
ecan-python-2019 Documentation

Release 1.0.0

Mike Kittridge

Mar 18, 2019

Sections

1	Intended audience	3
2	Course summary	5
3	Prerequisites	7
4	Python Installation	9
5	Registration	11
6	Instructors	13
6.1	Prerequisites	13
6.2	Course 1 - Python Fundamentals	14
6.3	Course 2 - Pandas Fundamentals	14
6.4	Installing Python	15
6.5	License and terms of usage	17

The volume of data collected and available to scientists has grown in magnitudes over the past decades. It has grown to an extent that cannot be effectively handled by spreadsheets passed from one person to another. Fortunately, the toolsets to handle large amounts of data have grown as well, although it can be quite hard to keep up with the rapid developments of these toolsets.

The Python programming language is one of the most popular languages for both general purposes and scientific. This is due in part by the ease of use, open source code, and the large open source community that has developed a number of professional level toolsets for a wide range of applications.

The general goal of these courses is to teach Python tools that will benefit people who handle, process, and analyse lots of data. This will be accomplished through a combination of practical exercises and presentations.

All materials are accessible via the [course website](#).

CHAPTER 1

Intended audience

The intended audience for the courses are people with very little to some experience with programming (Python or otherwise). Those people with a lot of Python programming experience will not likely get much out of the courses unless they have not used the [Pandas package](#) in the past.

CHAPTER 2

Course summary

The courses will cover the python basics and the fundamental handling of tabular data and the associated processing and analysis tools. We will be primarily using the toolset contained within the [Pandas package](#). This will include reading/writing data, indexing, reshaping, computations, joining tables, time series handling, and visualization.

CHAPTER 3

Prerequisites

The only **prerequisite** for the courses is to familiarise yourself with Jupyter notebooks as this will be the primary way we will be interacting with Python. Please go through “Using the Jupyter Notebooks for the course modules” in the section *Prerequisites*.

CHAPTER 4

Python Installation

Installing Python is **not** a requirement of the courses. But you will need to install Python after the courses for your own work. The section *Installing Python* has a tutorial on how to install and use Python environments.

CHAPTER 5

Registration

Please sign up on the ME internal ECan site. There will be a maximum of 15 attendees for the workshop. Suggestions for advanced topics or examples are welcome.

- Mike Kittridge
 - Senior Scientist - Hydrologist
 - Environment Canterbury
 - mike.exner-kittridge@ecan.govt.nz
- Wilco Terink
 - Senior Scientist - Hydrologist
 - Environment Canterbury
 - wilco.terink@ecan.govt.nz

6.1 Prerequisites

6.1.1 Using the Jupyter Notebooks for the courses modules

This courses uses self contained code sets called [Jupyter Notebooks](#). The courses will not explicitly require you to install python on your PC, but you are welcome to try as described in the next paragraph. Consequently for the courses, the preferred method to run the notebooks will be through the [binder](#) links that build the correct python environment for the notebooks to be run under. This ensures that no one will have issues properly running the notebooks.

Please run through the short notebook [A quick tour of Jupyter/IPython Notebooks](#). It will familiarise you with Jupyter notebooks and some of its capabilities, but don't worry if you don't understand everything.

6.2 Course 1 - Python Fundamentals

6.2.1 Date

The course will take place at 13:00 on Wed March 13th or 9:00 on Thursday 14th, 2019.

6.2.2 Location

The location of the course will be in the Waimakariri Room.

6.2.3 Course Material

Through much internal debate, we've decided to utilise the [Introduction to Python](#). All are completely self-contained with associated exercises.

6.2.4 Schedule

Time	Module
9:00 - 10:30	Introduction to Python part 1
10:30 - 10:45	BREAK
10:45 - 12:15	Introduction to Python part 2

6.2.5 Reference material

- Basic Python
 - [Python Quick Reference \(Jupyter Notebook\)](#)
 - [Introduction to Python for Environmental Scientists](#)
 - [General Introduction to Python](#)

6.2.6 Glossary of terms

[Python 3 glossary](#)

6.3 Course 2 - Pandas Fundamentals

6.3.1 Date

The course will take place at 13:00 on Wed March 19th or 9:00 on Thursday 20th, 2019.

6.3.2 Location

The location of the course will be in the Waimakariri Room.

6.3.3 Course Material

We will be utilising the fantastic open-access course provided by the [Data School](#). All are completely self-contained and the Pandas fundamentals include an accompanying Youtube video series associated with the exercises.

6.3.4 Schedule

Time	Module
9:00 - 10:30	Pandas fundamentals part 1
10:30 - 10:45	BREAK
10:45 - 12:15	Pandas fundamentals part 2 and Time series

6.3.5 Post-workshop exercises and/or courses

- More Pandas functionality
 - [Best practices with Pandas](#)
 - [Official Pandas docs](#)
- Python and GIS spatial analysis
 - [Automating GIS Processes](#) (Fantastic course!)

6.3.6 Reference material

- Basic Python
 - [Python Quick Reference](#) (Jupyter Notebook)
 - [Introduction to Python for Environmental Scientists](#)
 - [General Introduction to Python](#)
- Pandas
 - [Pandas cheat sheet](#)
 - [Official Pandas docs](#)

6.3.7 Glossary of terms

[Python 3 glossary](#)

6.4 Installing Python

Installing Python is relatively simple and does not require admin rights to install.

6.4.1 Miniconda

The first step is to download and install the Python 3 64bit version of [Miniconda](#). Miniconda is a variant of the [Anaconda](#) distribution, but Miniconda only contains the base Python installation and the package manager [conda](#). See the earlier link for more details about Miniconda and how to install it.

6.4.2 Packages

Everything other than the base Python installation is considered an ancillary package. The base Python is meant to provide a foundation for other people and organizations to build upon... and many people have build many great packages. With tens of thousands of packages, something needs to manage the handling and cross-dependencies of all these packages. That's what conda does.

6.4.3 Channels

Conda also has the concept of Channels. Different people and organizations can have their own set of packages that they are working on and be kept separate from the greater ecosystem of packages. The default channel is managed by the Anaconda organization, while the channel called conda-forge is a community driven channel of packages. It is recommended to set conda-forge as the default channel to ensure all of the packages are installed consistently.

Go to the start menu and open up the recently added Anaconda prompt. Copy and paste the following line to the prompt to make conda-forge the default channel:

```
conda config --prepend channels conda-forge
```

Then update your packages to make everything nice and consistent:

```
conda update conda
```

6.4.4 Installing Packages

Now that you have Miniconda all set up, you can install any number of additional packages with one line on the Anaconda prompt opened from earlier:

```
conda install pandas spyder numpy
```

Where “conda install” tells conda to install packages which should be followed by the package names with spaces in between (in our case we want to install pandas, spyder, and numpy). Conda handles all of the downloading and installing of those packages.

At this point, you should be ready write Python code and run it in the [Spyder](#) development environment!

6.4.5 Environments

Just to add a bit more complexity to the existing Python, conda, packages, and channels from earlier, conda also has the concept of python environments. When you installed Python/Miniconda you created the base Python environment. When you opened the Anaconda prompt, you started in this base environment and any packages you install will be placed in this base environment. Additional environments allows you to create additional independent Python installations with their own independent set of packages. This becomes very important when you create multiple scripts with different sets of packages that you don't want to mix.

To create a new environment, use the following syntax:

```
conda create --name newenv python=3.6 pandas spyder
```

Where newenv is the name you are calling the new environment then followed by the packages you want to install.

Please download and look at the [conda cheat sheet](#) for many of the commands to use conda.

6.5 License and terms of usage

This package is licensed under the terms of the Apache License Version 2.0 and can be found on the [GitHub project page](#).