “Z” drive. Thus, when you are saving files, you should click on “This PC” and then use the Z drive:



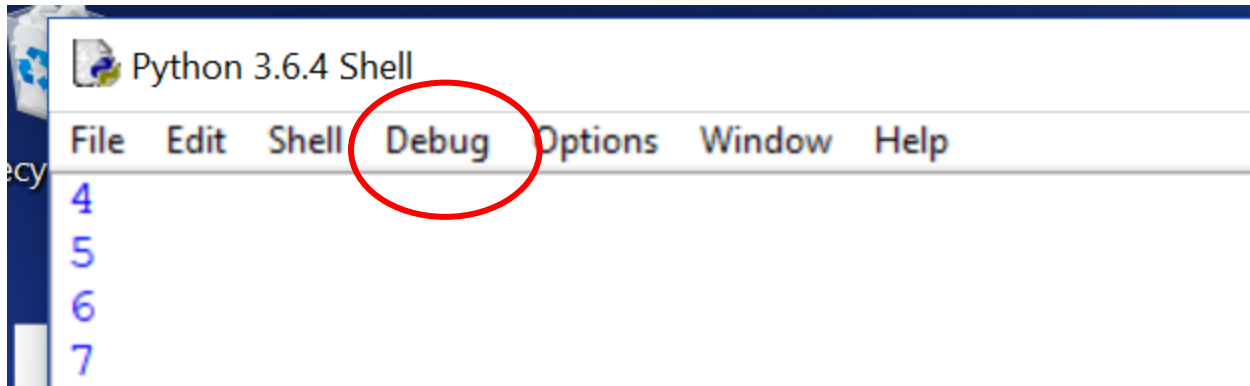
Once you're finished with that online tutorial, you're ready to learn how to use IDLE as a *debugger*. Start with the same program as before:



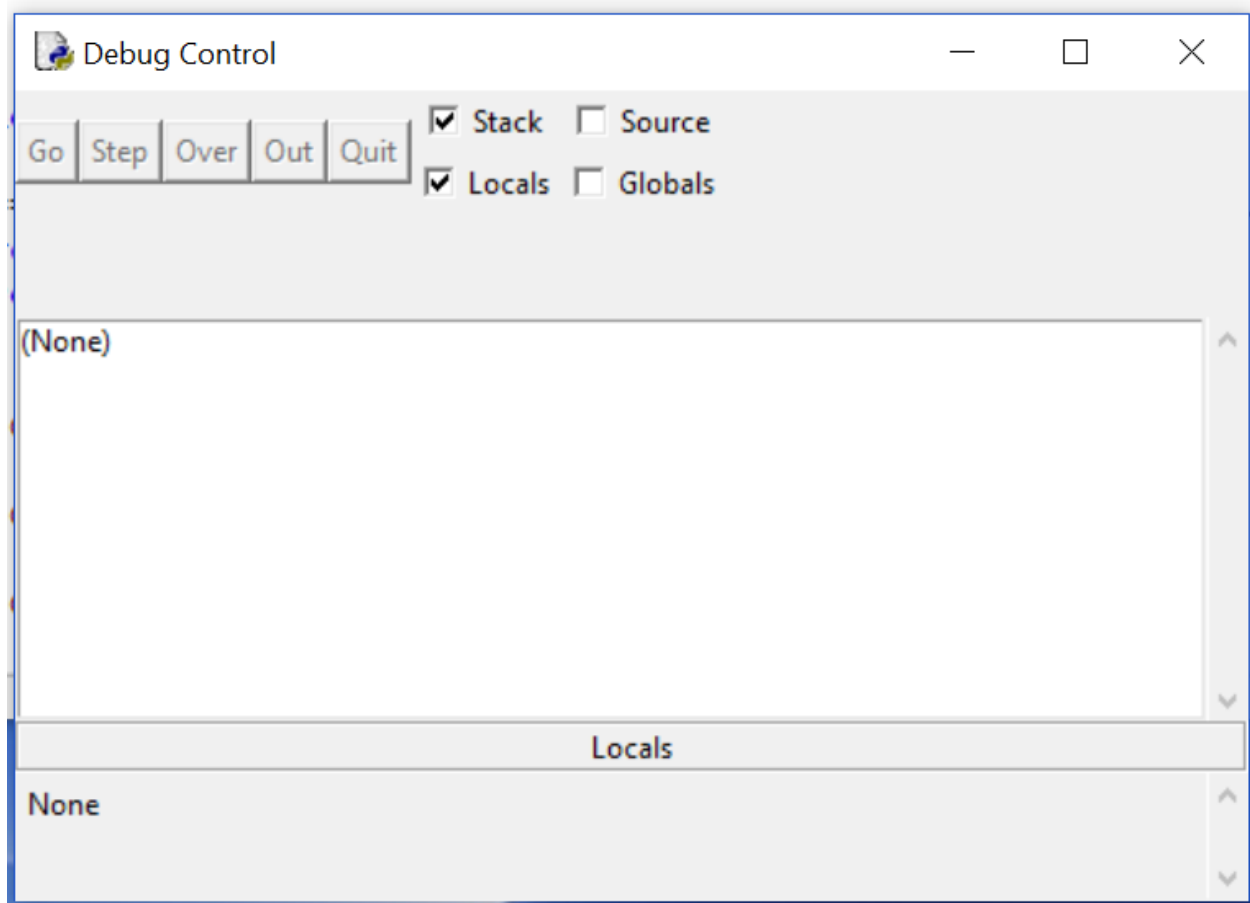
When you run it, you get an output in the shell window:



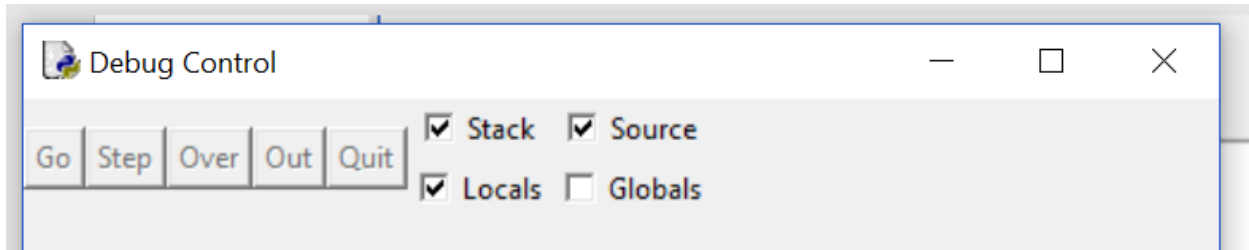
In this small example, it's fairly obvious which line of code does what. For a bigger program, it may be a lot less obvious, and we may want to step through our program line by line to watch it run. IDLE lets you do that also. Start by turning on the debugger in the IDLE shell window:



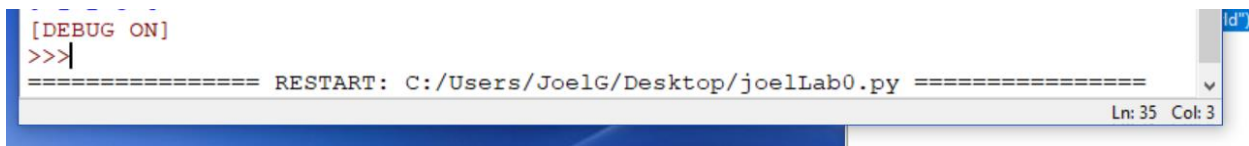
When you click on “Debug”, a submenu will pop up and you should click on “Debugger”. This will pop up the *debug-control* window



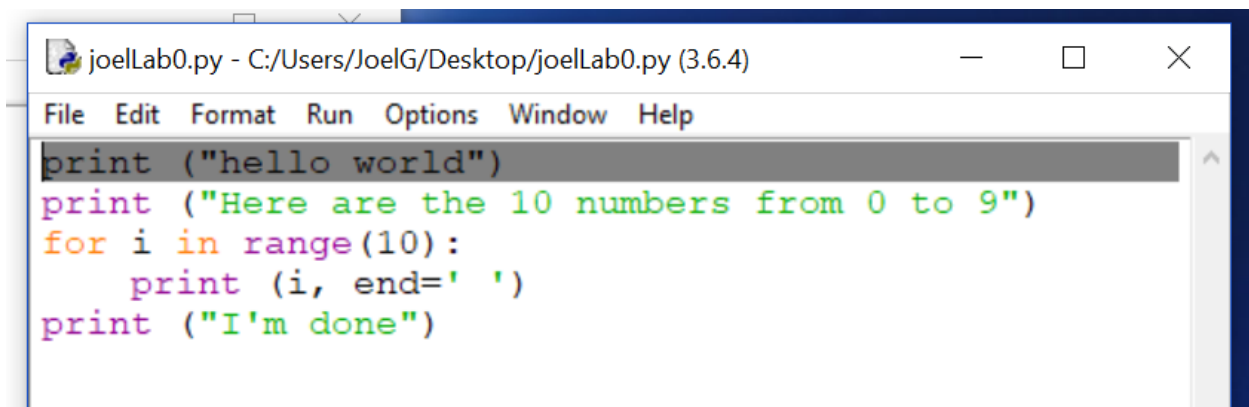
In this window, you should then click on the “Source” checkbox, so that it becomes checked:



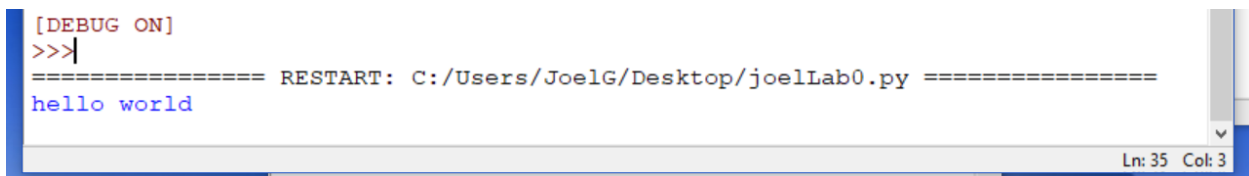
With this done, go back to your Program window and click on “Run” and then “Run Module” again. You will notice that the shell window does not start typing out the results of the program. Instead, it just types out the “RESTART” message and then waits:



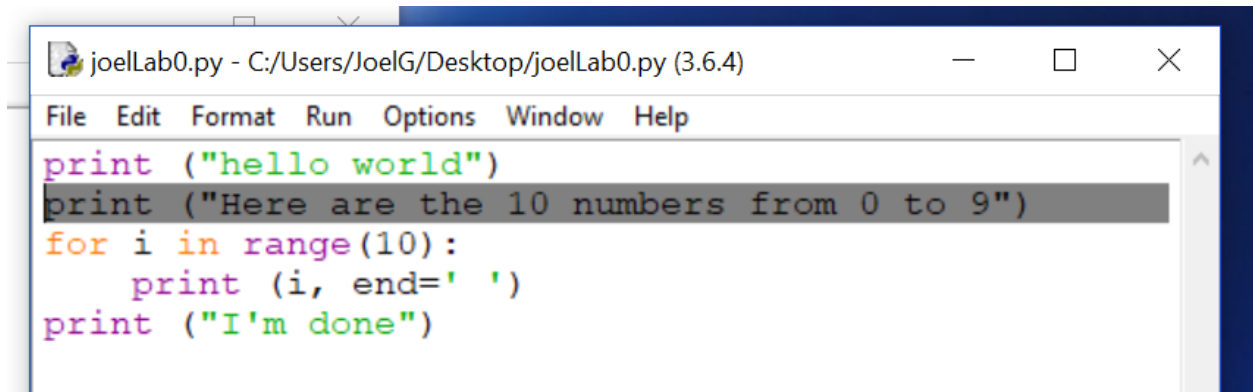
You will also notice that your Program window has the top line highlighted in grey:



This lets you know that Python is ready to execute that line. Next, go back to the Debug Control window and click once on the “Over” button. Do not click on the other buttons, such as “Step” or “Out” yet. You will notice that a few things happen. First, the results of executing the “print” statement show up in the Shell window:



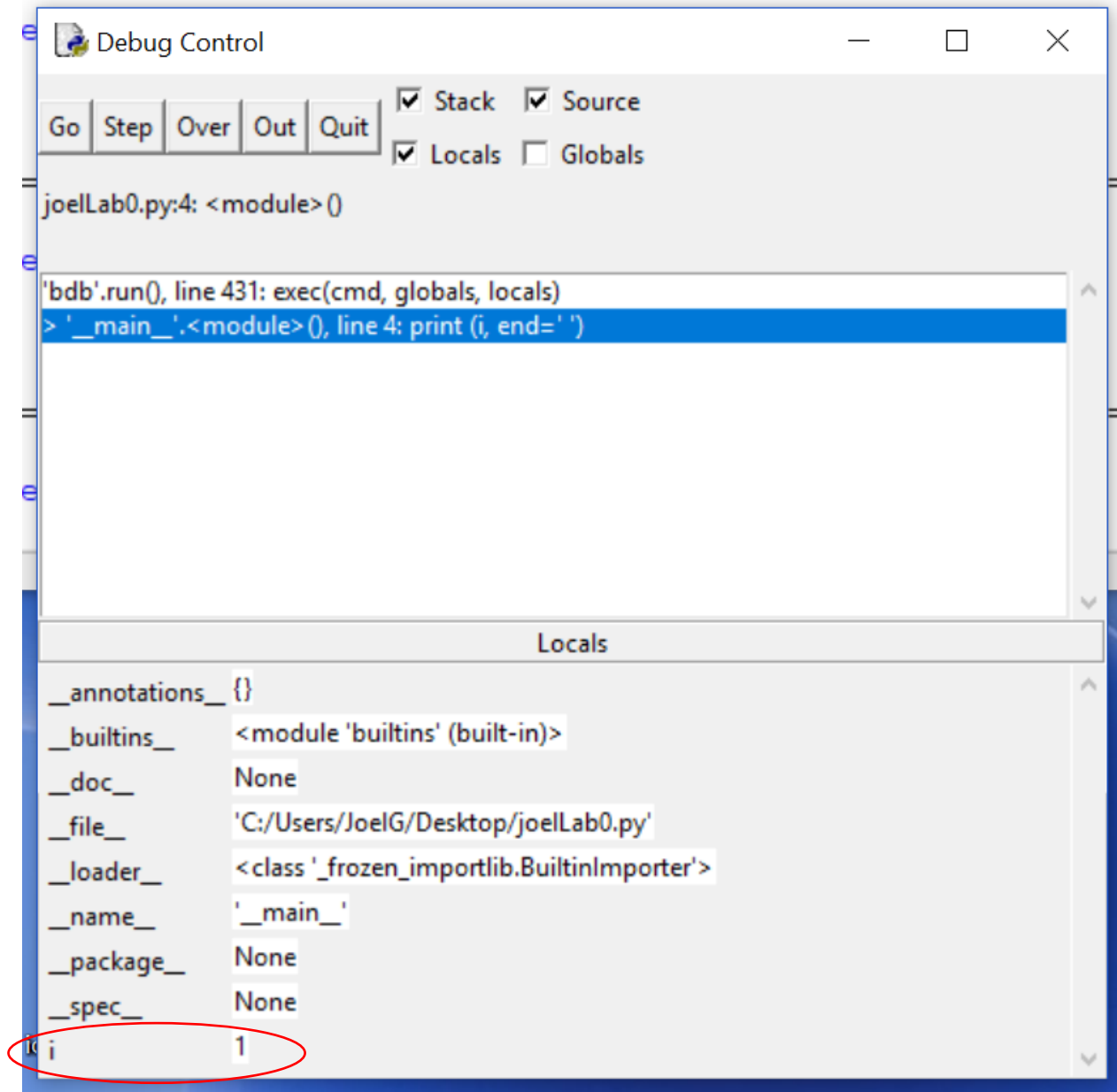
Next, the grey highlight moves down one line in the Program window, indicating the Python is ready to execute the next line:



```
joellab0.py - C:/Users/JoelG/Desktop/joellab0.py (3.6.4)
File Edit Format Run Options Window Help
print ("hello world")
print ("Here are the 10 numbers from 0 to 9")
for i in range(10):
    print (i, end=' ')
print ("I'm done")
```

Keep clicking on “Over” repeatedly, and observe what happens. If you like, you can also try clicking on “Go” or “Out”. The “Step” button isn’t that useful to use yet, though.

At some point, you may notice that the Debug-Control window has a new line on the bottom:



It's telling you that the variable "i" has a value of 1 at this point. If we've not covered this in class yet, don't worry: we will shortly!

That's it for our short IDLE tutorial. Feel free to play around with the other menu items in the various IDLE windows. IDLE is an extremely simple debugger that comes free with Python; most commercial debuggers are substantially more complex (but more powerful as well). We'll be using IDLE for most of our debugging in this course.