lesson 01

Introduction to Python: Python in Finance

Python for Financial Analysis

Rajah Chacko

elvtr

Syllabus Review

1. **Introduction to Python: Python in Finance**
2. Python Basic Syntax: Importing Libraries
3. Working with Pandas
4. Pandas Underneath the Hood: Working with NumPy
5. Data Wrangling and Visualization
6. Extracting Financial Insights from Charts and Graphs
7. Financial Calculations with Python: Part 1
8. Financial Calculations with Python: Part 2
9. CAPM and Portfolio Management
10. Linear Regression
11. Time Series Analysis
12. Algorithmic Trading

Bonus Class: Cryptocurrency Beyond the Basics with a Fintech Guest Speaker

About the instructor

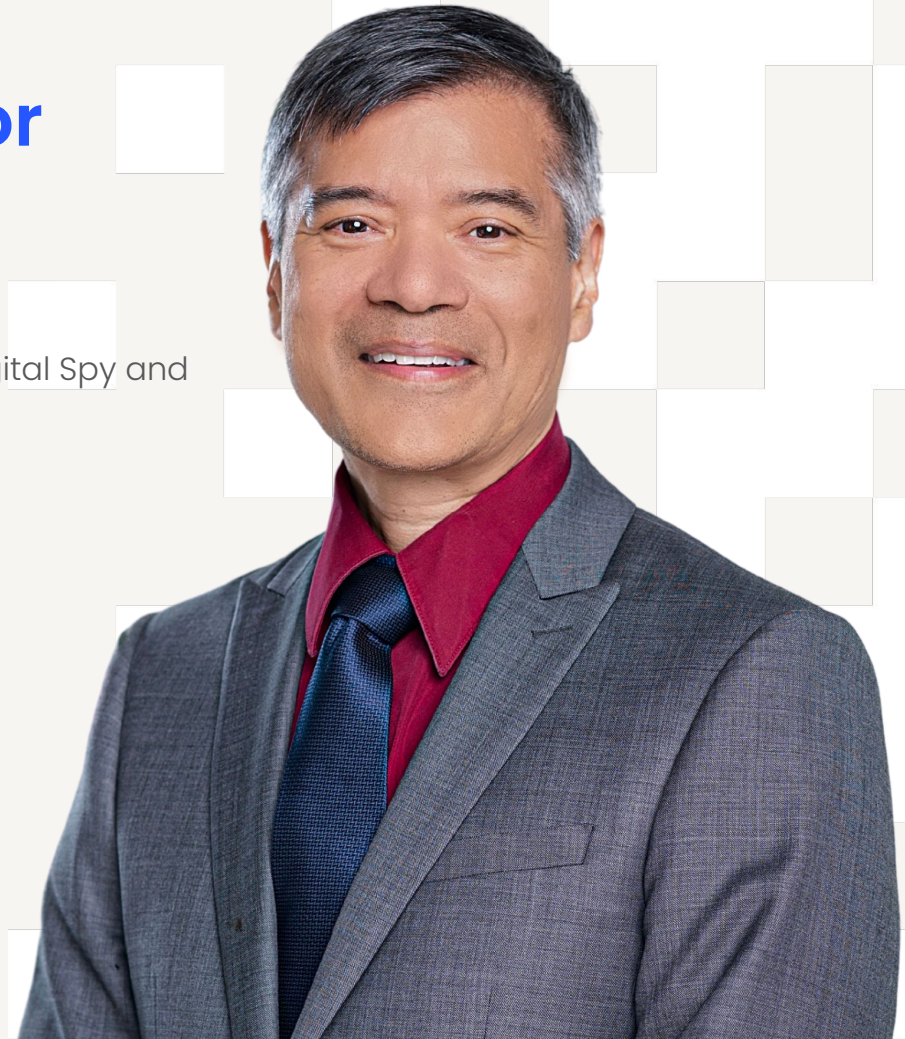
My educational background

My work background (and latest jobs as Digital Spy and Quantitative Analyst)

Where I live

Linkedin : send me a note

A personal note



Course Structure

- Mon & Wed, 4:30 PM PST / 7:30 PM EST
  - 1 hour of instruction
  - 15 minutes for Q&A
- One assignment each lesson
  - 100 total points, 80 needed to receive course certificate
- Weekly office hours on Thursdays 4:30 PM PST / 7:30 PM EST
- Discord channel for asynchronous communication and collaboration with fellow students

Class agenda

- Python: General overview
- What makes Python a great tool for financial specialists
- Python vs. Excel and Python 2.x for Finance
- Anaconda and Jupyter notebook: Interface overview
- Data types in Python

Intro to the course

Assignments
The challenge
Grading policy

Eleven assignments, eleven (optional) Python assignments that give bonus points.
Final project, 20 points.
80 points to receive a certificate

Module	Main Assignment	Optional Python
1	6	2
2 through 11	8	2
final project	15	n/a
total	100	22

Python (a gentle introduction)

From: https://www.python.org/doc/essays/blurb/

What is Python? Executive Summary Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy-to-learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Guido van Rossum, BDFL

Installing all the things

- Anaconda

  https://www.geeksforgeeks.org/how-to-install-jupyter-notebook-in-windows/

  https://www.geeksforgeeks.org/how-to-install-anaconda-on-windows/

  Some cool things you can do with a Jupyter notebook

- PyCharm

  https://www.jetbrains.com/pycharm/download/

Umm… which should I choose?

Why Python 3?

(Why not Excel?)

(Why not Python 2.x?)

Where to get help

My favorites are stackoverflow.com and https://www.geeksforgeeks.org/. And yes, a Google search for “python what is my version” will greatly help in the first question of the Module 1 assignment.

The site https://realpython.com/ is a good go-to resource.

Hands on Python

Assignment #1

part 1

Install the Python IDE of your choice. Then answer these questions. Create a PDF from your notebook (or screenshot if you're using PyCharm).

1. What versions?
   a. What version of Python are you running? (You can do this from the Anaconda command prompt, or you can do it in Python).
   b. What version of Pandas and NumPy do you have? (You can use the pip command in the Anaconda command prompt or the PyCharm terminal prompt. You can also write programs to do this in Python with a little research.)
2. Ints and Floats – Compound interest – If you invested $1000 (one-time) in a tax-free account that got 8% each year, how much would you have after 40 years?
3. (No datetime yet; We'll defer that to Assignment 2).
4. Booleans – Create two variables, cat_count and dog_count. From those variables, create two boolean variables, has_cats and has_dogs. These will be true if the corresponding count is greater than 0.
   a. Let's say you have a happy home if you have at least one cat (but no dogs). Write a boolean expression that prints True if you have a happy home (but False otherwise).
   b. Let's say you have a happy home if you have at least one cat (but no dogs), or at least one dog (but no cats). Write a boolean expression that prints True if you have a happy home (but False otherwise).

Assignment #1

part 2

5 Lists – Create a list of the integers from 1 to 10. Remove all the non-primes (including 1). Create a second list of primes between 10 and 20.
   a. Print the results when you append the list over 10 to the list under 10.
   b. Print the results when you extend the list over 10 to the list under 10.

6 Dictionaries – Build a dictionary similar to the example in the lesson. You may use d1 = {'USD': 1.00, 'BTC': 51013.93, 'EUR': 1.131735}
   c. Add a currency of your choice. (Hint: Web search for "swiss franc to dollar")
   d. Given a FROM and a TO currency, look up in your dictionary and calculate the exchange rate for the pair
   e. What happens if the currency is not in your dictionary?

7 Tuples – Create two points as tuples and calculate their magnitude. (Magnitude is the square root of the x ** 2 + y ** 2.)
   f. You will need to add this as the top line of your cell: from math import sqrt

Take-home Assignment #1 (optional)

Find a challenging problem on one of these three online platforms. Place a link to the question and answer in the IDE of your choice.

1. Go to GeekForGeeks (https://www.geeksforgeeks.org/python-programming-examples/), sign up for free, and do five challenging exercises from each of the Basic, String, Dictionary, Tuple, and Date-Time categories. Please use both their test data and your own.
2. Go to HackerRank (https://www.hackerrank.com/domains/python), sign up, and Solve a Challenge (which is challenging for you). If you run the code in HackerRank, make sure you use Python 3.
3. Go to StackOverflow (https://stackoverflow.com/) and search for “Python puzzle” or “Python interview.” Find a challenging puzzle or interview question, glance at the solution (don’t copy and paste it) and develop your own solution to the problem.

Resources

- Python executive summary: https://www.python.org/doc/essays/blurb/
- A comparison of Python with other languages: https://www.python.org/doc/essays/comparisons/
- Guido van Rossum, BDFL. Home page https://gvanrossum.github.io/
- Install Anaconda on Windows: https://www.geeksforgeeks.org/how-to-install-anaconda-on-windows/
- Install PyCharm: https://www.jetbrains.com/pycharm/download/
- Other help: GeekForGeeks.org, hackerrank.com, stackoverflow.com, realpython.com

Q&A