

Pythonic Object-Oriented Programming

Welcome

I'm your host, Aaron Maxwell.

- Author of **Powerful Python**
- aaron@powerfulpython.com
- On twitter:
 - @powerfulpython (professional)
 - @redsymbol (personal)



Our focus in this class: Understanding Python's object model... and the object-oriented concepts you need to get the most out of it.

How we will proceed

Download courseware ZIP:

- Click on the green "Resource List" icon
- Or fetch from: <https://powerfulpython.com/courseware-oop.zip>

What's included:

- RESOURCES.txt with the link to your course book
- Slides
- Labs (i.e., programming exercises - more on that later)

Give you a break every hour (10 minutes).

Give me a thumbs up. (Let's try it now)

Ask questions anytime.

Python versions

In this class, we focus on Python 3.

Python 3.6 is **required**. We will not talk about earlier versions in this class. (There are just too many differences.)

I expect you have Python 3.6 installed on your system, so you can do the programming exercises.

If you use Python 2.7 at work: read PYTHON2.txt after you've completed the class.

What makes perfect?

Practice, practice, practice.

- Practice syntax (typing things in)
- Practice programming (higher-level labs)

I expect you to do your part!

You **exponentially** get out of this what you put into it.

GO FOR IT.

Running the labs

Labs are the main programming exercises. You are given a failing automated test; your job is to write Python code to make it pass.

Simply run it as a Python program, any way you like. (For example, `"python3 helloworld.py"`)

Run unmodified first, so you can see the failure report.

When done, click the thumb's up, and find someone to high-five.

Then: Move on to the extra credit.

Lab: helloworld.py

Let's do our first lab now: 'helloworld.py'

- In labs folder in the courseware

Instructions are in LABS.txt in the courseware.

When you finish: Give me a HIGH FIVE! in the chat room, and click Thumbs up, so I know you're done.

You'll know the tests pass when you see:

```
*** ALL TESTS PASS ***  
Give someone a HIGH FIVE!
```

THEN: Start browsing through the "Python 3 Object-oriented Programming" book. Look it up under the green "Resource List" icon, or find a direct link in RESOURCES.txt.

Getting the most

We'll take some class time for each lab. You may not finish, but it's **critically important** that you at least start when I tell you to.

After we're done for the day, find time to finish all the main labs before tomorrow.

Solutions are provided. Use them wisely, not foolishly:

- After you get the lab passing, compare your code to the official solution.
- Other than that, don't look at them if you can avoid it.
- The more work you can do on your own, the more you will learn. Peek at the solution to get a hint when you really need it.

Optional (**only** for future master Pythonistas): Do all the extra labs as well, as soon as you can manage.

Python Trainings

- **Scaling Python with Generators:** Write robust, reliable code to gracefully handle increasing amounts of data
- **Pythonic Object-Oriented Programming:** Master the principles of OOP, in Python
- **Pythonic Design Patterns:** Understand higher-level design patterns. Pythonically
- **Test-Driven Development In Python:** Master automated testing. It's a *superpower*.
- **Beyond Python Scripts: Logging, Modules, & Dependency Management:** For larger Python applications. Part 1
- **Beyond Python Scripts: Exceptions, Error Handling and Command-Line Interfaces:** Continued with Part 2
- **Python - The Next Level:** See into the Matrix of Python. Master facets of this language that 99% of Python developers will *never* learn

For detailed course descriptions and sign-up information:

<https://powerfulpython.com/safari-trainings/>

(Select "Python Trainings" under the green "Resource List" icon.)