

Jupyter lab and markdown

• V1: 2024/03/11(Muriel Pressigout)

The pdf file has been generated from a Markdown that can be studied as a Markdown syntax example.

1. Jupyter lab

1.1 Context

It is a new web-based interactive development environment : Jupyter notebooks (files .ipynb) where a kernel can execute cells as code, texts or plot diagrams, figures as in matlab. Jupyter lab is the interface that gather Jupyter notebooks and others components useful to interact with data and code: terminals for example.

We will use them only to have an interactive tutorial: this notebook is an interactive document with cells that are either markdown or raw text.

It is quite convenient when you want to have explanations, see the test commands and write the outputs as a document (title, comments,...) all together.

You can export those documents as pdf (subject and answers) if you have properly set up the pdf generator.

2. How to use it? (and test setup)

It is recommended to use the VM to have the Jupyter notebooks on your PC : no network or server issues, and you can set up the pdf tools.

2.1 Using the provided VM (recommended)

In a terminal, type :

- conda activate cling
- jupyter-lab tuto-make-main-getopt.ipynb &

It will launch in your web browser the jupyter lab where you can add a notebook (.ipynb) choosing a python kernel.

When you have finished, you can type:

- conda deactivate

2.2 Edit the cells

Alls cells are using Markdown syntax.

Markdown is a simple and easy-to-use markup language : <https://en.wikipedia.org/wiki/Markdown>

Basic syntax :

Heading

Sub-heading

Alternative heading

Alternative sub-heading

Paragraphs are separated by a blank line.

Two spaces at the end of a line
produce a line break.

Text attributes *italic*, **bold**, monospace.

Horizontal rule:

Bullet lists nested within numbered list:

1. fruits
 - apple
 - banana
2. vegetables
 - carrot
 - broccoli

2.3 Tips

- 'Shift'+ 'enter' to run a cell
- select a cell and type 'c' to copy it
- select a cell and type 'v' to paste the copied cell just after the selected one
- select a cell and type twice 'd' to delete it
- you can clear all output in the Edit Menu
- restart the kernel if you have modified a class before running the related cell (as the code is managed a a script, it will be considered a new declaration with same name leading to a compiler error)