



PYTHON FOR FINANCE

COURSE GUIDE (V1): FEBRUARY – JULY 2021

© 2021 PYTHON CHARMERS®

Python for Finance

A specialist course

Audience: This is a course for financial analysts, traders, risk analysts, fund managers, quants, data scientists, statisticians, and software developers.

Context: Python is an important language in the financial services industry, useful in both analysis (modelling) and production systems.

Overview: This intensive, hands-on, practical training course will teach you how to apply powerful Python-based tools for processing, analyzing, modelling, and visualizing various kinds of data, with a focus on financial time-series.

Format: Live instructor-led training (online). Each topic is a mixture of expert instruction, worked examples, and hands-on exercises with help from the instructor(s).

Expert instructors: See bios below.

Duration: 5 days

Modules:

Days 1–5: “Python for Finance”
Days 1–4: “Python for Predictive Data Analytics”
Days 1–2: “Introduction to Python”
Days 3–5: for graduates of other Python Charmers courses or people with 3–6 months Python experience

Price:

Regular course (5 days): AUD \$3,500 (excl GST)
Modular / partial courses: AUD \$800 (excl GST) per day

Dates (February – July 2021):

15–19 March
19–23 April
24–28 May
28 June–2 July



Skills & Activities

Skills

Overall: You will gain a broad understanding of methods and tools in modern data science and come to appreciate the elegance and power of the Python language and its powerful ecosystem of packages.

Days 1–2: You will learn the fundamentals of Python’s powerful data types and how to manipulate tabular data with ease in a variety of formats, including SQL databases, CSV, Excel spreadsheets, JSON, API endpoints, and time-series.

Days 3–4: You will learn how to use machine learning to solve problems involving clustering, classification and nonlinear regression; and how to refine, evaluate and deploy machine learning models in production.

Day 5: You will learn in depth about munging, modelling, and visualization of financial time-series data.

Activities

Exercises: There will be practical exercises throughout the training course. These will be challenging and fun, and the solutions will be discussed after each exercise and provided as source code. During the exercises, the trainer will offer help and suggestions.

Worked examples: To prepare you for the exercises, the trainer will present worked examples and demos and help you to follow along on your own computer.



Topic outline

Day 1: Python basics

Day 1 covers how to use Python for basic scripting and automation tasks, including tips and tricks for making this easy:

- Why Python? What's possible?
- The *Jupyter* notebook for rapid prototyping
- Modules and packages
- Python concepts: an introduction through examples
- Essential data types: strings, tuples, lists, dicts
- Worked example: retrieving real-time data from a REST web API
- Raising and handling exceptions

Topic outline

Day 2: Handling, analyzing, and presenting data in Python

Python offers amazingly productive tools like Pandas for working with different kinds of data. Day 2 gives a thorough introduction to analyzing and visualizing data easily:

- Reading and writing essential data formats:
CSV, Excel, SQL, time-series (others on request)
- Indexing and selecting data in *Pandas*
- Data fusion: joining & merging datasets
- Summarization with “group by” operations; pivot tables
- Visualization and statistical graphics with *Seaborn*
- Automated reporting; interactive dashboards *ipywidgets* and *voilà*

Topic outline

Day 3: Time-series, simulation, inference and modelling

Day 3 demonstrates more advanced features of *Pandas* for working with data, including time-series data. It then describes Monte Carlo simulation methods and walks you through using powerful methods of inference and modelling as well as clustering and outlier detection:

- Time-series analysis: parsing dates, resampling, handling time-zones, rolling-window operations
- Introduction to *NumPy* for manipulating vector and matrix data: data types, powerful indexing, reshaping, *ufuncs*
- Linear regression with *statsmodels*
- Statistics; Monte Carlo simulation; density estimation
- Clustering with *scikit-learn*
- Outlier and anomaly detection with *pyod*

Topic outline

Day 4: Machine learning

Day 4 gives you a practical and comprehensive introduction to machine learning for powerfully inferring complex models from data, with examples selected from a range of industries, including time-series and spatial datasets:

- Concepts of machine learning (ML)
- Overview of the ML package ecosystem in Python
- Nonlinear regression, with application to forecasting
- Classification with *scikit-learn*:
Application to diagnosis, AI systems, and time-series prediction
- Validation and model selection; diagnostic tools; *yellowbrick*
- Feature engineering and selection; *eli5*
- Overview of “classical” ML algorithms:
Naive Bayes, logistic regression, SVMs, random forests
- Deploying machine learning models in production

Topic outline

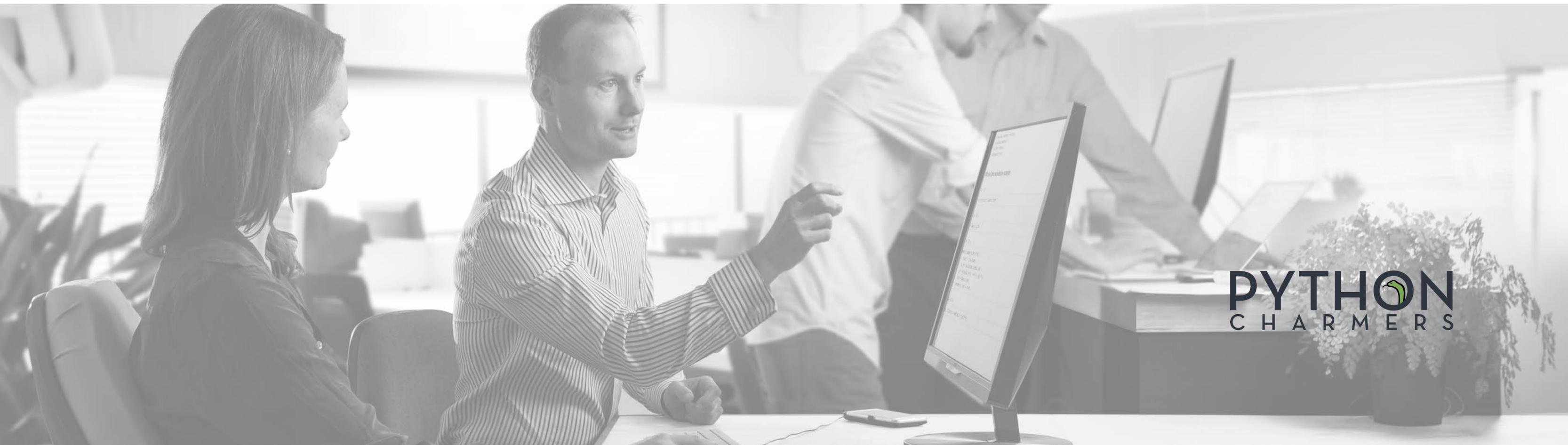
Day 5: Financial time-series in-depth

Day 5 teaches you in-depth about munging, modelling, and visualization of financial time-series data in Python:

- Reshaping of dataframes with *stack/unstack/melt*; hierarchical indices
- Categorical, string, and datetime columns; binning continuous data
- Joining of time series; *searchsorted*
- Styling dataframes for reporting; interactive use via *qgrid*
- Data cleaning; interpolation; imputing missing data
- Common time-series models in Python: *AR, ARIMA*
- Simulation examples: Monte Carlo risk analysis; Black-Scholes pricing
- (Optional): 2-way integration of *Python* with *Excel* using *xlwings*
- Visualizing time-series data interactively with *Altair*

Personal help

We are happy to offer on-the-spot problem-solving after each day of the training for you to ask one-on-one questions — whether about the course content and exercises or about specific problems you face in your work and how to solve them. If you would like us to prepare for this in advance, you are welcome to send us background info before the course.



PYTHON
CHARMERS

Other information

Format: Courses are conducted online via video meeting using Python Charmers' cloud notebook server for sharing code with the trainer(s).

Computer:

- **Hardware:** we recommend ≥ 8 GB of RAM and a webcam. Preferably also multiple screens and a quiet room (or headset mic).
- **Software:** a modern browser: Chrome, Firefox, or Safari (not IE or Edge); and Zoom.
- **Coding:** we have a cloud-based coding server that supports running code and sharing code with the trainer(s).

Timing: Most courses will run from 9:00 to roughly 17:00 (AEDT) each day, with breaks of 50 minutes for lunch and 20 minutes each for morning and afternoon tea.

Certificate of completion: We will provide you a certificate if you complete the course and successfully answer the majority of the exercise questions.

Materials: You will have access to all the course materials via the cloud server.

We will also send you a bound copy of the course notes, cheat sheets, and a USB stick containing the materials, exercise solutions, and further resources.

```
data, cmap='winter' )
```

```
python3.7/site-packages/matpl
```

```
cnis+ elementwi comparison
```

```
but the future will perfo
```

```
str('ce')
```

PYTHON
CHARMERS

Instructor bio



Dr Edward Schofield

Ed has consulted to or trained over 2500 people from dozens of organisations in Python, including ANZ, Atlassian, Barclays, Cisco, CSIRO, Dolby, Epoch Capital, Harvard University, IMC, Oracle, Shell, Singtel Optus, Telstra, Tibra, Toyota, Verizon, and Westpac. He is well-known in the Python community as a former release manager of *SciPy* and the author of the widely used *future* package. He runs leads the Python user group in Melbourne and regularly presents at conferences in data analytics and Python in Australia and internationally.

Ed holds a PhD in machine learning from Imperial College London, with a focus on generative models for sequence prediction. He also holds BA and MA (Hons) degrees in mathematics and computer science from Trinity College, University of Cambridge. He has 20+ years of experience in programming, teaching, and public speaking.



Instructor bio



Dr Robert Layton

Robert is the author of the book "Data Mining in Python", published by Packt. He provides analysis, consultancy, research and development work to businesses, primarily using Python. Robert has worked with government, financial and security sectors, in both a consultancy and academic role. He is also a Research Fellow at the Internet Commerce Security Laboratory, investigating cybercrime analytics and data-mining algorithms for attribution and profiling.

Robert is a contributor to the Python-based *scikit-learn* open source project for machine learning and writes regularly on data mining for a number of outlets. He is also the author of the website "LearningTensorflow.com". He has presented regularly at a number of international conferences in Python, data analysis, and its applications.



PYTHON
CHARMERS

Instructor bio



Henry Walshaw

Henry has over 15 years of experience in application development with Python, particularly in the natural resource management field. Henry's core technical expertise relates to the development and analysis of large scale spatial datasets and communicating this understanding to people including subject matter experts and the general public. Henry has given Python training courses to staff of dozens of organisations in Australia, including AGL, Commonwealth Bank, CSIRO, National Australia Bank, the NSW Department of Finance, Shell, and Telstra.

Before joining Python Charmers, Henry worked in both government (at GA, DSE Victoria, and the EPA) and in the private sector as Senior Spatial Consultant with Sinclair Knight Merz (SKM) and we-do-IT. He holds a Bachelors in Computational Science.



Instructor bio



Dr Ned Letcher

Ned is a data scientist and software engineer who has helped a range of organisations in projects involving machine learning, natural language processing, information retrieval, and data visualisation. Ned has been using Python for data analysis, visualisation, machine learning, and web development for over 10 years. Ned is a contributor to the Plotly *Dash* library and an active member of the *Dash* community.

Ned has a PhD in computational linguistics from the Natural Language Processing group at the University of Melbourne. He also has a Bachelor of Arts (philosophy and linguistics) and a Bachelor of Science with Honours (computer science). Ned regularly presents at local meetups and organises the Melbourne Data Visualisation Meetup.





About Python Charmers®

Python Charmers is the leading provider of Python training in the Asia-Pacific region, based in Australia and Singapore. Since 2010, Python Charmers has given over 450 training courses and bootcamps to over 4,500 delighted people from organizations such as AGL, Atlassian, Barclays, CSIRO, Cisco, Deloitte, Dolby, IMC, pwc, Singtel Optus, Shell, Sportsbet, Telstra, Toyota, Verizon, Westpac, and Woolworths. Python Charmers specializes in teaching programming and data science to scientists, engineers, data analysts, quants, and computer scientists in the Python language.

Python Charmers' trainers boast years of Python experience and deep roots in the open source community, as both speakers at events and contributors to well-known open source projects, including *NumPy*, *SciPy*, *Scikit-Learn*, *Pandas*, and *Python-Future*.

Testimonials: Testimonials from past participants of similar bootcamps and training courses are available at

<https://pythoncharmners.com/testimonials/>

Questions: We are happy to customise this program further on request. Please let us know if you would like to discuss this or have any other questions.

Contact:

Phone: +61 1300 963 160

Email: info@pythoncharmners.com

Web: pythoncharmners.com

The logo for Python Charmers features the word "PYTHON" in a large, bold, sans-serif font. The letter "O" is replaced by a stylized green Python snake head. Below "PYTHON", the word "CHARMERS" is written in a smaller, all-caps, sans-serif font.

PYTHON
CHARMERS

The logo features the word "PYTHON" in a bold, white, sans-serif font. The letter "O" is replaced by a stylized green Python logo. Below "PYTHON" is the word "CHARMERS" in a smaller, white, spaced-out sans-serif font, followed by a registered trademark symbol (®).

PYTHON
CHARMERS®