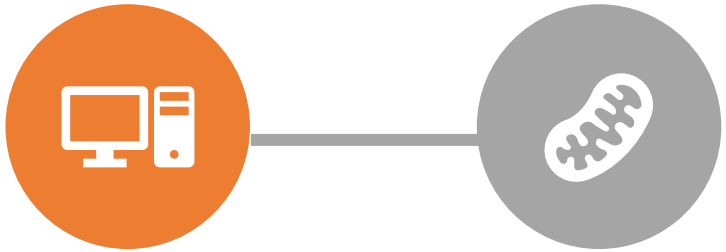
Python Tsunami

Installation and Tools

– April 21st-23rd –

UCPH HeaDS

How do I work with Python?



SETTING UP PYTHON

VIRTUAL ENVIRONMENTS

JUPYTER NOTEBOOKS

GOOGLE COLAB

Setting up Python

Pros:

- Your **own local** installation
- Easy **access to data** stored locally
- More **control** over installed versions
- **Not difficult** if done **right**

Cons:

- May require solving **issues** (sometimes obscure) if **incorrectly installed**
- Each **Operating System** (Windows, MacOS, Linux, …) may require **specific steps**

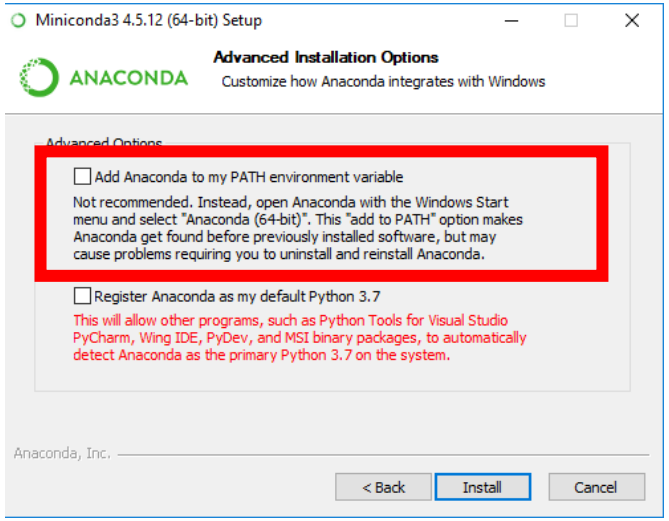
Setting up Python

- **Installation:**
  - **Miniconda:** a free minimal installer for **conda**

an open source **package management system** that quickly **installs**, **runs** and **updates** packages and their dependencies. Runs on **Windows**, **macOS** and **linux**.

https://docs.conda.io/en/latest/miniconda.html

- **Find your Operating System**
- **Download Python 3.8**
- **Install it following the steps and make sure to check:**



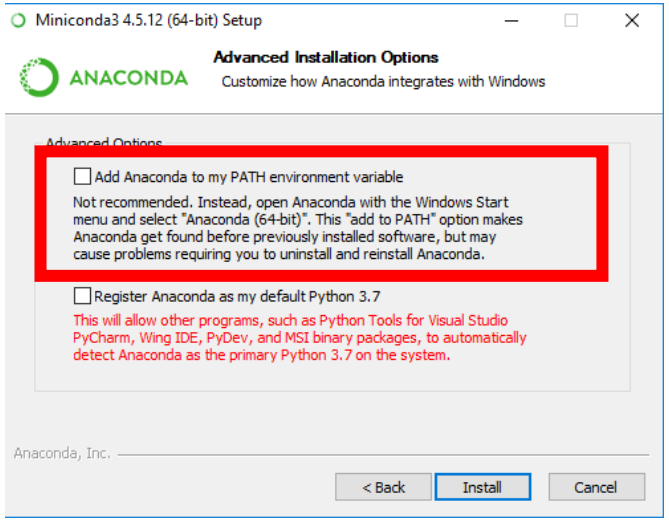
Setting up Python

- **Installation:**
  - **Miniconda:** a free minimal installer for **conda**

an open source **package management system** that quickly **installs**, **runs** and **updates** packages and their dependencies. Runs on **Windows**, **macOS** and **linux**.

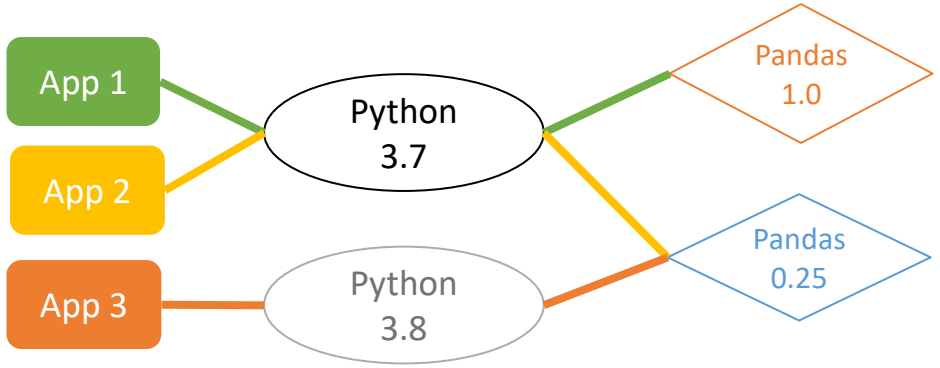
https://docs.conda.io/en/latest/miniconda.html

- **Find your Operating System**
- **Download Python 3.8**
- **Install it following the steps and make sure to check:**

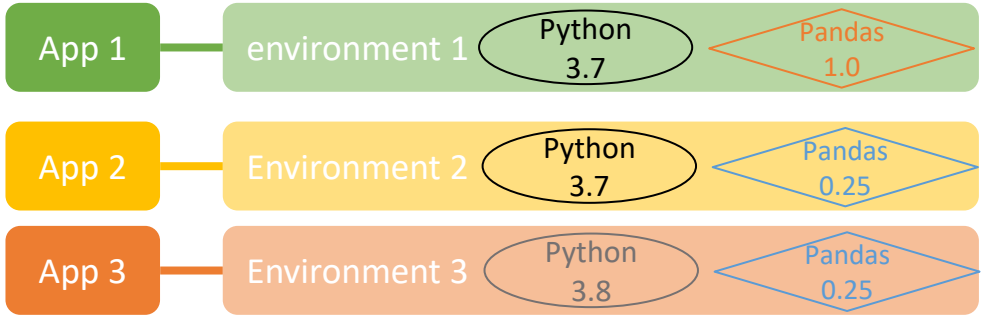


Virtual Environments

- **Python apps** will often use **packages and modules** that are **not part of the standard library**
- Apps sometimes need a **specific version of a library**
- This means, **one single Python installation may not meet the requirements of every application**



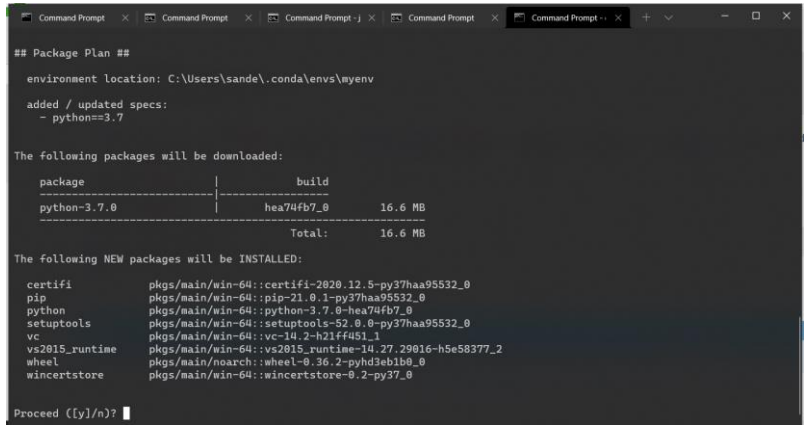
- The **solution** is to **create a virtual environment**, a self-contained directory that contains **a Python installation** (version) and the **necessary packages**



Virtual Environments

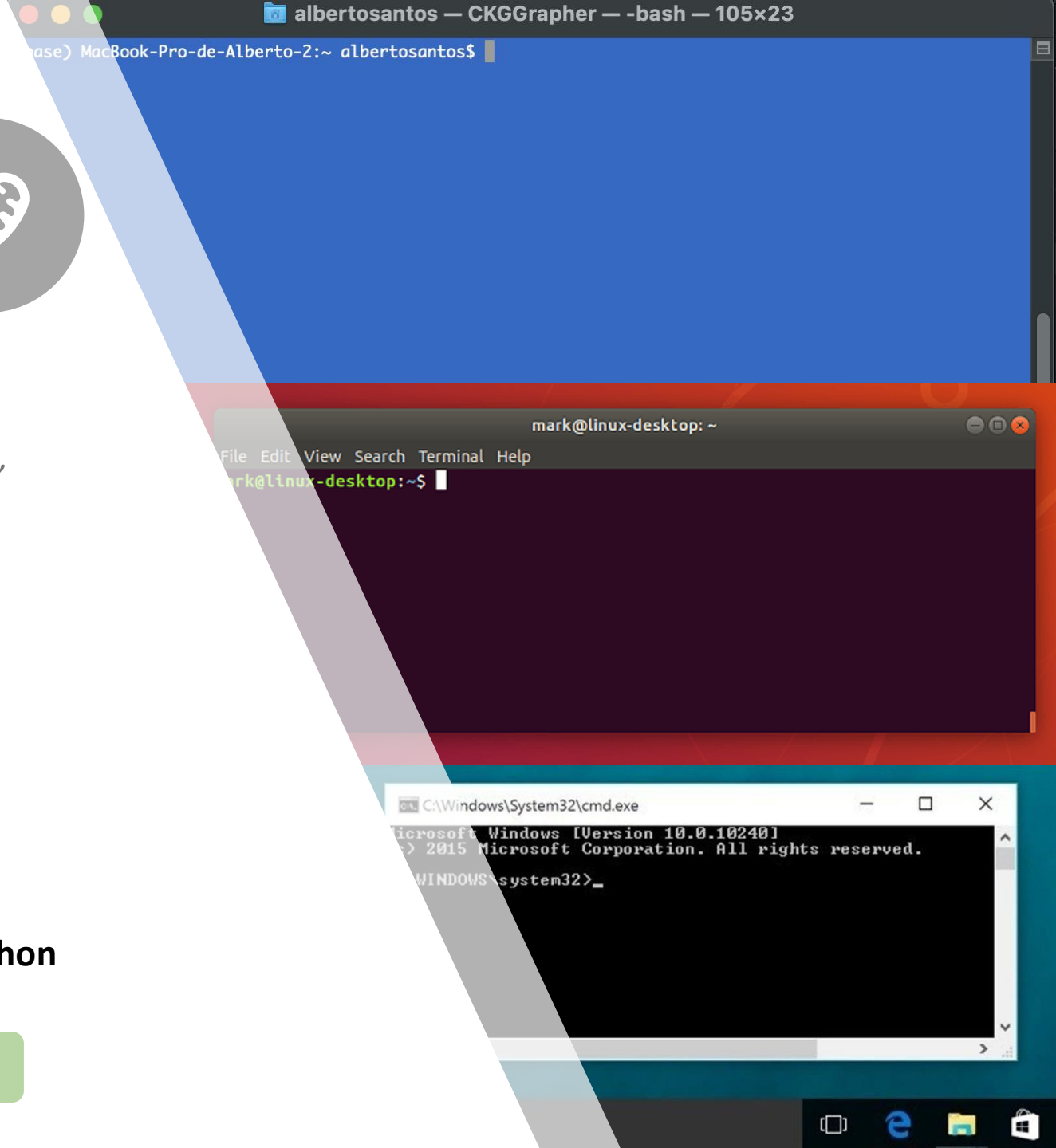
- Let's create a virtual environment:
  - To **create an environment** we use:

  conda create -n name_env python==version
  - **Open a Terminal** window
  - Type *conda create -n myenv python==3.7* and *'y'*

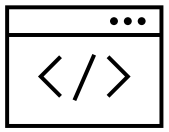


  - **To** activate the **environment type:**

  conda activate myenv
  - Now, if we type *python* we have the **right python version**

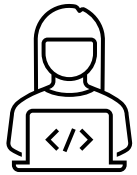


What tool do I use for coding?

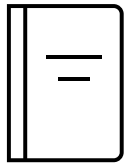


- **Terminal** → 2 options:
  - **Shell**: *python*
  - **Interactive** shell: *ipython*

- Simple **text editors:**
  - **Notepad++**
  - **Vim**



- **Integrated development environments** (IDEs) → Several options:
  - IDLE
  - PyCharm
  - Spyder
  - Visual Studio Code



- **Notebooks**

Jupyter Notebooks

The **Jupyter Notebook** is **an open-source application** to **create and share** documents that contain **code, equations, visualizations and text** (markdown)

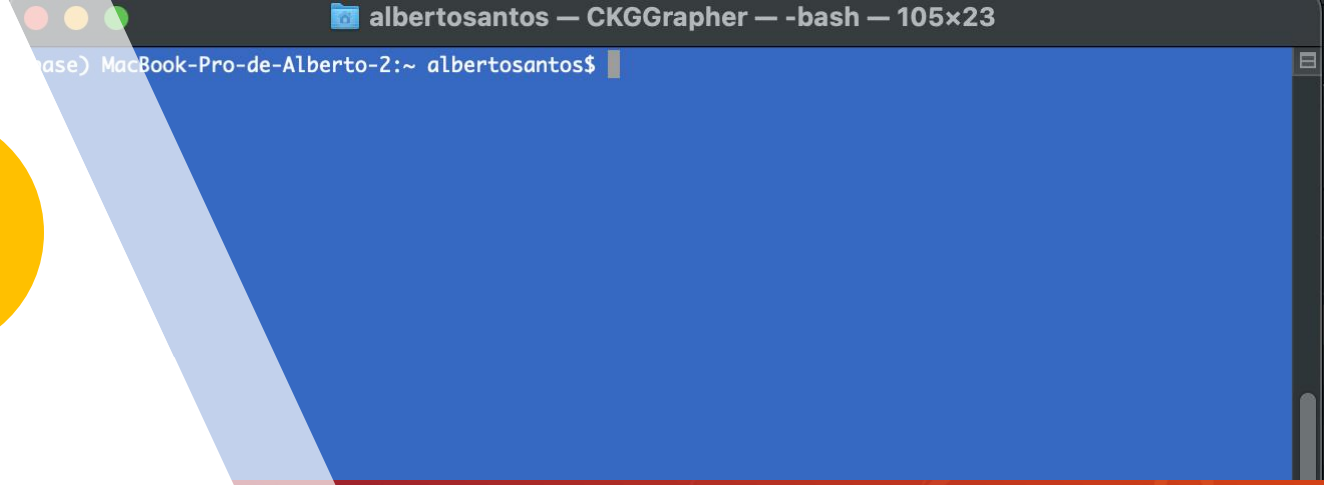
Jupyter Notebooks

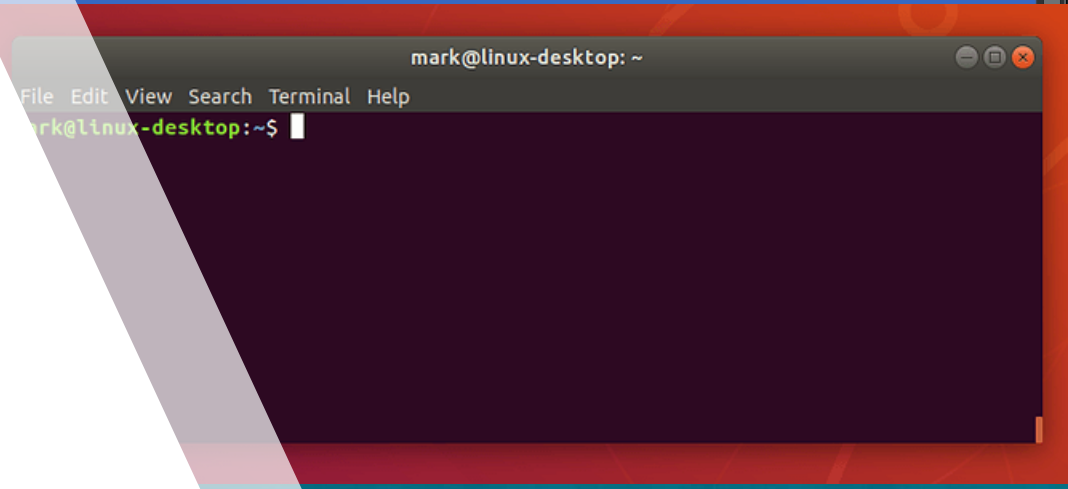
- Let's install Jupyter Notebooks:
  - Open a Terminal window
  - Activate your environment:

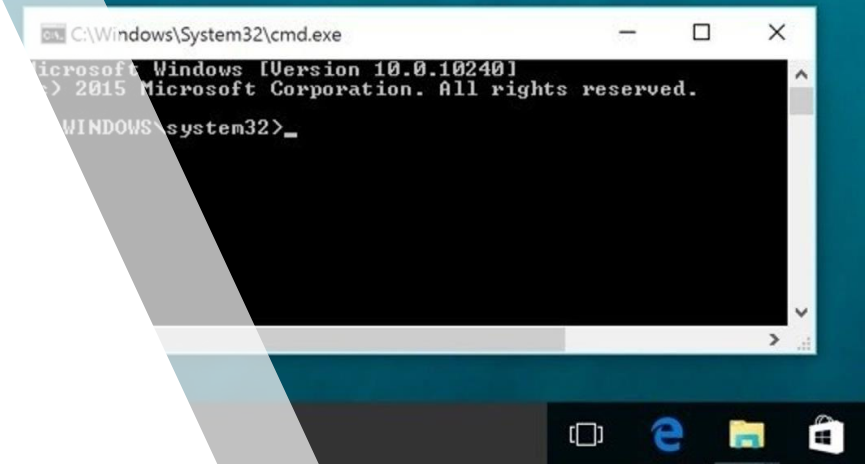
  conda activate myenv
  - Use *pip* to install Jupyter*:

  pip install notebook
  - To start notebooks:

  jupyter notebook







*This is where some OS may start complaining

Google Colab

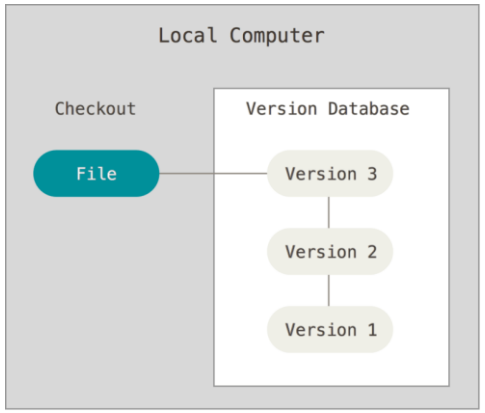
- **Colab** (Colaboratory) is a tool to **write**, **execute** and **share python code** through the **browser**
- Colab is especially well suited to **machine learning**, **data analysis** and **education**.
- Colab is a **hosted Jupyter notebook service** that requires **no setup** to use, while providing **free access** to computing **resources** including GPUs
- It is **connected** to a **Google account** and **data** can be **accessed** through **Google Drive**.

https://colab.research.google.com/

Taking Control of Your Code

Version control

- It is a system that **records changes** to a file or set of files over time so that you can **recall specific versions** (history)



- **Important** in coding, especially when the code is accessed by “many” people (>1)
- A good way to **share your code** with others

- **Git** is a software for **tracking changes** and **coordinating** work among collaborators developing code
- **GitHub** facilitates access to Git’s functionality

https://github.com/Center-for-Health-Data-Science/PythonTsunami

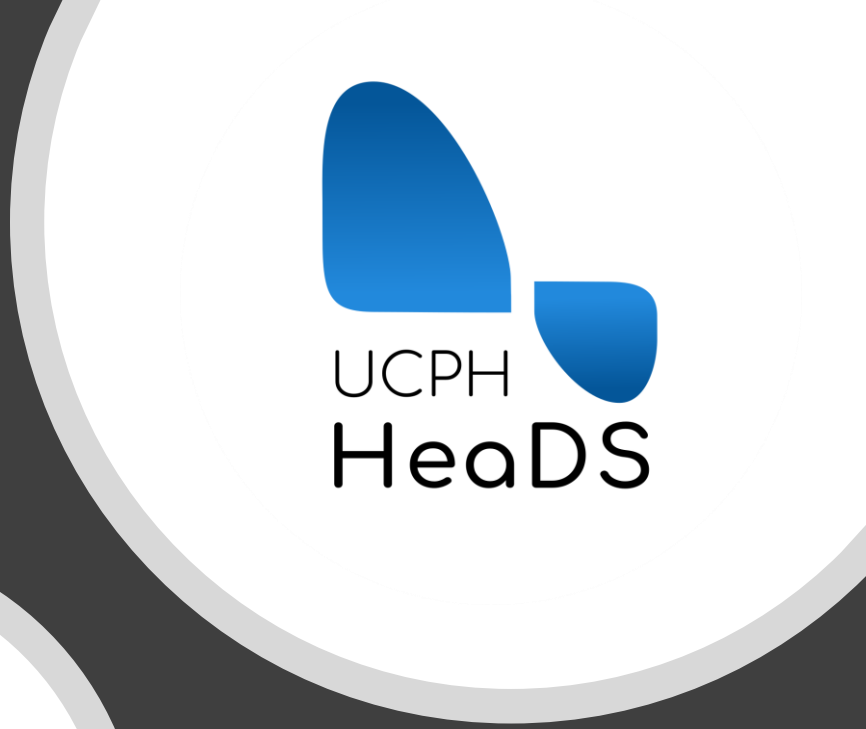
Recommended

git & GitHub Workshop

How we learned to stop worrying and love the git

UCPH HeaDS

BIG DATA
INSTITUTE


UCPH
HeaDS

The Novo Nordisk Foundation
Center for Protein Research

HAFNIENSIS
REFORMATÆ 1537
SIGILLVM VNIVERSITATIS
FVNDATÆ 1479