UNIVERSITÀ DI TORINO

Jupyter Notebooks

Prof. Fabio Ciravegna
Dipartimento di Informatica
Università di Torino
fabio.ciravegna@unito.it

© Prof. Fabio Ciravegna, Università di Torino

UNIVERSITÀ DI TORINO

Jupyter notebooks

- Jupyter notebooks are interactive documents that contain live code, equations, visualisations, and narrative text
  - They are a popular tool for data science, machine learning, and scientific computing
- Jupyter notebooks can be used to create and share documents, collaborate on projects, and teach and learn

© Prof. Fabio Ciravegna, Università di Torino

UNIVERSITÀ DI TORINO

Why use Jupyter notebooks?

- Jupyter notebooks have a number of advantages over other programming environments, including:
  - Interactivity: Jupyter notebooks allow you to run code and see the results immediately. This makes them ideal for exploratory programming and debugging
  - Rich text support: Jupyter notebooks support markdown formatting, which allows you to add text, images, and equations to your code cells. This makes it easy to create and share computational narratives
  - Collaboration: Jupyter notebooks can be easily shared with others, making them ideal for collaboration on projects
  - Language agnostic: Jupyter notebooks support a wide range of programming languages, including Python, R, Julia, and JavaScript

© Prof. Fabio Ciravegna, Università di Torino

How to use Jupyter notebooks

- To use Jupyter notebooks, you will need to install the Jupyter Notebook software.
  - This is available for free for all major operating systems
  - Once you have installed Jupyter Notebook, you can create a new notebook by typing jupyter notebook in a terminal window.
- Jupyter notebooks are divided into cells.
  - Each cell can contain either code or text. To create a new code cell, click the "+" button in the toolbar and select "Code".
  - To create a new text cell, click the "+" button and select "Markdown".
  - To run a code cell, click the "Run" button in the toolbar.
    - The output of the code will be displayed below the cell.
- You can add text, images, and equations to your code cells using markdown formatting.
  - To learn more about markdown, see the markdown documentation: https://www.markdownguide.org/

© Prof. Fabio Ciravegna, Università di Torino

UNIVERSITÀ DI TORINO

Markdown

Basic Syntax

These are the elements outlined in John Gruber's original design document. All Markdown applications support these elements.

Element	Markdown Syntax

Heading	`# H1`
	`## H2`
	`### H3`

Bold	`**bold text**`
Italic	`*italicized text*`
Blockquote	`> blockquote`

Ordered List	`1. First item`
	`2. Second item`
	`3. Third item`

Unordered List	`- First item`
	`- Second item`
	`- Third item`

Code	`` `code` ``
Horizontal Rule	`---`
Link	`[title](https://www.example.com)`
Image	`![alt text](image.jpg)`

https://www.markdownguide.org/cheat-sheet/

© Prof. Fabio Calvetti, Università di Torino

UNIVERSITÀ DI TORINO

Extended Syntax

These elements extend the basic syntax by adding additional features. Not all Markdown applications support these elements.

Element	Markdown Syntax

Table	`| Syntax | Description |`
	`| ---------- | ---------- |`
	`| Header | Title |`
	`| Paragraph | Text |`

Fenced Code Block	```` ```
	{
	  "firstName": "John",
	  "lastName": "Smith",
	  "age": 25
	}
	``` ````

Footnote	`Here's a sentence with a footnote. [^1]`
	
	`[^1]: This is the footnote.`

Heading ID	`### My Great Heading {#custom-id}`
Definition List	`term` `: definition`
Strikethrough	`~~The world is flat.~~`
Task List	`- [x] Write the press release` `- [ ] Update the website`

© Prof. Fabio Ciravegna, Università di Torino

UNIVERSITÀ DI TORINO

Editing

jupyter my _first_botebook Last Checkpoint: 1 minute ago

File Edit View Run Kernel Settings Help

Code

```
# This is my new notebook
## let's see how it works
Honestly, you should really pay attention now!
```

[1]:
```
x= 3
print(x)
```

```
3
```

[2]:
```
def my_function(parameter=4):
    return parameter*44

my_function(1234)
```

[2]:
```
54296
```

[ ]:

[ ]:

[ ]:

UNIVERSITÀ DI TORINO

Running

jupyter my _first_botebook Last Checkpoint: 1 minute ago

File Edit View Run Kernel Settings Help

Code

This is my new notebook

let's see how it works

Honestly, you should really pay attention now!

[1]:
```
x= 3
print(x)
```

```
3
```

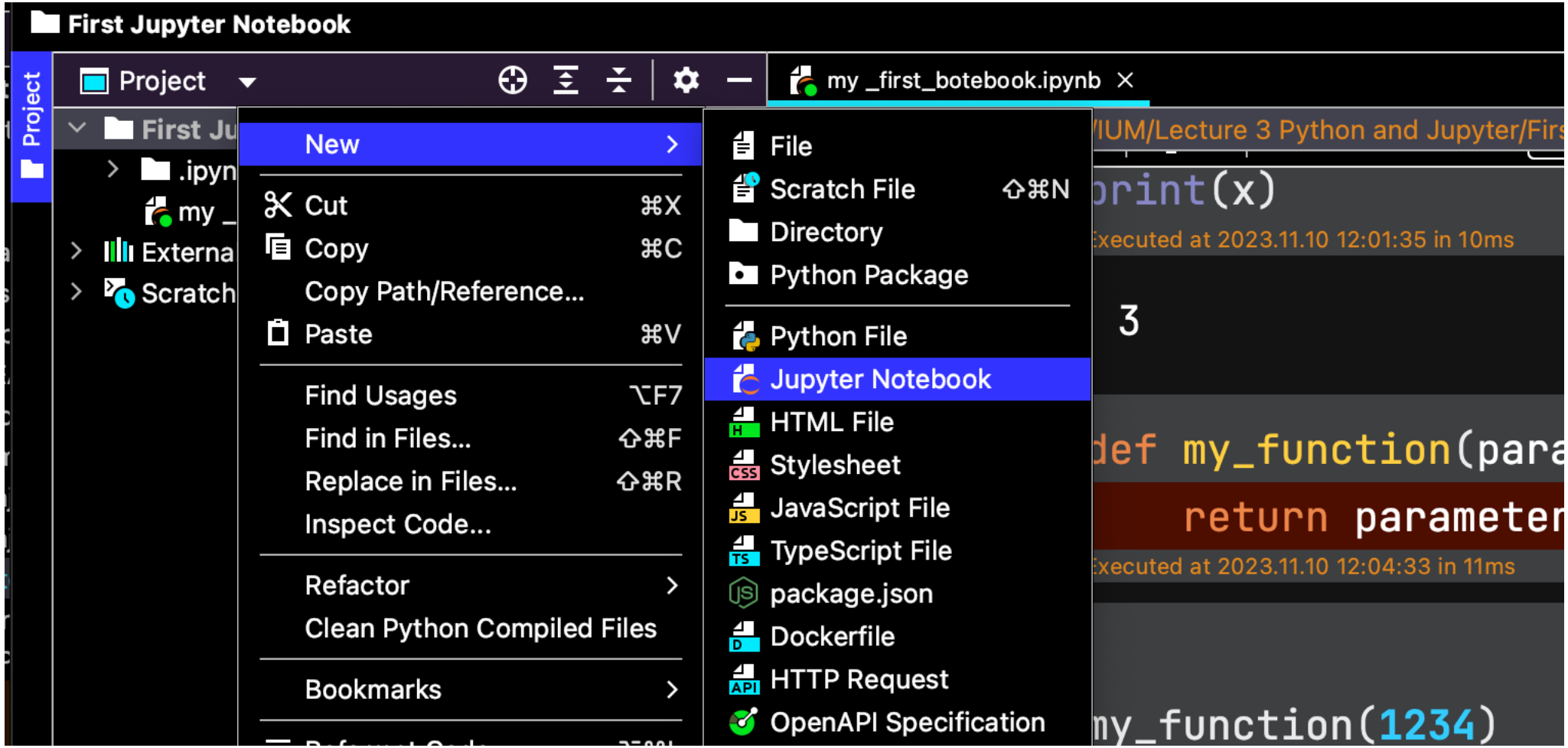
[2]:
```
def my_function(parameter=4):
    return parameter*44

my_function(1234)
```

[2]:
```
54296
```

[ ]:

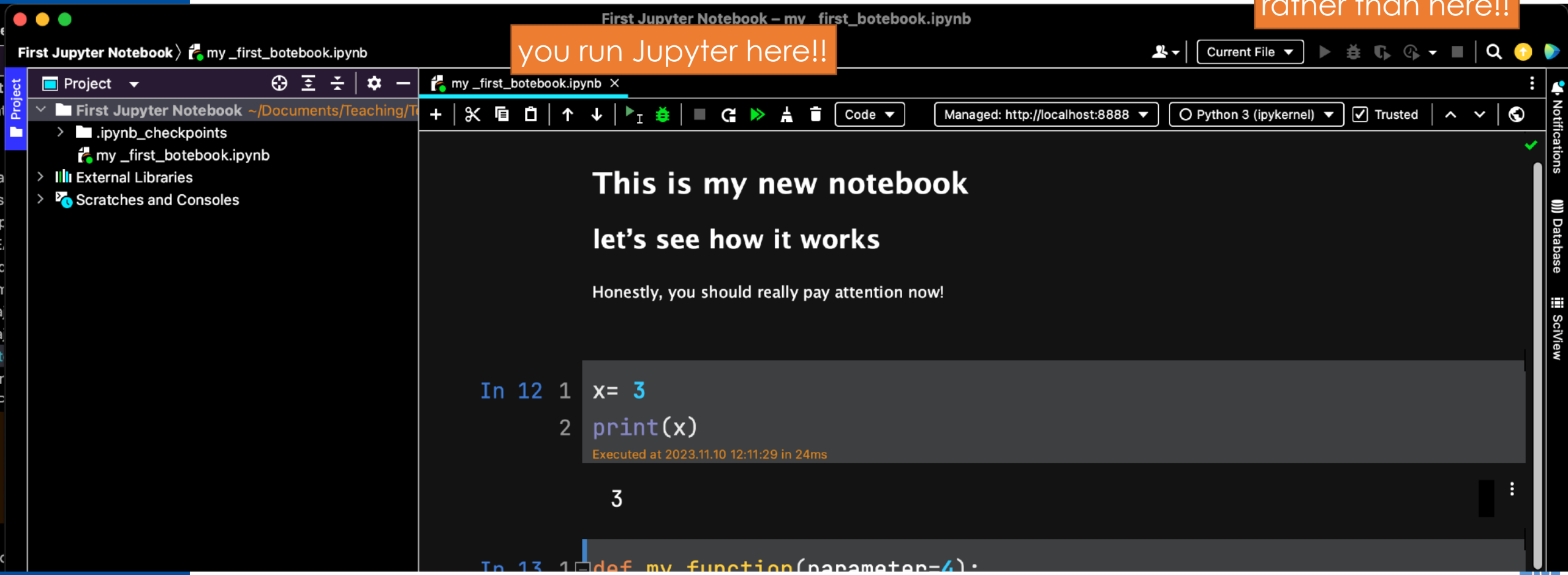
We will use PyCharm

- So to Allow debugging



© Prof. Fabio Ciravegna, Università di Torino

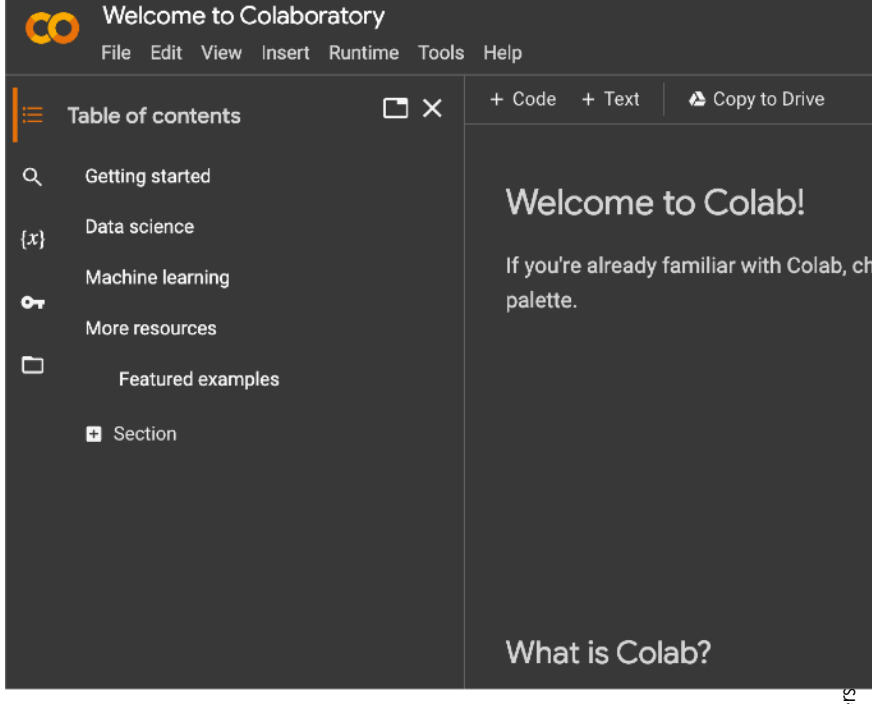
UNIVERSITÀ DI TORINO

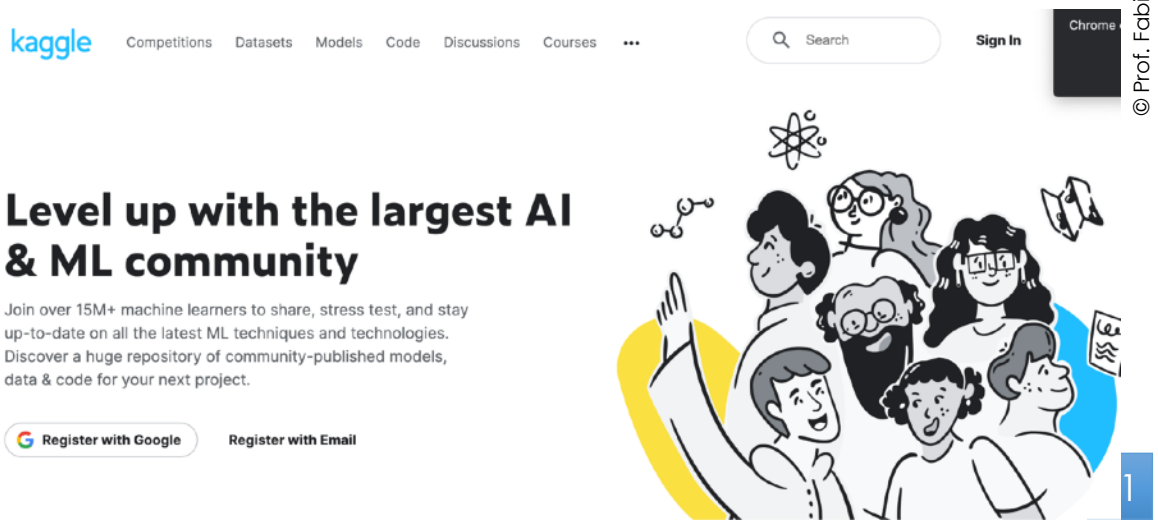


UNIVERSITÀ DI TORINO

Other tools

- Google Colab
  - pretty standard in development of AI
  - quite powerful machines available
  - https://colab.research.google.com/
- Kaggle
  - advantage: lots of available data repositories that you can use directly
  - https://www.kaggle.com/





© Prof. Fabio Ciravegna, Univers