

# Pandas

Module pour l'analyse de données, pouvant se substituer à l'utilisation d'un tableur. Une différence fondamentale de la librairie pandas avec NumPy, c'est que les tableaux NumPy (NumPy arrays) ont le même type (dtype) pour le tableau entier, tandis que les tableaux pandas (pandas DataFrames) sont caractérisés par un type unique (dtype) par colonne.

## Installation

- [Instructions sur le site officiel](#)
- Installé avec Anaconda
- Ubuntu : `pip3 install pandas`

## Documentation

- [Documentation officielle](#)
- [10 minutes to pandas](#)
- [cookbook](#)
- [Visualisation](#)

## Applications, exemples

### Statistiques sur les dimensions des humains (body dimensions)

Programme basé sur [Exploring Relationships in Body Dimensions](#).

Extensions :

- Tester et utiliser en mode "Jupyter"
- créer des régressions
- autres représentations
- différenciation suivant le genre, l'âge
- ...

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
```

```
Created on Tue Mar  5 04:13:51 2019
Statistics on Body dimensions :
http://jse.amstat.org/v11n2/datasets.heinz.html
```

```
without requests lib, using pandas.read_csv
```

```
@author: Didier Villers
```

```
"""
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

# using this VARIABLE DESCRIPTIONS with PEP 8 Python style :
names = [
    'Biacromial diameter',
    'Biiliac diameter',
    'Bitrochanteric diameter',
    'Chest depth',
    'Chest diameter',
    'Elbow diameter',
    'Wrist diameter',
    'Knee diameter',
    'Ankle diameter',
    'Shoulder girth',
    'Chest girth',
    'Waist girth',
    'Navel girth',
    'Hip girth',
    'Thigh girth',
    'Bicep girth',
    'Forearm girth',
    'Knee girth',
    'Calf maximum girth',
    'Ankle minimum girth',
    'Wrist minimum girth',
    'Age',
    'Weight',
    'Height',
    'Gender',
]

# using Pandas column names without white spaces
names = [name.replace(' ', '_') for name in names]
print(names)

namesfr = [
    'Largeur des épaules',
    'Largeur des hanches',
    'Largeur entre têtes de fémur',
    'Epaisseur du thorax',
    'Largeur du thorax',
    'Largeur du coude',
    'Largeur du poignet',
    'Largeur du genou',
    'Largeur de la cheville',
    'Tour d'épaules',
    'Tour de poitrine',
    'Tour de taille',
]
```

```
'Tour au niveau du nombril',
'Tour de hanches',
'Tour de cuisse',
'Tour du biceps',
'Tour de l'avant-bras',
'Tour de genou',
'Plus grande circonférence du mollet',
'Plus petite circonférence de la cheville',
'Plus petite circonférence du poignet',
'Âge',
'Poids',
'Taille',
'Genre',
]

dict_names_fr = dict(zip(names, namesfr))
print(dict_names_fr)

file_url = "http://linus.umons.ac.be/body.dat.txt" # file copy
#file_url = "http://jse.amstat.org/datasets/body.dat.txt"
# using read_csv
#
#https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_csv.html
# https://www.datacamp.com/community/tutorials/pandas-read-csv

df = pd.read_csv(file_url, header=None, names=names, delimiter=' | ',
engine='python', index_col=False)
#
print(df)

# pandas misc
#
#https://stackoverflow.com/questions/15315452/selecting-with-complex-criteria-from-pandas-dataframe
#
print(df.columns)
print(df.Age)
print(df.dtypes)
print(df.describe())
print(df[df.Gender == 1].describe())
print(df[df.Age == 20].describe())
print(df.sort_values(by = 'Height'))

print(df.query('Age > 25 and Age < 30'))
print(df.query('25 < Age < 30'))

plt.figure()
ax = df[df.Gender == 1].plot.scatter(x='Height', y='Weight', color='Red',
label='Male');
df[df.Gender == 0].plot.scatter(x='Height', y='Weight', color='Green',
```

```
label='Female', ax=ax);

plt.figure()
df.Height.plot.hist()

plt.figure()
df.Weight.plot.hist()
```

## Références

- [https://www.tutorialspoint.com/python\\_pandas/index.htm](https://www.tutorialspoint.com/python_pandas/index.htm)
- <https://medium.com/@devopslearning/introduction-to-pandas-for-data-analysis-c14bb9b1c21b> (limité)
- [First Python Notebook. A step-by-step guide to analyzing data with Python and the Jupyter Notebook](#) (The course will teach you how to use pandas to read, filter, join, group, aggregate and rank structured data. You will also learn how to record, remix and republish your analysis using the Jupyter Notebook) → commencer au chapitre 3 “Import pandas into a Jupyter Notebook”
- [Python Data Analysis with pandas](#)
- [Python Pandas Tutorial : Learn Pandas for Data Analysis](#)
- [Python for Data Science: 8 Concepts You May Have Forgotten](#)
- [23 great Pandas codes for Data Scientists](#)
- fonction merge :  
<https://towardsdatascience.com/why-and-how-to-use-merge-with-pandas-in-python-548600f7e738>
- [Helpful Python Code Snippets for Data Exploration in Pandas](#)
- [Selecting Subsets of Data in Pandas: Part 1](#)
- [Statistical Data Analysis in Python](#)
- [How to Make Boxplots in Python with Pandas and Seaborn?](#) (and Gapminder dataset)
- [Box plot visualization with Pandas and Seaborn](#)

## Exemples divers

- [http://pbpython.com/pandas\\_transform.html](http://pbpython.com/pandas_transform.html)
- <http://blog.yhat.com/posts/visualize-nba-pipelines.html>
- <https://tomaugspurger.github.io/>
- régressions linéaires :
  - [http://www.xavierdupre.fr/app/ensae\\_teaching\\_cs/helpsphinx/notebooks/td2a\\_eco\\_regressions\\_lineaires.html](http://www.xavierdupre.fr/app/ensae_teaching_cs/helpsphinx/notebooks/td2a_eco_regressions_lineaires.html)

From:

<https://dvillers.umons.ac.be/wiki/> - **Didier Villers, UMONS - wiki**

Permanent link:

<https://dvillers.umons.ac.be/wiki/teaching:progappchim:pandas?rev=1552283659>

Last update: **2019/03/11 06:54**

