Introduction to Python

Hao Zhu

Department of Computer Science
University of Freiburg

Setting up Python

- many possible options (conda, mamba, venv, *etc.*)
- our suggestion: we will use uv for dependency management
  - install uv at
    `https://docs.astral.sh/uv`
  - clone our GitHub repository at
    `https://github.com/nrgrp/cvxopt_gfm_workshop`
  - open the repository and install dependencies:

```
cd cvxopt_gfm_workshop && make install
```

Python notebooks

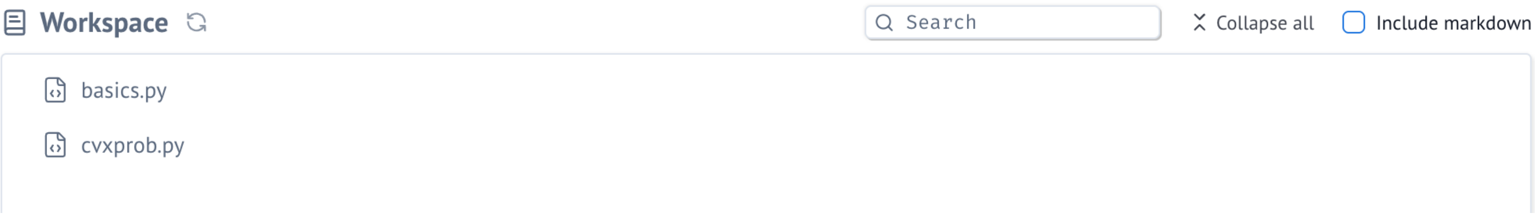
- many possible options (Jupyter, JupyterLab, VSCode, *etc.*)

- we will use marimo in this workshop
  - marimo vs Jupyter: guarantee reproducibility, better version control, native interactive elements, . . .
  - to start a marimo server:

    ```
    make marimo
    ```

    then open the given URL in your browser

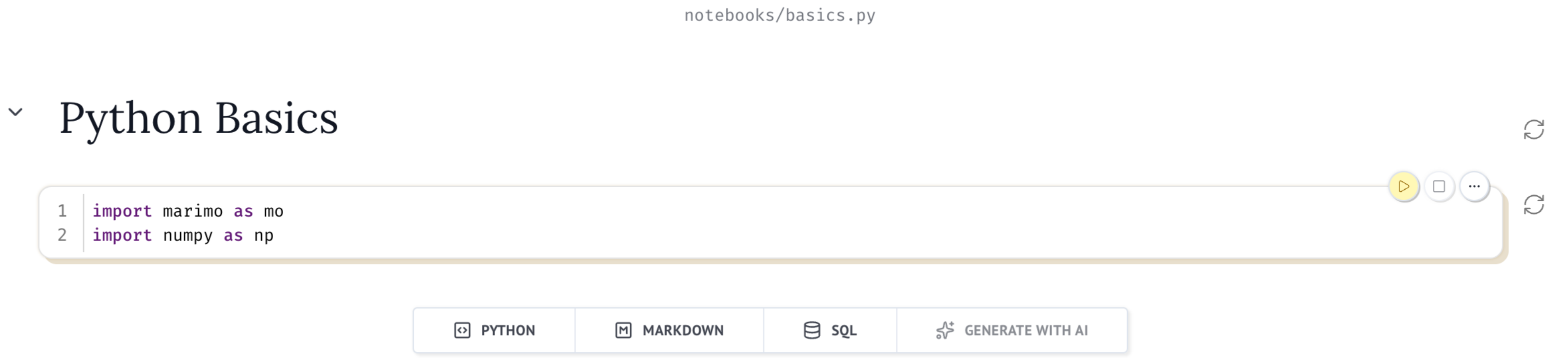
- notebooks in the launch folder are listed



basics.py
cvxprob.py

- click on a notebook to open it

- a new page should open in your browser that looks like



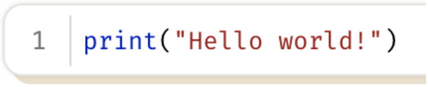
notebooks/basics.py

Python Basics

```python
import marimo as mo
import numpy as np
```

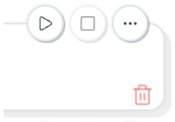
Coding in a marimo notebook

- the Jupyter notebook is organized into cells
- you can type code directly into a cell

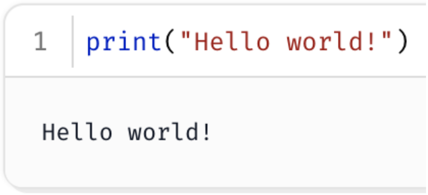


```
print("Hello world!")
```

- you can run a cell by clicking it to select it, and then either clicking on the play button in the toolbar or pressing Shift + Enter



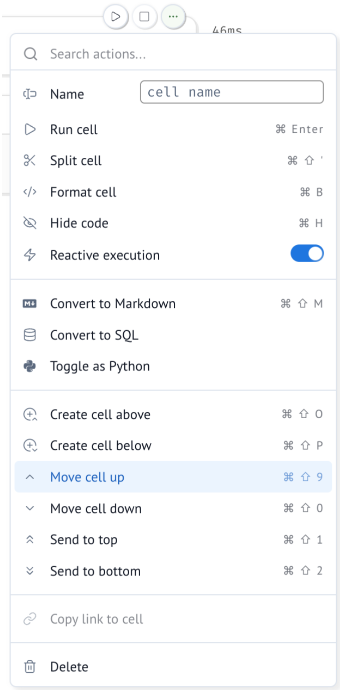
- the output of the cell will be displayed after it is run



```
print("Hello world!")
```

```
Hello world!
```

- the menu bar will contain options for creating/deleting cells, converting cell types, *etc.*



- more useful materials about coding with marimo at

  `https://docs.marimo.io`

Migrate from Jupyter

- notebooks are stored as standard python script (`.py`) files
- the same variable cannot be defined in multiple cells
- some suggestions (or really, requirements):
  - encapsulating code into functions when possible to minimize the number of global variables
  - prefixing temporary variables with an underscore (*e.g.*, `_my_var`), which makes the variable **local** to a cell
  - mutating variables in the cell that defines them
- more information at

  `https://docs.marimo.io/guides/coming_from/jupyter`