Setting breakpoints

WE'LL COVER THE FOLLOWING

A breakpoint is a line in the code where you want to pause execution. You can set a breakpoint by calling the **break** (or **b**) command followed by a space and the line number that you want to break on. You can also prefix the line number with a filename and colon to specify a breakpoint in a different file. The break command also allows you to set a breakpoint with the **function** argument. There is also a **tbreak** command which will set a temporary breakpoint which is automatically removed when it gets hit.

Here's an example:

```
c:\py101>python -m pdb debug_test.py
> c:\py101\debug_test.py(4)<module>()
-> def doubler(a):
(Pdb) break 6
Breakpoint 1 at c:\py101\debug_test.py:6
(Pdb) c
> c:\py101\debug_test.py(6)doubler()
-> result = a*2
```

We start the debugger and then tell it to set a breakpoint on line 6. Then we continue and it stops on line 6 just like it's supposed to. Now would be a good time to check the argument list to see if it's what you expect. Give it a try by typing **args** now. Then do another **continue** and another **args** to see how it changed.

Wrapping Up

There are a lot of other commands that you can use in the debugger. I recommend reading the documentation to learn about the others. However, at this point you should be able to use the debugger effectively to debug your own code.