Geometric Progression

We Bring AI to Reality.

Fast-Track Python Programming:

With a focus on Financial Markets & Business applications

Overview

Welcome to our Python course, a transformative programme designed to equip you with the Python programming skills essential for today's business world starting from scratch. Python is now incorporated into Excel and is becoming an indispensable tool for financial analysis & modelling. This course addresses the urgent need for proficiency in this versatile language.

Our curriculum is meticulously tailored to fast-track your learning, blending core Python programming skills with specific applications in business and the financial markets. From understanding basic data types and control structures to mastering data manipulation with Numpy and Pandas.

What sets our course apart is not just the content but the format. We combine the best of in-person and online learning, ensuring you're not just passively watching pre-recorded lectures but actively engaging with instructors and peers. This interactive approach is invaluable, especially when starting out, as it provides immediate feedback, personalized support, and the dynamic exchange of ideas and questions with fellow participants. The course will also be recorded to get the best of both experiences.

Moreover, we incorporate the use of advanced AI tools like ChatGPT and Gemini to accelerate your learning process. In a landscape where Python is the new Excel, this course ensures you're not left behind.

Additional Features:

- Choose from a host of optional modules.
- Includes Lifetime access to comprehensive electronic course materials, including insightful videos and a curated list of resources for deeper exploration.
- Cheat Sheets.
- Recorded sessions for future reference.

Please refer to the detailed agenda below for more information.

Level	Introduction to Intermediate.
Duration	Customized to your needs.
Requirements	Computer with internet access. Contact us for files to download and setup.
Suitable For	Anyone who wants to learn to program, automate tasks, or learn Python.
Not Suitable For	Anyone who does not understand the basics of using a computer.

Tutor: Mark Raffaelli
Mark is obtained his CFA Charter in 2000 and became fellow member of the Global Association of Risk Professionals (FRM) in 2001. Mark's extensive experience includes:

- Trading in Spot & Derivative Products professionally.
- Development of quantitative financial models for Surveillance, Performance Attribution, Price Validation, Price Models, Risk and Automation.
- Developments of Apps for the investment and insurance industry.
- Machine Learning and Deep Learning with Sklearn, Tensorflow & Pytorch in the Financial Markets for Banks, Asset Managers and general business.
- Time Series Analysis, Regime Change & Optimization with and without AI.
- Strategic implementation of AI.

Those who have attended Mark's courses will know about his passion and ability to cut through jargon, simplify technical issues and provide real life examples.

Detailed Agenda

1. The stuff you need to know before we start.

Delegates will be able to:

- Understand why it is so important to know how to Program in today's world.
- Distinguish between Modelling versus professional programming.
- Navigate and efficiently use Jupyter notebooks.
- Create and use notebooks.
- Create comments in notebook cells.
- Create markup cells to create a rich user experience incorporating markup language for headings, paragraphs, images, links etc.

2. Datatypes

Delegates will be able to create, use and explain of the different types of datatypes. Delegates will be able to:

- Use the print statement and the type statement.
- Create a string and output string.
- Output a string using a host of different string formatting methods.
- Create, use and explain integers.
- Create, use and explain floating type numbers.
- Create code to cast from one datatype to another.
- Create and use variables.

3. The world of Functions

Delegates will be able to:

- Create a function.
- Understand and correctly indent code to make the function work correctly.
- Understand and use white-space appropriately.
- Define the function Signature.
- Create arguments and parameters and understand the difference.
- Create and use “docstrings”.
- Create user-defined functions.
- Create, use and understand default arguments and optional arguments.
- Create and implement python built in functions.

4. Scope

Delegates will be able to:

- Understand and define the difference between local and global scope.
- Understand the concept of enclosing scope and variable clash.

5. Conditional logic

Delegates will be able to:

- Create an if statement and use the else-if and else Logic.
- Put together some nested if statements.
- Understand the concept of XOR

6. Loops

Delegates will be able to:

- Define, create and use the range function.
- Create a for loop and understand the logic of how the code is processed.
- Create a while loop and understand the concept of incrementation.
- Create nested loops.

7.Useful Loop Functions

Delegates will be able to:

- Solve algorithmic problems using flags.
- Implement the “break” out of a loop code.
- Implement the “continue” loop code.
- Use and understand the “pass” place-holder.
- Understand the concept of the “iter” method and implement it.

8. Lists

Delegates will be able to:

- Use and create lists.
- Slice a list.
- Append, delete and extend lists.
- Pop an item off a list.
- Loop through items of a list.
- Implement list functions.

9. Sets & Tuples

Delegates will be able to:

- Create a set and implement set operations.
- Use a set to derive unique items.
- Use set mathematics to filter data.
- Create a tuple.
- Implement tuple functions.

10. Dictionaries

Delegates will be able to:

- Create, use and develop a dictionary.
- Use economic stats to fill a dictionary.
- Slice a dictionary.
- Implement a dictionary and list comprehension.
- Loop through the dictionary.

11. Modules

Delegates will be able to:

- Understand and perform a pip install.
- Import external modules.
- Import self-created modules.
- Create a “namespace”.
- Import modules with wild cards.
- Create a module.

12. Visual Studio Code (or equivalent)

Delegates will be able to:

- Independently set up the Visual Studio Code environment.
- Download a python extension.
- Select a python interpreter.
- Implement debugging such as step through, step over and break points.

14. Object Orientated Programming (OOP)

Delegates will be able to:

- Explain the difference between procedural code and object orientated code.
- Create a class.
- Encapsulate functions inside a class.
- Create an object constructor.
- Create an object instance.
- Call an instance attributes and methods.
- Implement dunder methods.
- Using “self” in code.
- Build their own OOP Calculator.

15. Functional Programming, Lambda Functions and the @ Decorator

Delegates will be able to:

- Understand and use a first class function.
- Pass a function as an argument.
- Create a lambda function.
- Implement the “filter” statement on a lambda function.
- Implement the “map” statement on a lambda function.
- Implement a “reduce” statement on a lambda function.

16. Getting a Chatbot to Assist us with Programming

To Accelerate the course going forward, we will program with an AI assistant such as ChatGPT, Google Gemini, Anthropic or any other chatbot that your organization uses. As a result, we will spend a brief amount of time to ensure that all users can use the chatbot efficiently for programming purposes.

17. Wrapping up with Environments, Zipping, widgets, decorators and Excel.

Delegates will be able to:

- Use Jupyter widgets.
- Use the Python Zip function.
- Understand and use @Decorators.
- Implement and use the Datetime module.
- Implement and use the DateUtil module.
- Use PysimpleGUI to create a graphical user interface.
- Create an exe using pyinstaller.
- Setting up a Python Environment.
- Controlling Excel (office) with built in Python.

Core Python Libraries for Analysts or anyone doing Analysis / Analytics.

18. Numpy.

Delegates will be able to:

- Convert different datatypes to numpy arrays.
- Measure Array shapes, dimensions and size.
- Use vectors in numpy.
- Numpy math.
- Portray images as vectors.
- Reshape, ravel and flatten arrays.
- Implement Numpy functions and seeding.

19. Pandas.

Delegates will be able to:

- Convert datatypes to a pandas series
- Create, update and delete a series.
- Implement Series functions and filter a series.
- Convert datatypes to a pandas dataframe.
- Create, update and delete a dataframe.
- Implement dataframe functions.
- Distinguish between the commands “display vs print”.
- Implement different reference conventions (loc and iloc etc).
- Perform selection, filtering and assignment with loc, iloc, at,iat.

20. Intermediate Pandas.

Delegates will be able to:

- Loop through big data with Pandas.
- Use superior alternatives to loops with referencing with Pandas.
- Import data from, csv, Excel and pdf into Pandas.
- Perform HTML Webscraping with Pandas.
- Create code for remote access of data with Pandas
- Implement Pandas TimeSeries and datetime.
- Build an algo trader and backtest the results.
- Perform data interpolation with pandas.
- Output to Excel with pandas.

20. Plotting Charts with Python.

Delegates will be able to:

- Chart with Matplotlib.
- Plot points, scatterplots and a line.
- Add some labels and features to the graph with code.
- Plot various charts with code.
- Create multiple plots with subplots.
- Create variations of subplots.
- Add axes manually in code.
- Use formulas to plot with numpy
- Create 3D Plots.
- Rotate the plots and Azimuth's.
- Create quick plots from pandas to matplotlib

21. An introduction Portfolios and Algo Trading with Python

- PyPortfolio and other Portfolio libraries.
    - Optimizing a portfolio.
- Algo trading and Algo trading libraries.
    - A simple Algo Trade
- Valuation libraries.

Optional Modules

Please note that all the modules listed below have detailed agendas and can be customized to your specific needs. Please contact us for more information or customization. The focus of many the applications will be centered around portfolios, trading, risk and financial markets unless directed otherwise.

- **Automation.**
- **Webscraping.**
- **Mathematics with Python.**
- **Statistics with Python.**
- **Optimization with Python.**
- **The Art of Modelling.**
- **Machine Learning with SKLearn.**
- **Deep Learning with Tensorflow.**
- **Deep Learning with Pytorch.**
- **Time Series Analysis with Python.**
- **Time Series Analysis with AI.**
- **Markov Chains with Python.**
- **Reinforcement Learning.**
- **Anomaly detection**
- **Regime Change.**
- **Algo Trading**
- **Portfolio Management.**
- **A more detailed look of Python and Excel.**