[**How should I start learning Python?**](https://www.quora.com/How-should-I-start-learning-Python-1)

[](https://www.quora.com/profile/Neha-Ahuja-178)

[Neha Ahuja](https://www.quora.com/profile/Neha-Ahuja-178), studied Bachelor of Engineering in Computer Engineering at Savitribai Phule Pune University

[Updated Sep 23, 2019](https://www.quora.com/How-should-I-start-learning-Python-1/answer/Neha-Ahuja-178) · Upvoted by [Kratu Nandan](https://www.quora.com/profile/Kratu-Nandan), Programmer at Software and Applications (2000-present) and [Arvind Patil](https://www.quora.com/profile/Arvind-Patil-68), M.Tech Computer Science & Python, REVA University (2019)

Rather than giving you a boring **step** by **step** process of learning **Python**, I would share my personal journey about how I started learning **Python**.

Here is my personal learning experience:

**What motivated me to start learn Python ?**

I fell in love with **Python** after reading a bunch of answers on Quora about how people were doing wonderful things with **Python**.

Some were writing scripts to automate their Whats app messages.

Some wrote a script to download their favourite songs,

while some built a system to receive cricket score updates on their phones.

All of this seemed very excited to me and I finally decided that I would love to learn **Python**.

**How I started learning Python ?**

I started learning **Python** from The Complete **Python** Masterclass: Learn **Python** From Scratch as it was the recommended course for beginners.

I found the course to be really interesting yet simple for beginners like me.

However, if you are a professional programmer then I would recommend you to learn from the official docs : 3.7.2 Documentation

**How much time it took to learn ?**

**Topic:** **Python** basics, control structures, functions.

**Time :**- 1 Week.

**Learning experience:**

Learning **Python** basics is a piece of cake, it is extremely simple to get up and running with **Python**. Basics like variables, operators and control structures are extremely easy to learn as opposed to other languages like Java.

**Topic**: OOP & Regular Expressions

**Time:** 1 Week.

**Learning Experience:**

This section was a bit tough as I didn’t had a clear understanding of OOP principles. I had **used** OOP in Java but still was not clear about the concept but eventually with some practice I was able to understand OOP in **Python**, the self keyword and the init method.

If you are a beginner then I would recommend you to focus on this section as it is widely **used** while making complex applications.

You might not understand a lot of things at first, but give things time to sink in and it will make sense.

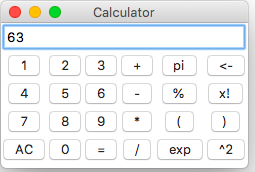
**Topic:**Tkinter

**Time:**1 Week

Learning experience:

This was the most exciting as I finally learned how to build desktop GUI applications **using Python**. Learned about the Tkinter library which allowed us to build interactive GUI with **Python**. It felt as if everything I have learned till now started making sense. When I was finally complete with this section, I was able to build my very own calculator.

Here is a screenshot of what I built:



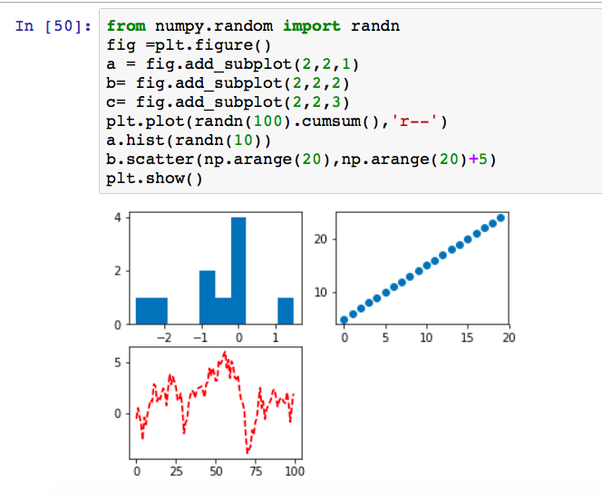
**Topic:**Data Analysis

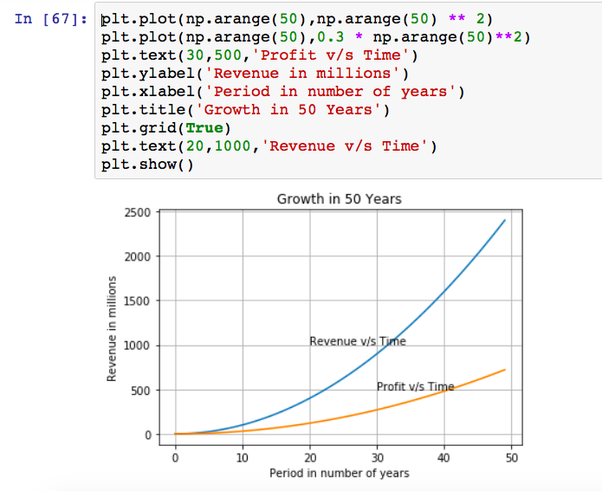
**Time:**1 Week

**Learning experience:**

Learned about the Numpy and Pandas library which are extensively **used** to perform data analysis with **Python**. If you aspire to be a data analyst/ data scientist I say you focus on learning the basics well. I also learned how to read data from an excel/ CSV file and visualise the same data on a chart. In the process, I also learned about Jupyter notebooks which is an excellent IDE for data visualisation with **Python**.

Here is a screenshot of me plotting some graphs:





**Topic:** Django

**Time:** 2 Weeks

**Learning experience:**

This was by far the most difficult topics I had ever learned. I had no previous experience learning anything related to server side web **development** and hence it took me a while to understand terminologies like authentication, URL routing, API and models.

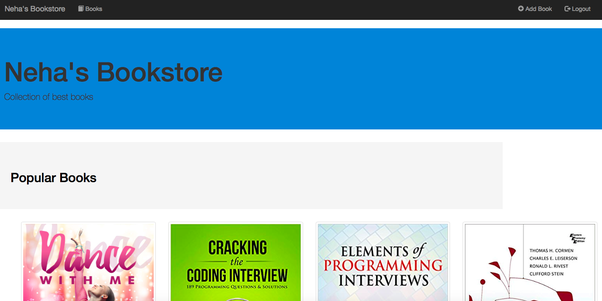
I had initially given up on this section as it felt very complex but I took my time and worked my way through it with patience. I took notes on pen and paper and made sure that I understand every line of code even before I **use** it.

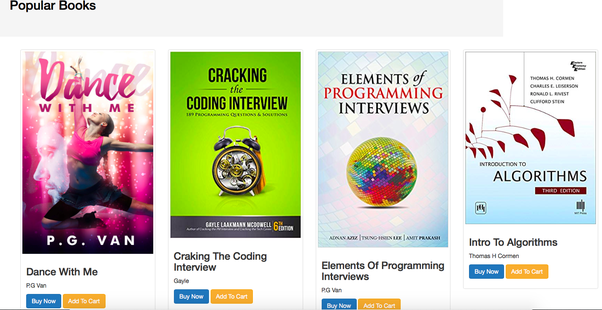
Eventually the efforts paid off and I finally started to understand what Django really is and how it works.

It took me 2 weeks to get done with the Django basics and a lot more practice to finally complete a Django project on my own.

I built my own todo app and a simple book store site **using** Django and a little bit of Bootstrap.

Here is a screenshot of the bookstore I built:





**Topic:** Flask & Web Crawler

**Time:** 2 Weeks.

Flask was a piece of cake after learning Django as Flask is just a micro framework.

I didn’t personally dabble much with Flask as I liked and preferred Django over Flask.

Building a web crawler was interesting as well, I built a crawler which was **used** by search engines back in the early days to crawl web pages, find links and store those links in a file.

**Topic:**Automation with selenium

**Time:** 1 Week.

After learning almost everything about **Python** selenium felt pretty easy.

Selenium is originally **used** for testing purposes but it turns out that you can automate various tasks with it. I learned how to build a Facebook auto poster with selenium.

It took me around 9–10 weeks to complete and I was happy with the process and eventually learned a lot about how **Python** works.

Note that I was dedicating 2–3 hours every day for learning.

**Final conclusion:**

It was a great experience learning **Python** and I now feel a lot more confident as a programmer. The main source of confidence was being able to build something of my own, however small it may be.

Although my learning process might seem easy but there was a lot of struggle and moments of frustration in between.

**Here are a few of my tips if you want to start learning Python:**

1. Have patience, if you are a beginner it will take time to let things sink in.
2. Aim at building small tools/projects, don’t just learn the syntax and jump to a new programming language.
3. Frustration and pain is a part of learning process, embrace it instead of avoiding it.
4. Errors/ issues are expected, don’t let that discourage you from learning.
5. Be consistent, if you are not consistent in learning it might take a lot more time and effort.

Happy learning !