

# Python Pandas

This Python Pandas training course will teach you all about Pandas, from installing it to creating one- and two-dimensional indexed data structures, indexing and slicing-and-dicing that data to derive results, loading data from local and Internet-based resources, and finally creating effective visualizations to form quick insights.

## Prerequisites / Further Training

Recommended sequence	Summary
<a href="#">Introduction to Programming.</a>	Fundamentals of Computer Programming
<a href="#">Beginner Python</a>	Python Language and OO Fundamentals
<i>Python Pandas</i>	<i>Using Pandas Library</i>
<a href="#">Django</a> or <a href="#">Flask</a>	Python Web development in detail

Also have a look at our [Python Bootcamp](#)

## Intended Audience

- Analysts who wants to more on data analysis and less on programming
- Programmers who wants to performing data exploration and analysis on Python using pandas

## Overview

- pandas and why it is important
- pandas and IPython Notebooks
- Referencing pandas in the application
- Primary pandas objects
- The pandas Series object
- The pandas DataFrame object
- Loading data from files and the Web

- Loading CSV data from files
- Loading data from the Web
- Simplicity of visualization of pandas data

## **Installing pandas**

- Getting Anaconda
- Installing Anaconda
- Installing Anaconda on Linux
- Installing Anaconda on Mac OS X
- Installing Anaconda on Windows
- Ensuring pandas is up to date
- Running a small pandas sample in IPython
- Starting the IPython Notebook server
- Installing and running IPython Notebooks
- Using Wakari for pandas

## **NumPy for pandas**

- Installing and importing NumPy
- Benefits and characteristics of NumPy arrays
- Creating NumPy arrays and performing basic array operations
- Selecting array elements
- Logical operations on arrays
- Slicing arrays
- Reshaping arrays
- Combining arrays
- Splitting arrays
- Useful numerical methods of NumPy arrays

## **The pandas Series Object**

- The Series object
- Importing pandas
- Creating Series
- Size, shape, uniqueness, and counts of values
- Peeking at data with heads, tails, and take
- Looking up values in Serie

- Alignment via index labels
- Arithmetic operations
- The special case of Not-A-Number (NaN)
- Boolean selection
- Reindexing a Series
- Modifying a Series in-place
- Slicing a Series

## **The pandas DataFrame Object**

- Creating DataFrame from scratch
- Example data
- S&P 500
- Monthly stock historical prices
- Selecting columns of a DataFrame
- Selecting rows and values of a DataFrame using the index
- Slicing using the [] operator
- Selecting rows by index label and location: .loc[] and .iloc[]
- Selecting rows by index label and/or location: .ix[]
- Scalar lookup by label or location using .at[] and .iat[]
- Selecting rows of a DataFrame by Boolean selection
- Modifying the structure and content of DataFrame
- Renaming columns
- Adding and inserting columns
- Replacing the contents of a column
- Deleting columns in a DataFrame
- Adding rows to a DataFrame
- Appending rows with .append()
- Concatenating DataFrame objects with pd.concat()
- Adding rows (and columns) via setting with enlargement
- Removing rows from a DataFrame
- Removing rows using .drop()
- Removing rows using Boolean selection
- Removing rows using a slice
- Changing scalar values in a DataFrame

- Arithmetic on a DataFrame
- Resetting and reindexing
- Hierarchical indexing
- Summarized data and descriptive statistics

## **Accessing Data**

- Setting up the IPython notebook
- CSV and Text/Tabular format
- The sample CSV data set
- Reading a CSV file into a DataFrame
- Specifying the index column when reading a CSV file
- Data type inference and specification
- Specifying column names
- Specifying specific columns to load
- Saving DataFrame to a CSV file
- General field-delimited data
- Handling noise rows in field-delimited data
- Reading and writing data in an Excel format
- Reading and writing JSON files
- Reading HTML data from the Web
- Reading and writing HDF5 format files
- Accessing data on the web and in the cloud
- Reading and writing from/to SQL databases
- Reading data from remote data services
- Reading stock data from Yahoo! and Google Finance
- Retrieving data from Yahoo! Finance Options
- Reading economic data from the Federal Reserve Bank of St. Louis
- Accessing Kenneth French's data
- Reading from the World Bank

## **Tidying Up Your Data**

- What is tidying your data?
- Setting up the IPython notebook
- Working with missing data
- Determining NaN values in Series and DataFrame objects

- Selecting out or dropping missing data
- How pandas handles NaN values in mathematical operations
- Filling in missing data
- Forward and backward filling of missing values
- Filling using index labels
- Interpolation of missing values
- Handling duplicate data
- Transforming Data
- Mapping
- Replacing values
- Applying functions to transform data

## **Combining and Reshaping Data**

- Setting up the IPython notebook
- Concatenating data
- Merging and joining data
- An overview of merges
- Specifying the join semantics of a merge operation
- Pivoting
- Stacking and unstacking
- Stacking using nonhierarchical indexes
- Unstacking using hierarchical indexes
- Melting
- Performance benefits of stacked data

## **Grouping and Aggregating Data**

- Setting up the IPython notebook
- The split, apply, and combine (SAC) pattern
- Split
- Grouping by a single column's values
- Accessing the results of grouping
- Grouping using index levels
- Apply
- Applying aggregation functions to groups
- The transformation of group data
- An overview of transformation

- Practical examples of transformation
- Filtering groups
- Discretization and Binning

## **Time-series Data**

- Setting up the IPython notebook
- Representation of dates, time, and intervals
- The datetime, day, and time objects
- Timestamp objects
- Timedelta
- Introducing time-series data
- DatetimeIndex
- Creating time-series data with specific frequencies
- Calculating new dates using offsets
- Date offsets
- Anchored offsets
- Representing durations of time using Period objects
- The Period object
- PeriodIndex
- Handling holidays using calendars
- Normalizing timestamps using time zones
- Shifting and lagging
- Frequency conversion
- Up and down resampling
- Time-series moving-window operations

## **Visualization**

- Setting up the IPython notebook
- Plotting basics with pandas
- Creating time-series charts with .plot()
- Adorning and styling your time-series plot
- Adding a title and changing axes labels
- Specifying the legend content and position
- Specifying line colors, styles, thickness, and markers
- Specifying tick mark locations and tick labels
- Formatting axes tick date labels using formatters

- Common plots used in statistical analyses
- Bar plots
- Histograms
- Box and whisker charts
- Area plots
- Scatter plots
- Density plot
- The scatter plot matrix
- Heatmaps
- Multiple plots in a single chart

## **Applications to Finance**

- Setting up the IPython notebook
- Obtaining and organizing stock data from Yahoo!
- Plotting time-series prices
- Plotting volume-series data
- Calculating the simple daily percentage change
- Calculating simple daily cumulative returns
- Resampling data from daily to monthly returns
- Analyzing distribution of returns
- Performing a moving-average calculation
- The comparison of average daily returns across stocks
- The correlation of stocks based on the daily percentage change of the closing price
- Volatility calculation
- Determining risk relative to expected returns

---

## **Duration and pricing**

In [Price Group A](#)

## **Certificate**

1. Upon completion of this course we will issue you with attendance certificate to certify your attendance and / or completion of the prescribed minimum examples.

2. You have the option to get the competency / academic certificate if you :  
hand in a project (pre-approved) covering most of the topics in the book.
3. If you have not enrolled for the course you may opt to sit only for the competency certificate, at a cost of R3500.

## **Schedule**

On the calendar on this page below.

If your browser doesn't display the calendar below, please click on [this link](#) or try using [Google Chrome](#), alternatively please enquire via our '[Contact Us](#)' page.

## **Bookings**

Please click [click here](#) or send us an [email](#).

## **Questions**

Please [email us](#)