
Bonus Reading Lecture

Using the PyCharm Debugger

Eric V. Level

Graduate Programs in Software

University of St. Thomas, St. Paul, MN

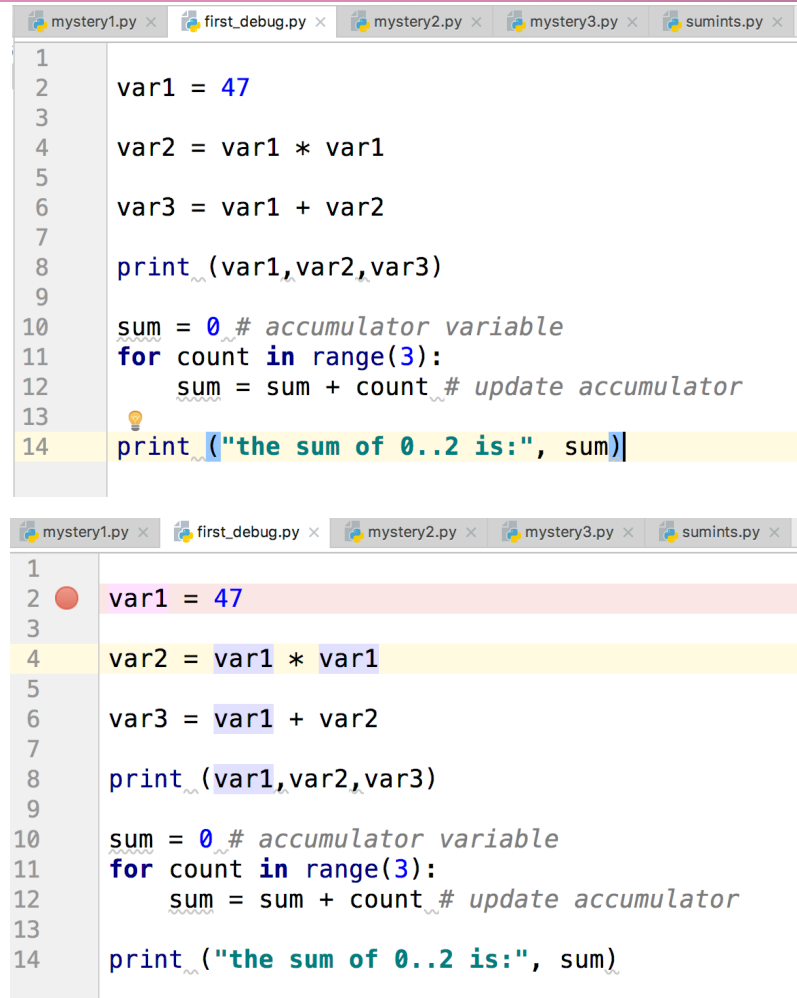
1 – Preparing to Debug

(c) copyright 2018, 47 software inc :-)

View the .py file to debug:

- Set a breakpoint on some statement by clicking to the left of it.

A red dot will appear next to this statement.



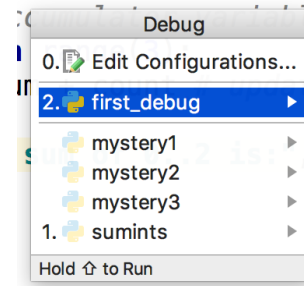
```
mystery1.py x first_debug.py x mystery2.py x mystery3.py x sumints.py x
1
2 var1 = 47
3
4 var2 = var1 * var1
5
6 var3 = var1 + var2
7
8 print(var1,var2,var3)
9
10 sum = 0 # accumulator variable
11 for count in range(3):
12     sum = sum + count # update accumulator
13
14 print("the sum of 0..2 is:", sum)
```

```
mystery1.py x first_debug.py x mystery2.py x mystery3.py x sumints.py x
1
2 var1 = 47
3
4 var2 = var1 * var1
5
6 var3 = var1 + var2
7
8 print(var1,var2,var3)
9
10 sum = 0 # accumulator variable
11 for count in range(3):
12     sum = sum + count # update accumulator
13
14 print("the sum of 0..2 is:", sum)
```

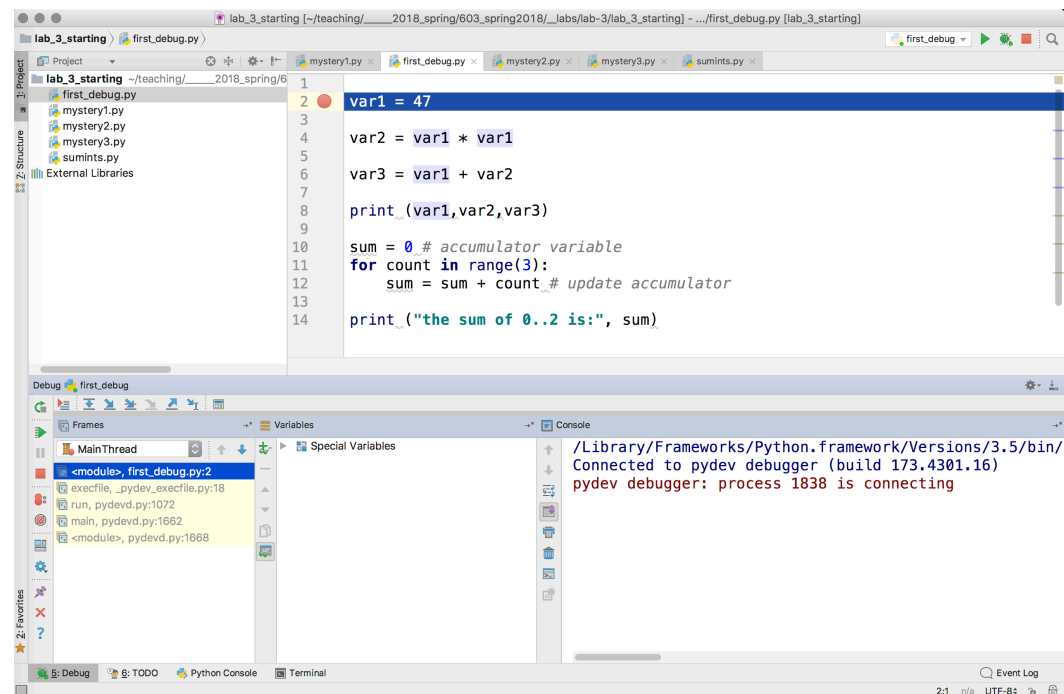
2 – Starting the Debugger

(c) copyright 2018, 47 software inc :-)

Select Run->Debug . . .
and select the file:



PyCharm displays the
Debugger "dashboard"
and executes the
program statements,
stopping just before
the set breakpoint:



3 – The Debugging Dashboard

(c) copyright 2018, 47 software inc :-)

The screenshot displays the PyCharm IDE interface with the following components and annotations:

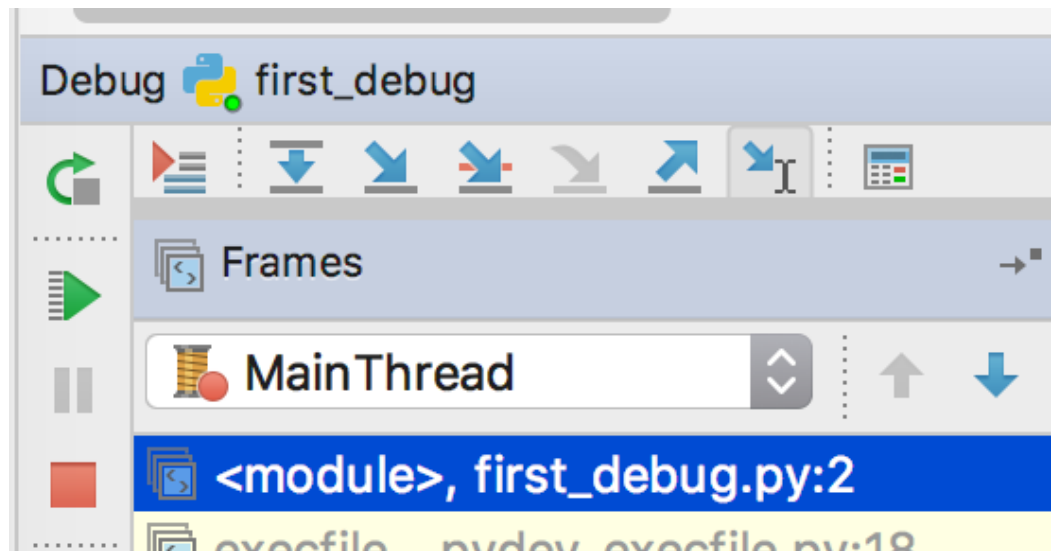
- Project Structure:** Located on the left, it shows the project 'lab_3_starting' with files 'first_debug.py', 'mystery1.py', 'mystery2.py', 'mystery3.py', and 'sumints.py'.
- Code Editor:** The main window shows the code in 'first_debug.py'. A red dot on line 2 indicates the 'Active Breakpoint'. Line 6 is highlighted in blue, labeled as the 'Next statement to execute (highlighted)'. The code includes:

```
1 var1 = 47 var1: 47
2
3
4 var2 = var1 * var1 var2: 2209
5
6 var3 = var1 + var2
7
8 print(var1, var2, var3)
9
10 sum = 0 # accumulator variable
11 for count in range(3):
12     sum = sum + count # update accumulator
13
14 print("the sum of 0..2 is:", sum)
```
- Debug Console:** At the bottom, it contains three panels:
 - Stack Frames:** Shows the call stack with the active frame being '<module>, first_debug.py:6'.
 - Active Variables:** Displays the current state of variables: 'var1 = {int} 47' and 'var2 = {int} 2209'.
 - Output Console:** Shows the output of the program, including the path to the Python framework and the message 'pydev debugger: process 1838 is connecting'.

4 – The Debugging "Palette"

(c) copyright 2018, 47 software inc :-)

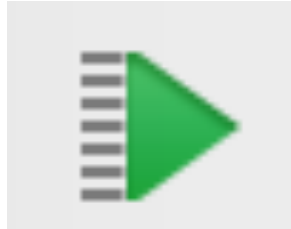
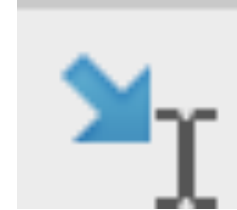
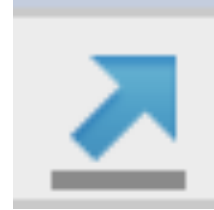
- This is the collection of controls in the lower left area.
- It allows you to select what to do next:



- Put your cursor over any of them to view a pop-up text description, along with a keyboard shortcut (not shown).

5 – Palette Commands

(c) copyright 2018, 47 software inc :-)



6 – Step Over

(c) copyright 2018, 47 software inc :-)



7 – Step Into

(c) copyright 2018, 47 software inc :-)



8 – Step Into My Code

(c) copyright 2018, 47 software inc :-)



9 – Step Out

(c) copyright 2018, 47 software inc :-)



10 – Run to Cursor

(c) copyright 2018, 47 software inc :-)



11 – Evaluate Expression

(c) copyright 2018, 47 software inc :-)



12 – Resume Program

(c) copyright 2018, 47 software inc :-)

