NOVA IMS Information Management School

Data Mining

MDSAA

Practical Class #1

Fall Semester 2023-2024

João Fonseca - jpfonseca@novaims.unl.pt

Farina Pontejos – fpontejos@novaims.unl.pt

Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

Please download the lab materials in the meantime

Check Moodle for instructions

About Us

João Fonseca

Academic background:

- Graduation in Economics (2016 – NOVA SBE)
- MSc in Management (2019 – NOVA SBE)
- MSc in Information Management (2019 – NOVA IMS)
- PhD in Information Management (2023 – NOVA IMS)

Research work:

- Research in Remote Sensing, Natural Language Processing and Machine Learning methods (synthetic data generation and active learning)
- Research in Responsible AI and Explainable AI

About Us

Farina Pontejos

Academic background:

- Graduation in Mining Engineering (2011)
- MSc in Data Science & Advanced Analytics (2023 – NOVA IMS)
- PhD in Information Management (since 2023 [Ongoing] – NOVA IMS)

Professional Experience:

- Mining Engineer
- Web Developer
- IT Consulting (Systems Analysis, ERP implementation)

Resources

- Bibliography
- Data Mining Github repo:
  - https://github.com/fpontejos/Data-Mining-23-24
- Class slides and Jupyter Notebooks
- Google, Stack Overflow, documentations, Github and YouTube

Resources

Recommended DataCamp courses

*Use the **invite link** (on Moodle) to get 6 months free access to all DataCamp features.*
*Make sure to **use your university email address** to sign up.*

Introduction to Python
https://app.datacamp.com/learn/courses/intro-to-python-for-data-science

Intermediate Python
https://app.datacamp.com/learn/courses/intermediate-python

Our working environment

- We will be using Anaconda: Currently one of the most popular Python distributions.
- Sets up a data science oriented working environment in Python
- It installs a set of libraries (for now, think of libraries as programming tools – like a toolbox in a woodshop)
- But it can be used for many different purposes (all it takes is installing the necessary libraries)

Anaconda

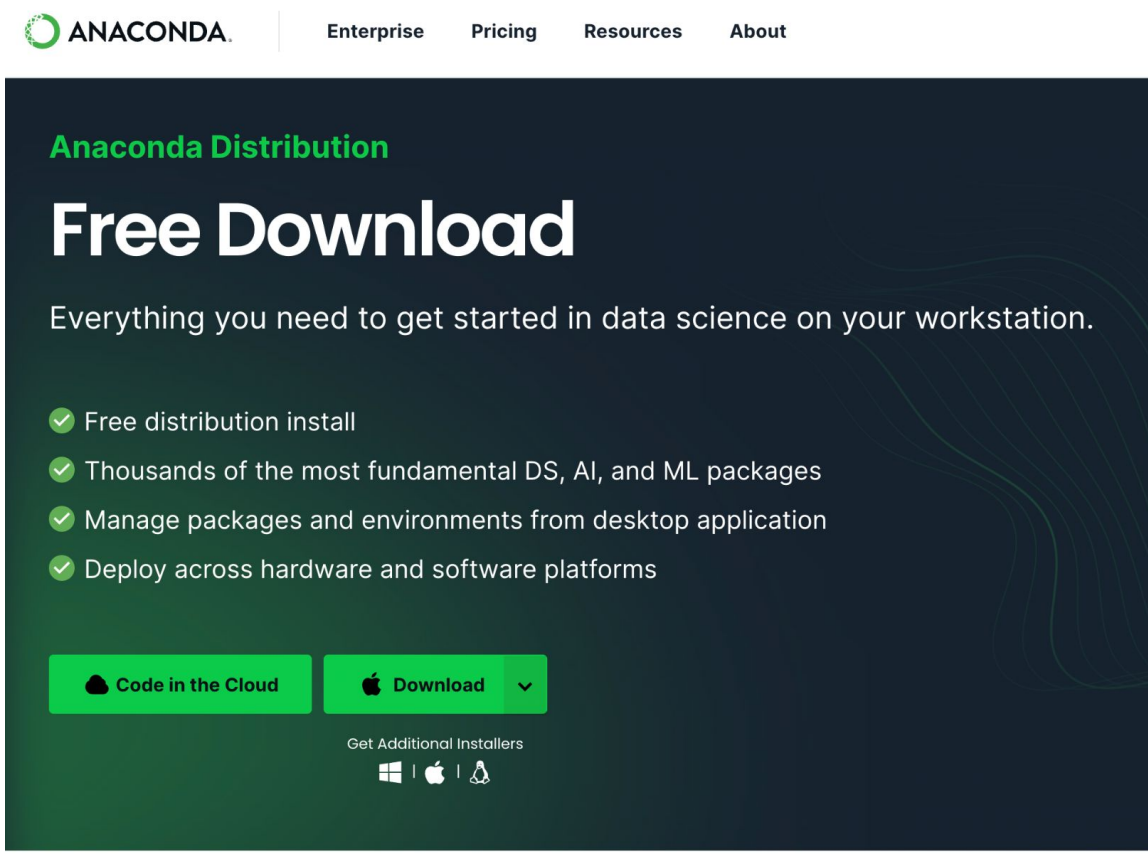
www.anaconda.org

- one of the most popular Python distributions for Data Science
- manages your packages and environments.
- reduce future issues dealing with the various libraries you will be using.
- comes with most of the main libraries for data manipulation
  - Pandas
  - Numpy
  - Matplotlib
  - Scipy
  - …
- easy to use and install

In the meantime: Download + Install Anaconda Navigator

Download:

https://www.anaconda.com/download



Install Anaconda Navigator

Please read the documentation applicable to your system:

https://docs.anaconda.com/free/anaconda/install/

> **Note**
>
> When installing Anaconda, you have the option to "Add Anaconda to my PATH environment variable." *This is not recommended* because it *appends* Anaconda to PATH. When the installer appends to PATH, it does not call the activation scripts.

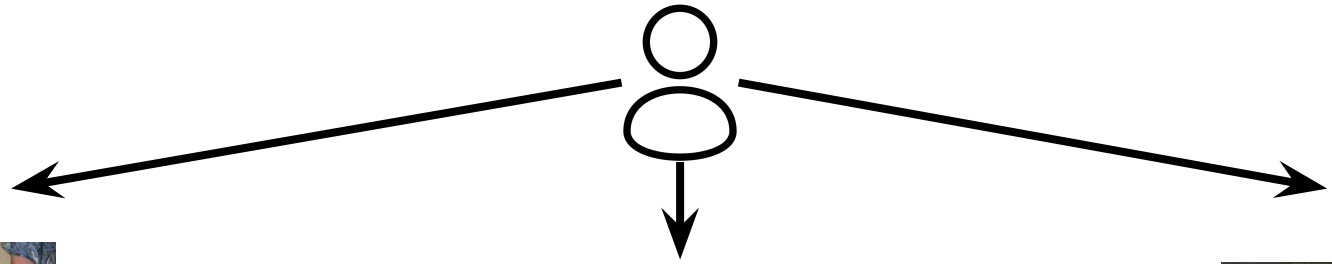
Virtual Environments

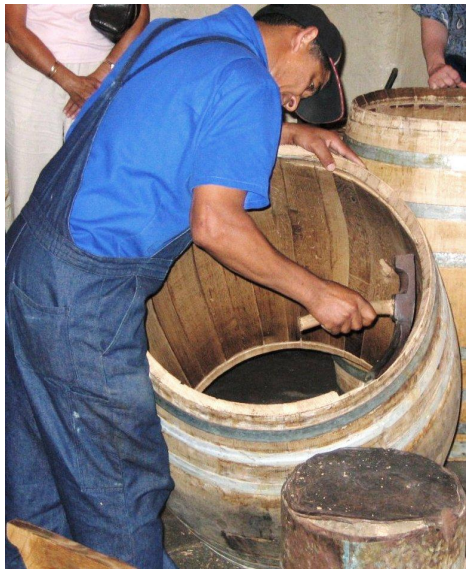
https://docs.conda.io/projects/conda/en/latest/user-guide/concepts/environments.html
https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html

- Isolated spaces that contain per-project dependencies (specific collection of installed conda packages)
- Using conda to manage environments:
  - Create, export, list, remove, and update environments
  - Switching or moving between environments (conda activate)
  - You can also share an environment file
- You can also use pip to manage environment

In other words

you may need to use different “versions” of the same types of tools, or entirely different “environments”





Cooper (making a barrel)

Formwork (setting a structure)

Luthier (making a guitar)

Icon: BUSAIRI from Noun Project | Cooper: CC BY-SA 2.0, https://commons.wikimedia.org/w/index.php?curid=45382 | Formwork: CC BY 2.5, https://en.wikipedia.org/w/index.php?curid=10166083 | Luthier: By Jo Dusepo - Own work, CC BY-SA 4.0

In other words

Suppose you will build a Caravel (a Portuguese ship from the 15th century).

However, you are also a Cooper!

To build the Caravel the way they did back then, you will need a specific set of tools, much more rudimentary than the ones you will have at your own workshop.

In other words

- You will need to get them first (i.e., “download” them)
- However, you should not mix these tools with the ones you already have!
  - They are not appropriate to build barrels, and the ones you already have are not appropriate to build Caravels
  - They will create clutter in your workshop (unnecessarily keeping unused packages)
  - You will have duplicate tools with different (version conflicts)
  - Other carpenters may want to build their own replica of you project (reproducibility)
  - If you are working with other carpenters, they will need to use the same types of tools you are using (collaboration)
- These tools (i.e., libraries) and their versions should be specified in the project’s requirements (including the Python version)!

NOVA IMS Information Management School

Python Packages

NetworkX
Network Analysis in Python

seaborn

Git and GitHub

https://guides.github.com/activities/hello-world/

https://docs.github.com/en/github/getting-started-with-github

- What is GitHub?
  - Code hosting platform for version control and collaboration
- What is Git?
  - At the heart of GitHub is an open source version control system (VCS) called Git. Git is responsible for everything GitHub-related that happens locally on your computer.
- Why Git and GitHub?
  - Optional: You can use Git and GitHub for collaborating and version control in your projects.
  - Also we have a GitHub repository with all the practical class contents:
  - https://github.com/fpontejos/Data-Mining-23-24

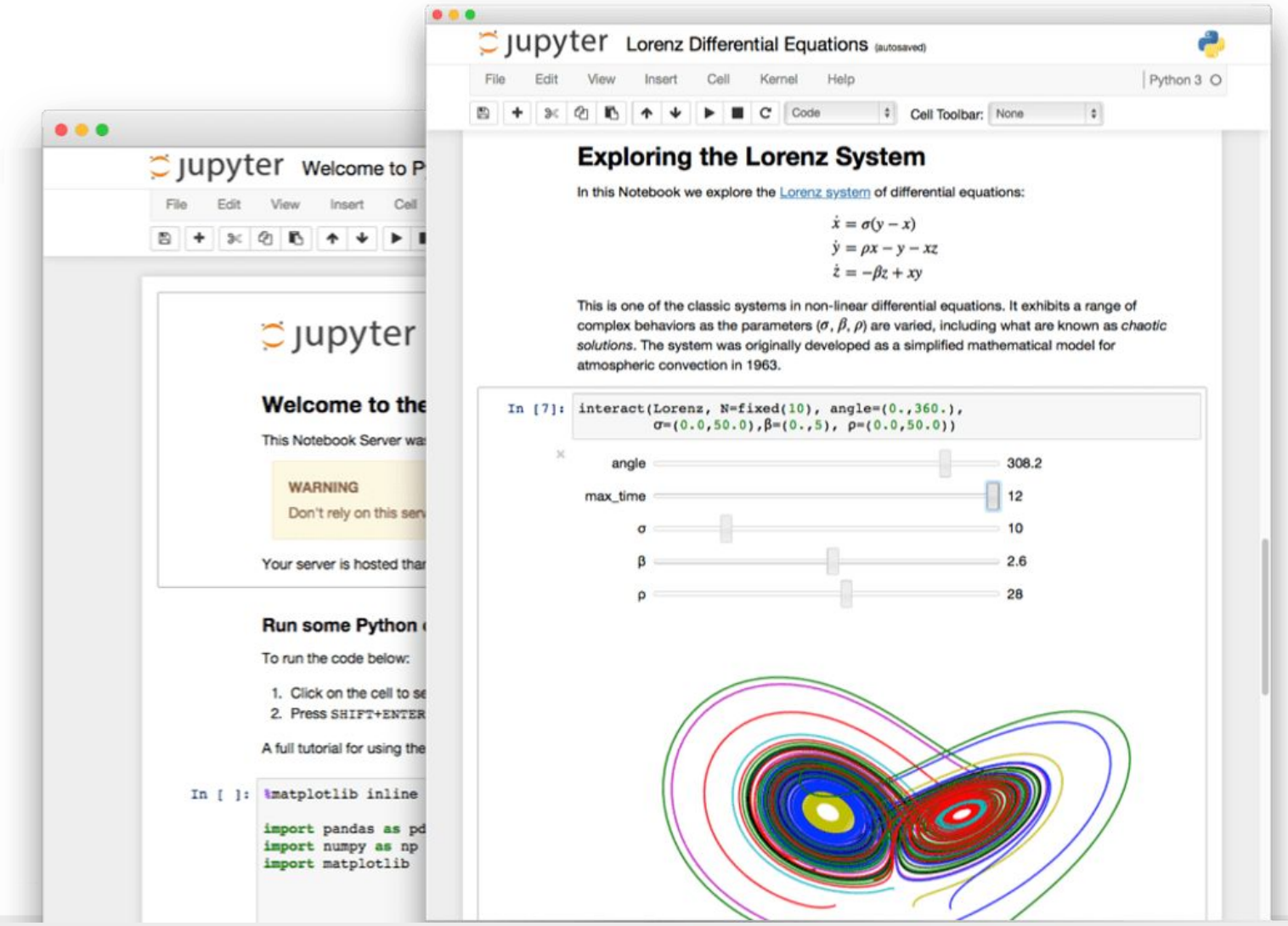
Main ways to access Python

- Python Shell and IPython
  - An interactive environment for writing and running code
- Jupyter Notebooks
  - A notebook that weaves code, data, prose, equations, analysis, and visualization
  - A tool for prototyping new code and analysis
  - A method for creating a reproducible workflow for scientific research
- IDE (Integrated Development Environment):
  - A software that helps you build code

Jupyter notebooks

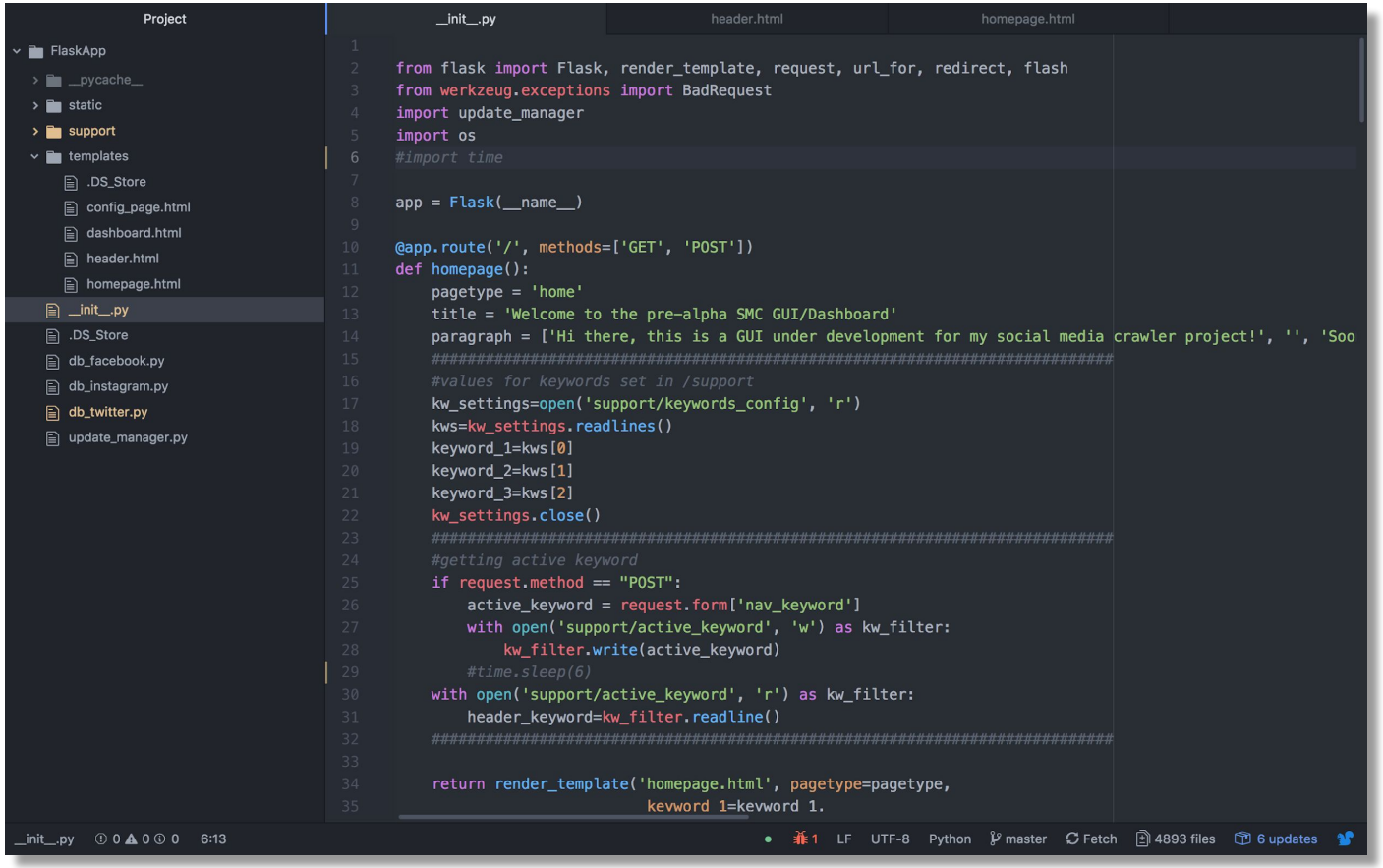
https://jupyter.org/

We will be using Jupyter notebooks for our practical sessions.



Text Editors

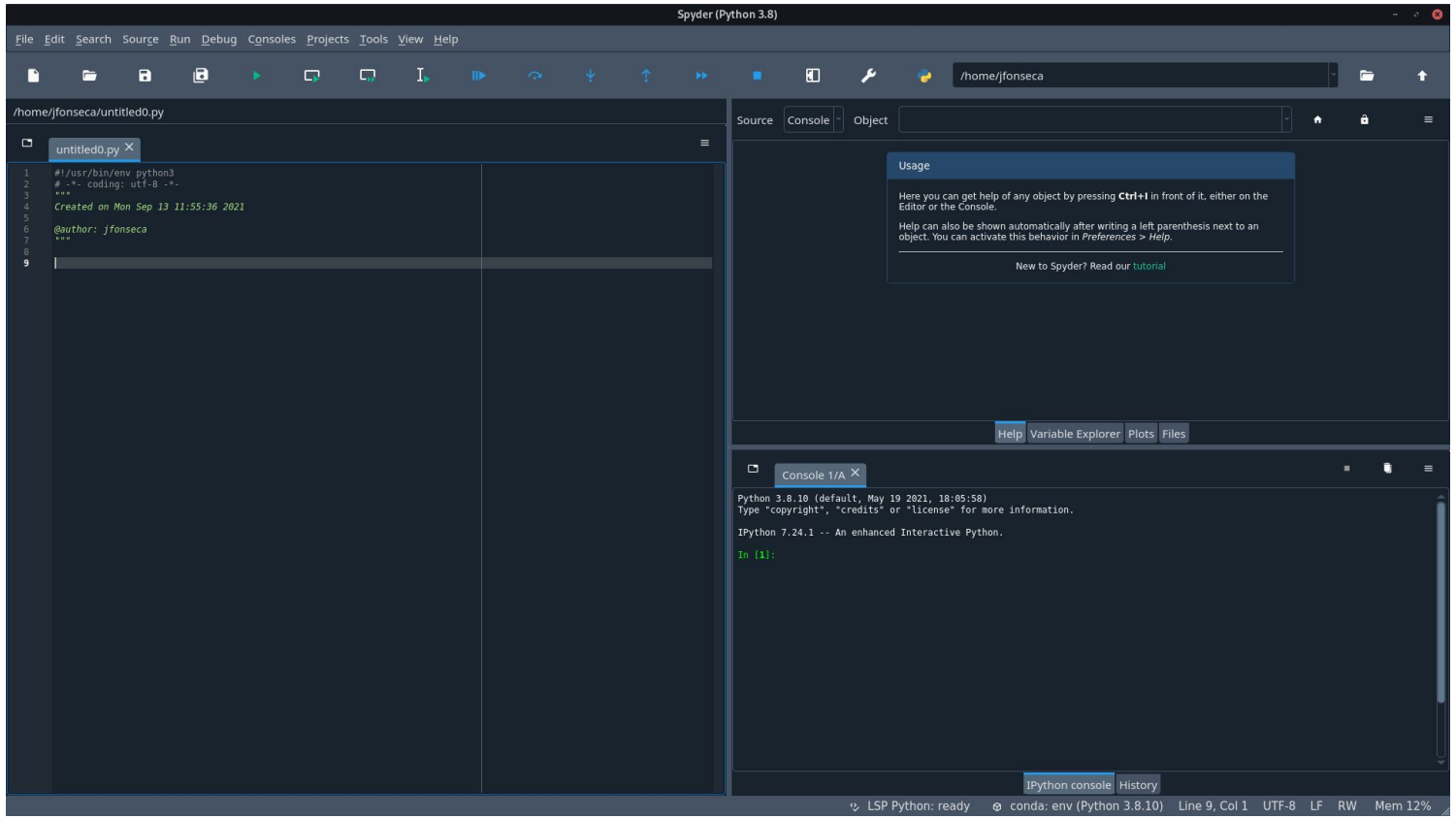
- Another method to write python scripts is using text editors
- Some popular text editors:
  - Vim (Linux terminal text editor)
  - Atom (popular open source editor)
  - Sublime Text (popular proprietary text editor)
  - Notepad ++ (Windows only)
- Usually highly customizable



Atom Text Editor

Integrated Development Environment (IDE)

- Popular IDE's:
  - Spyder
  - PyCharm
  - VSCode
  - Rodeo
- Anaconda comes with Spyder and VSCode



Usage of IDE and/or Text editor (and which ones to use) comes down to personal preference

Data Mining Project

- Form groups on Moodle **(up to 3 students)**
- Project guidelines will be released on Moodle
- Anonymized data from a real-world scenario
- Your goal is to develop a **Customer Segmentation** in such a way that it will be possible for the Marketing Department to better understand all the different Customers' Profiles.
- **This will be discussed in more detail later.**

Let’s get started!

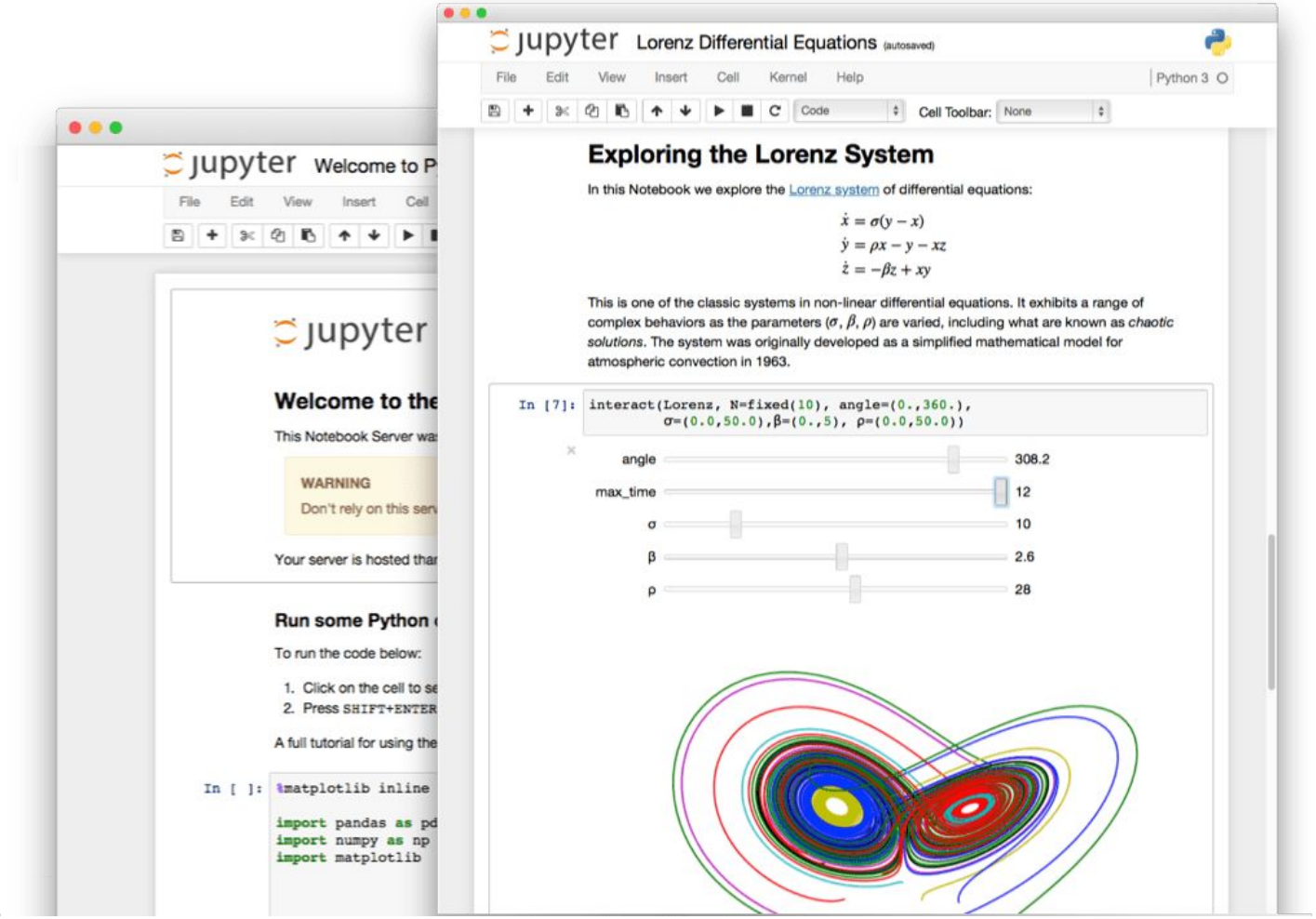
Next: Setting up our tools

The Jupyter Notebook

http://jupyter.org/

Let's try it out!

- Install and Open Anaconda Navigator
- Start Jupyter Notebook



Setting up our tools

Two options:

1. **With GUI (Anaconda Navigator)**
2. Command line (miniconda) (skip to slide 64)

Setting up our tools

Download lab files:

- `environment_dm.yml`
- `requirements_pip.txt`

Download installers:

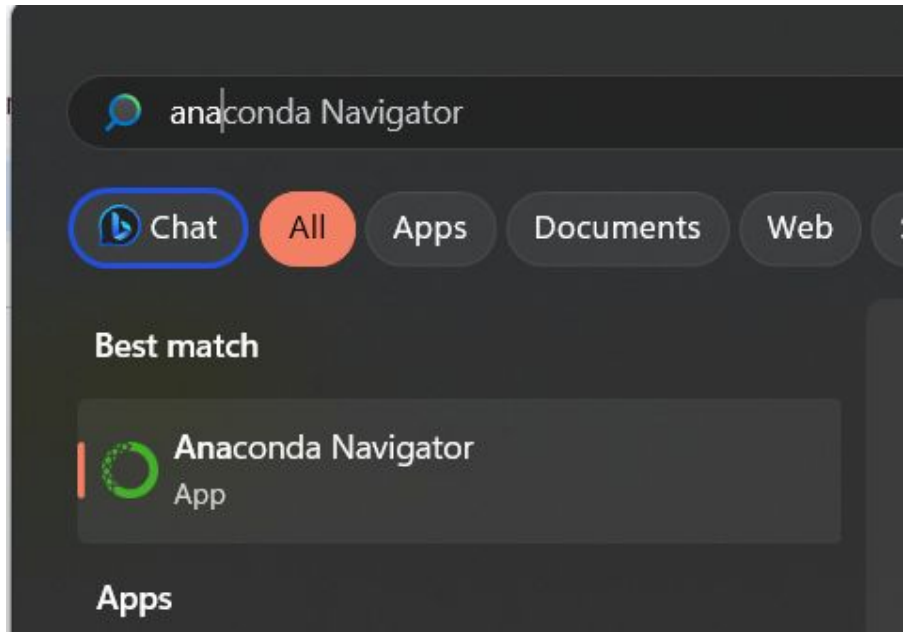
- Instructions on Moodle

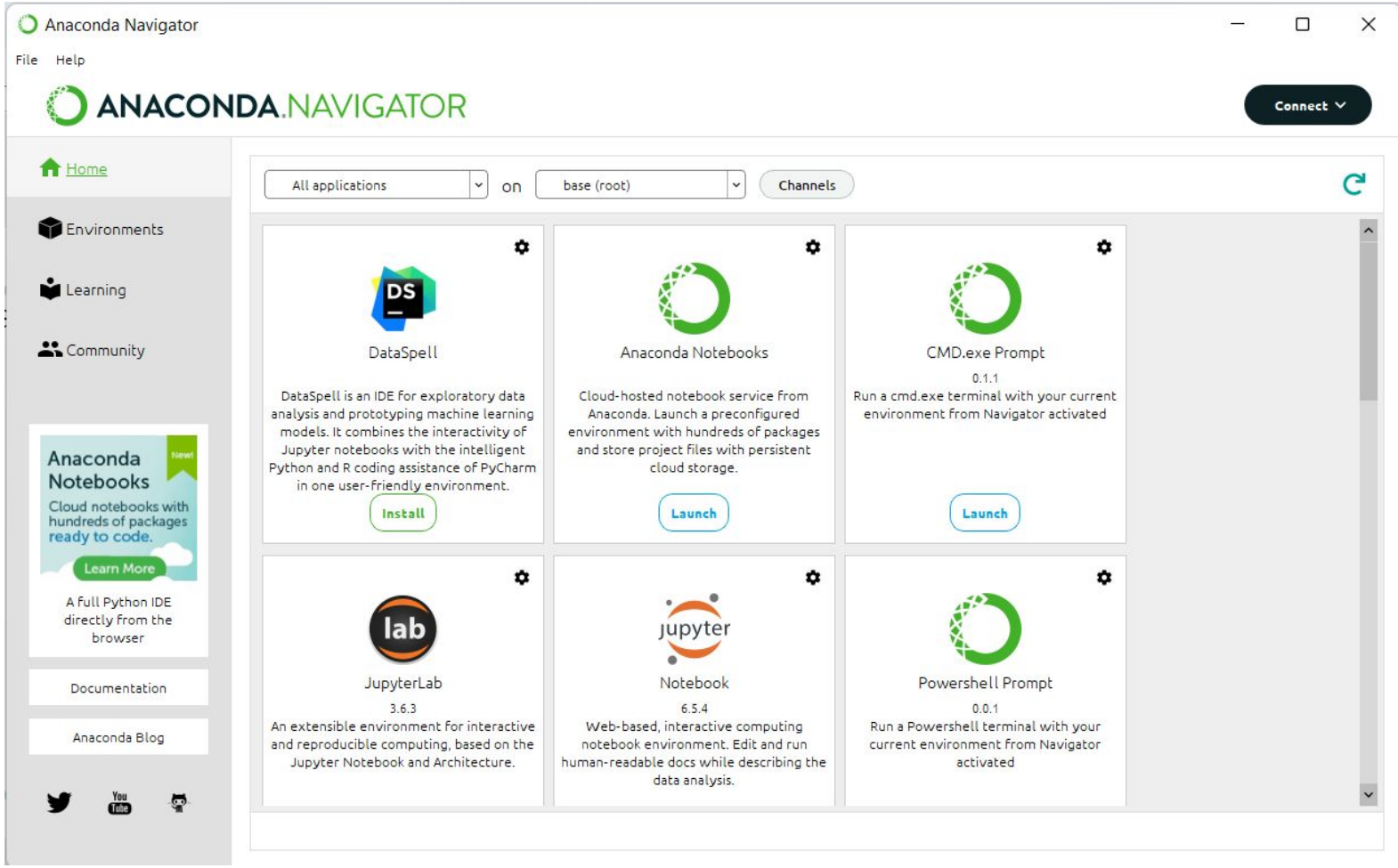
If you downloaded a zip file, unzip it to your **Downloads** folder

- This makes it easier to find later

Setting up our tools

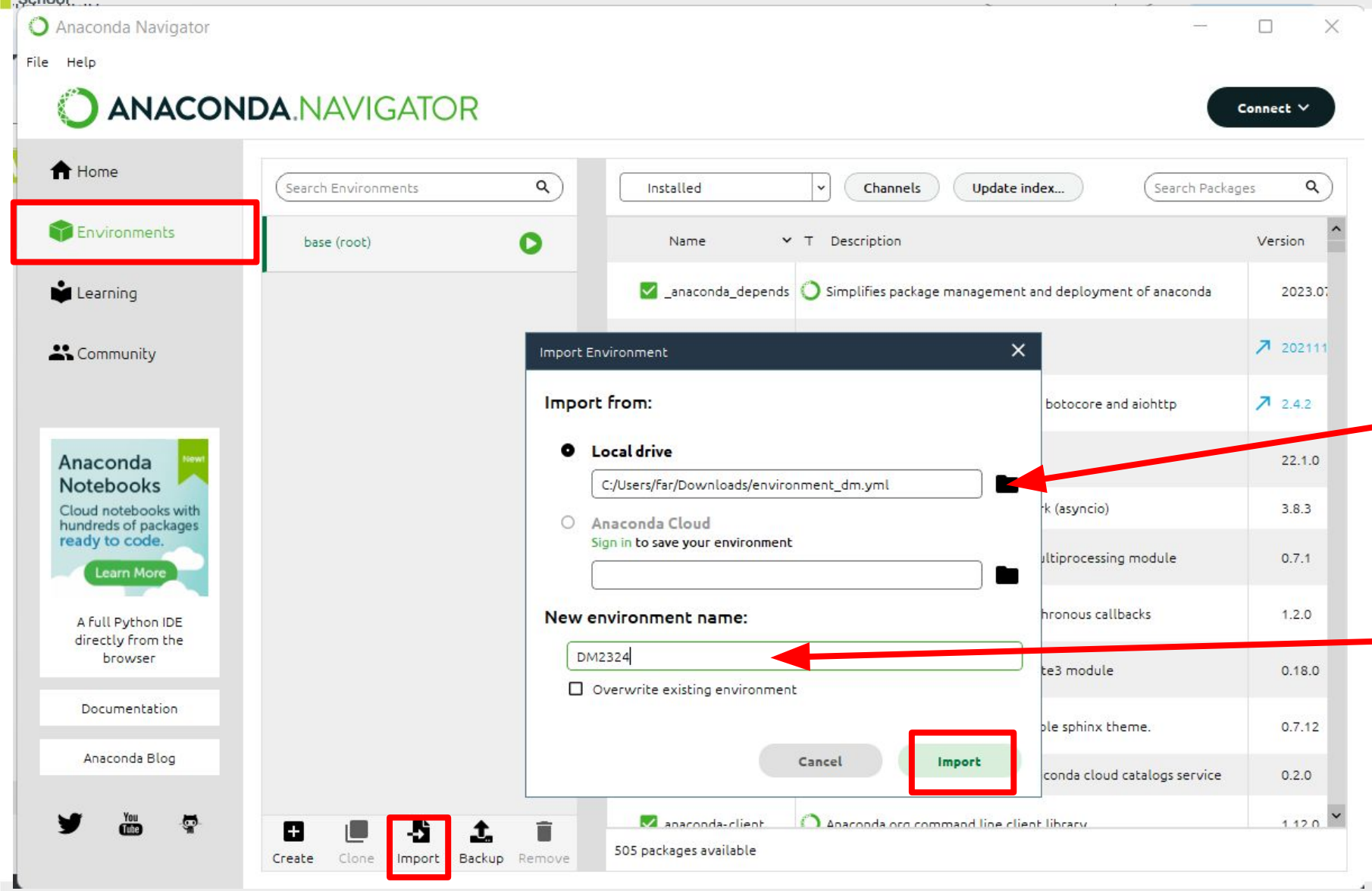
Load Anaconda Navigator





Import environment

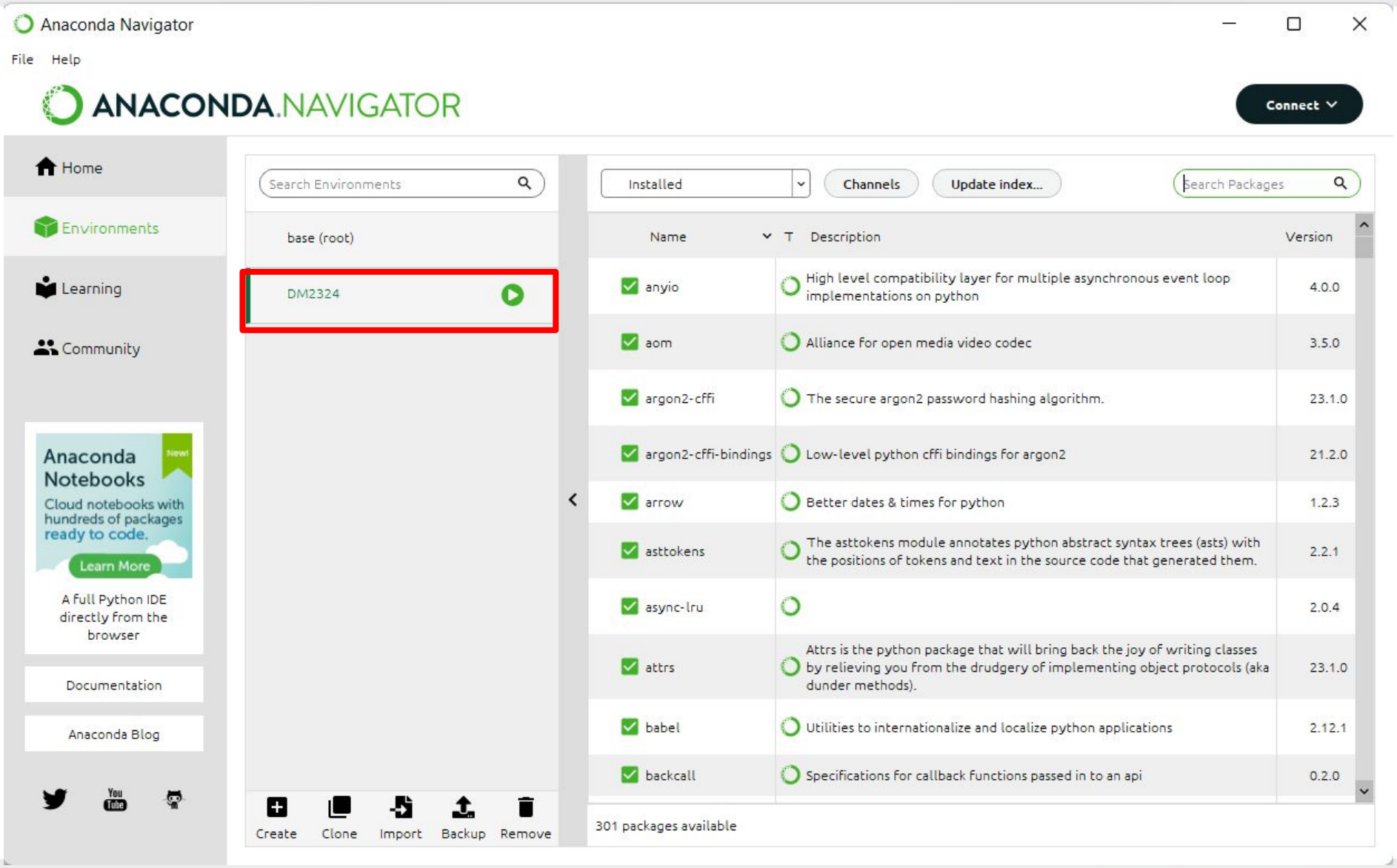
!!! This may take some time !!!



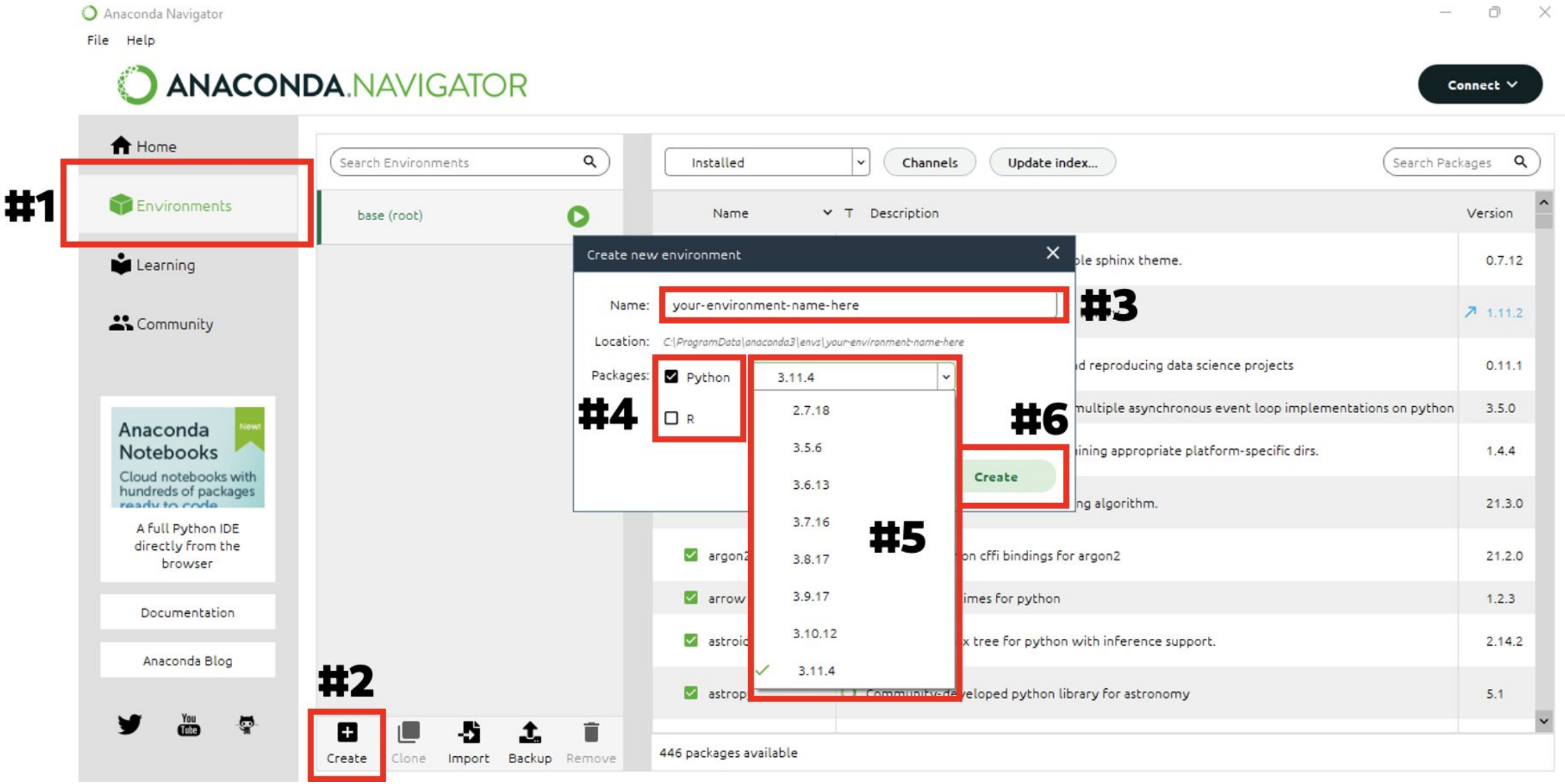
Where you have the file:
`environment_dm.yml`

Name the environment:
`DM2324`

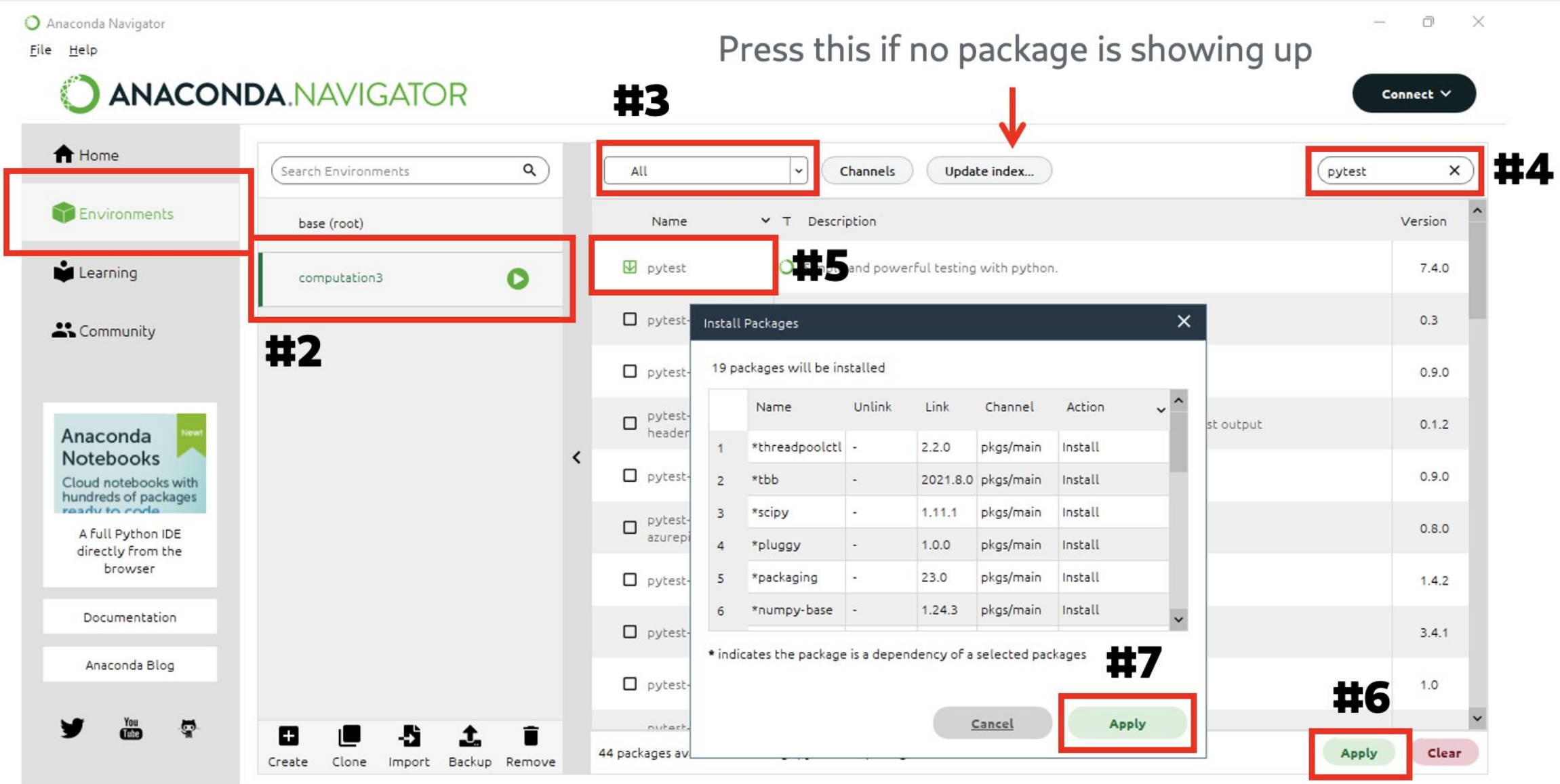
Import environment: Done



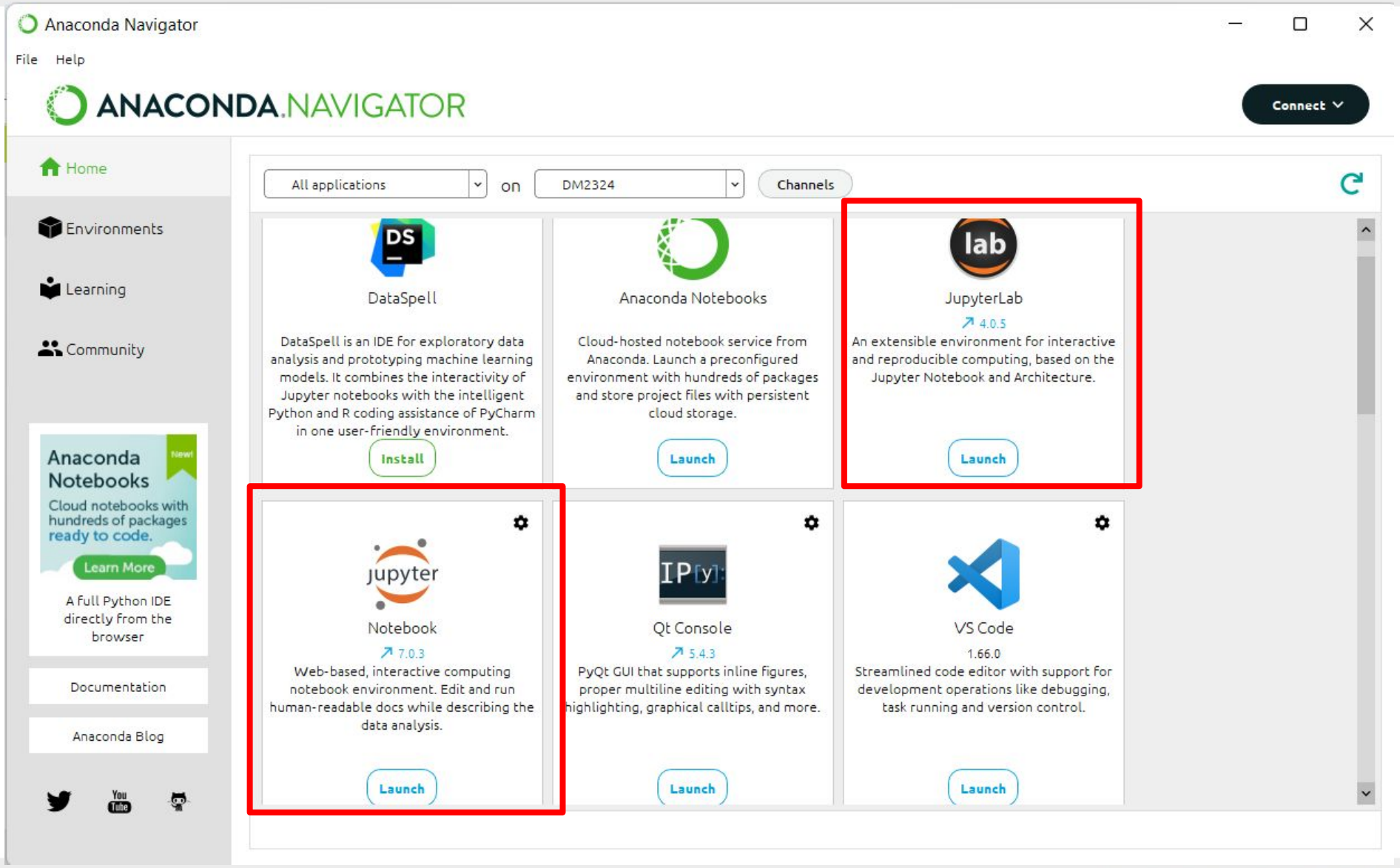
You can also create an environment from scratch



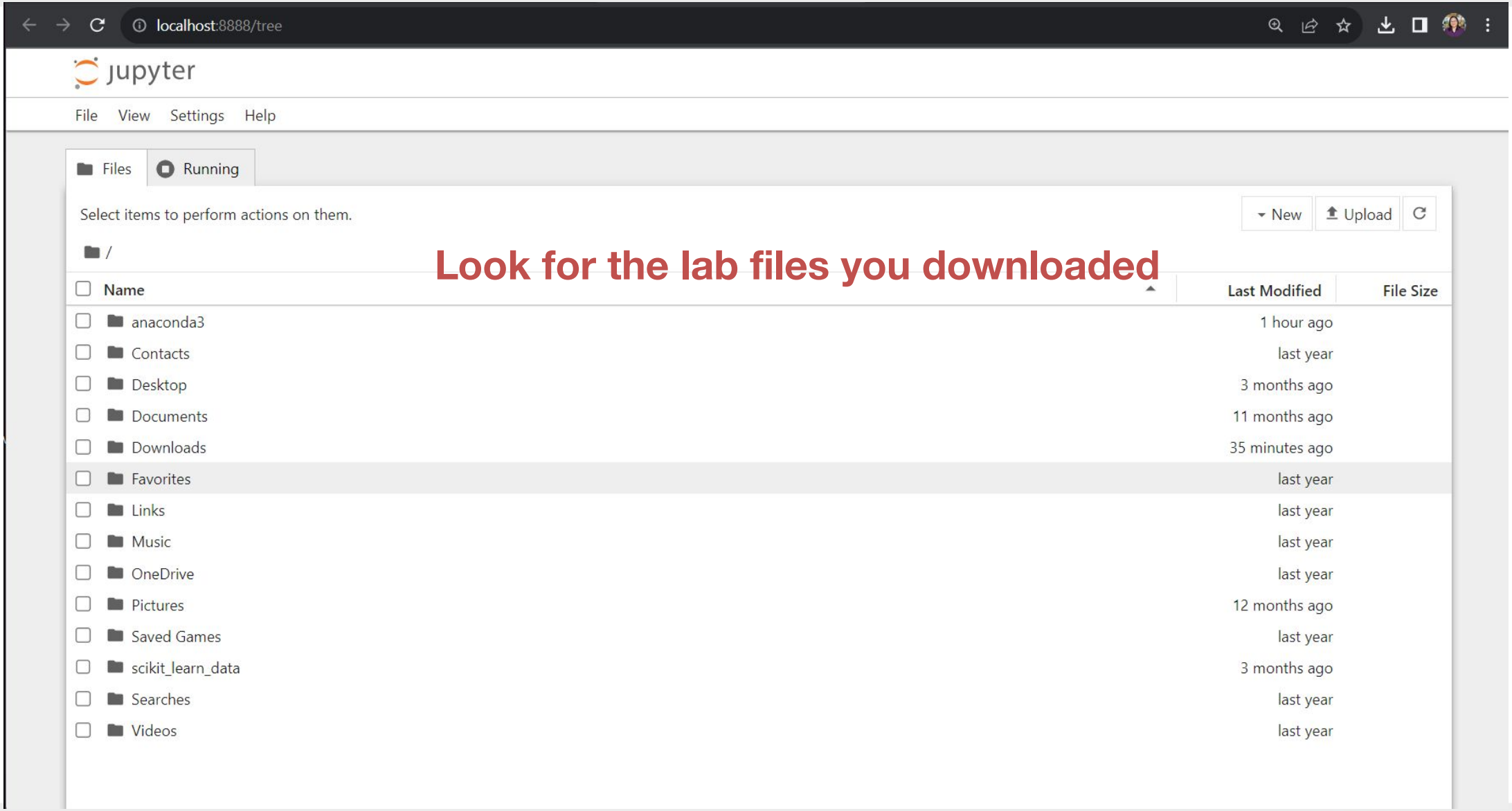
You can also install packages one by one



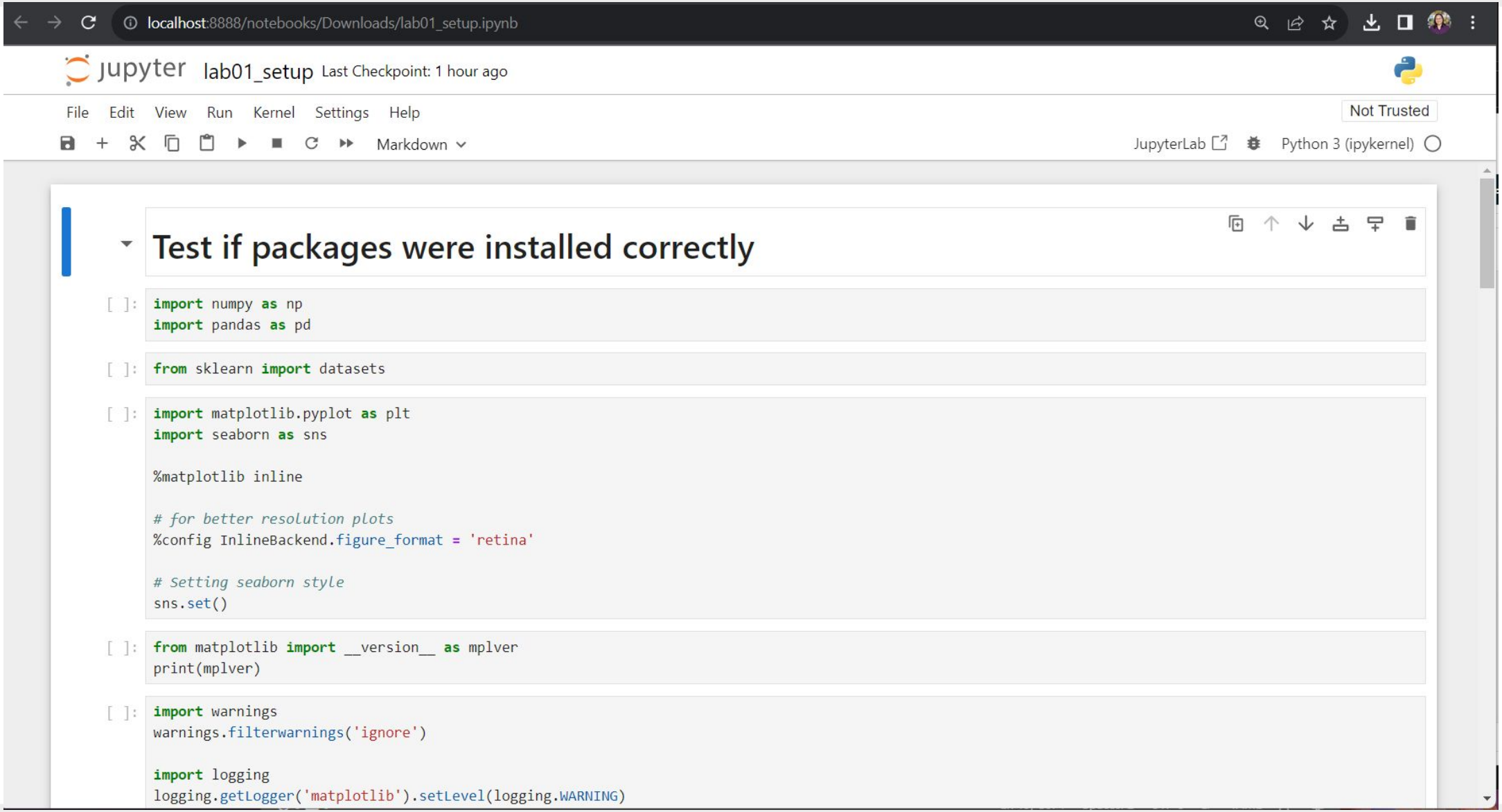
Test loading Jupyter notebook



Test loading Jupyter notebook



Load the lab01_setup.ipynb notebook file

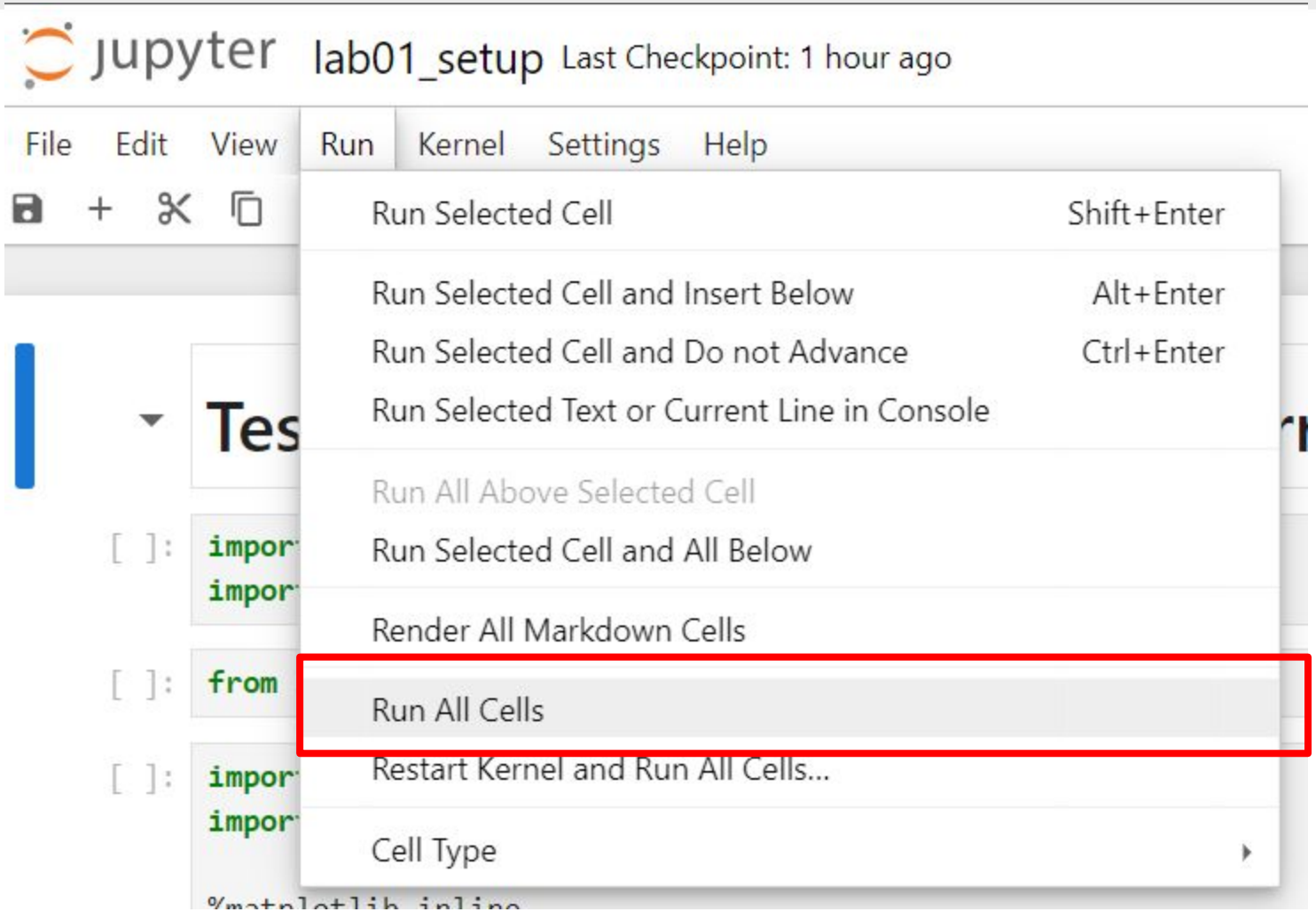


Don’t worry about understanding the code at this point

Right now we just want to make sure that all the packages we need are installed and work correctly

Run all cells

You will reach a part in the notebook that will give an error. We still need to install some additional packages.



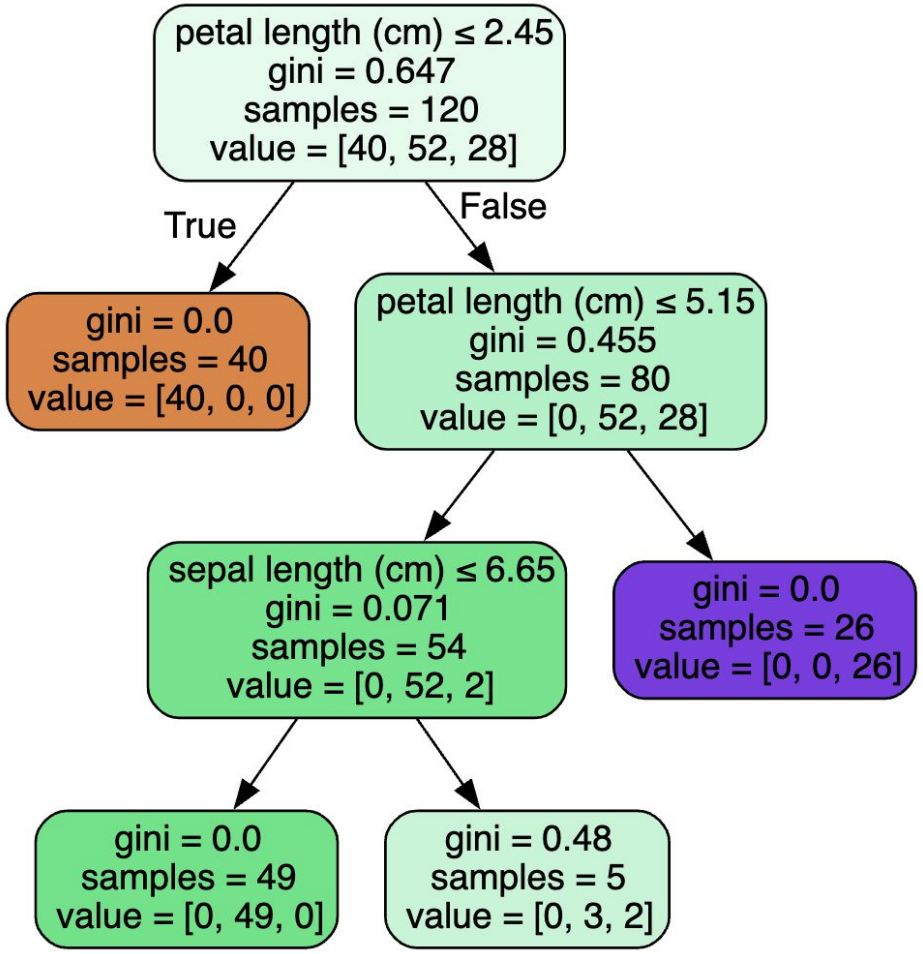
Install other packages

Some of these packages may require more steps to install depending on your system

Install other packages: graphviz

The graphviz package allows us to visualize graphs, trees, networks etc.

We will use it to visualize the results of a Decision Tree Classifier.



Install other packages: graphviz

Mac users:
(assuming you have homebrew installed; if not, install homebrew first)

Open Terminal (Applications > Utilities > Terminal)

```
brew install graphviz
```

Install other packages: graphviz

Ubuntu/Debian users

Open Terminal

```
sudo apt-get update && sudo apt-get install graphviz
```

Install other packages: graphviz

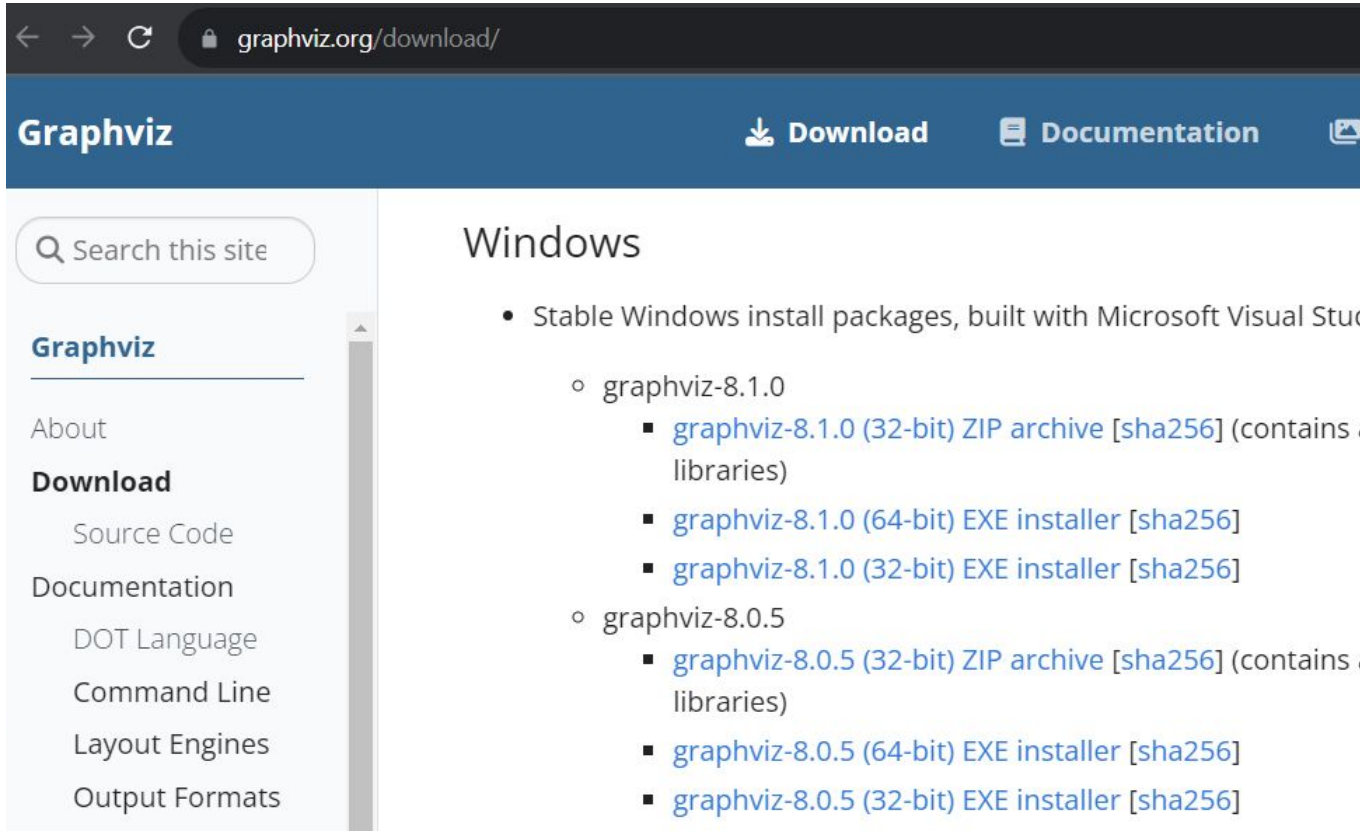
Windows users:

https://stackoverflow.com/questions/35064304/runtimeerror-make-sure-the-graphviz-executables-are-on-your-systems-path-aft

Install other packages: graphviz

Windows users:
download graphviz from

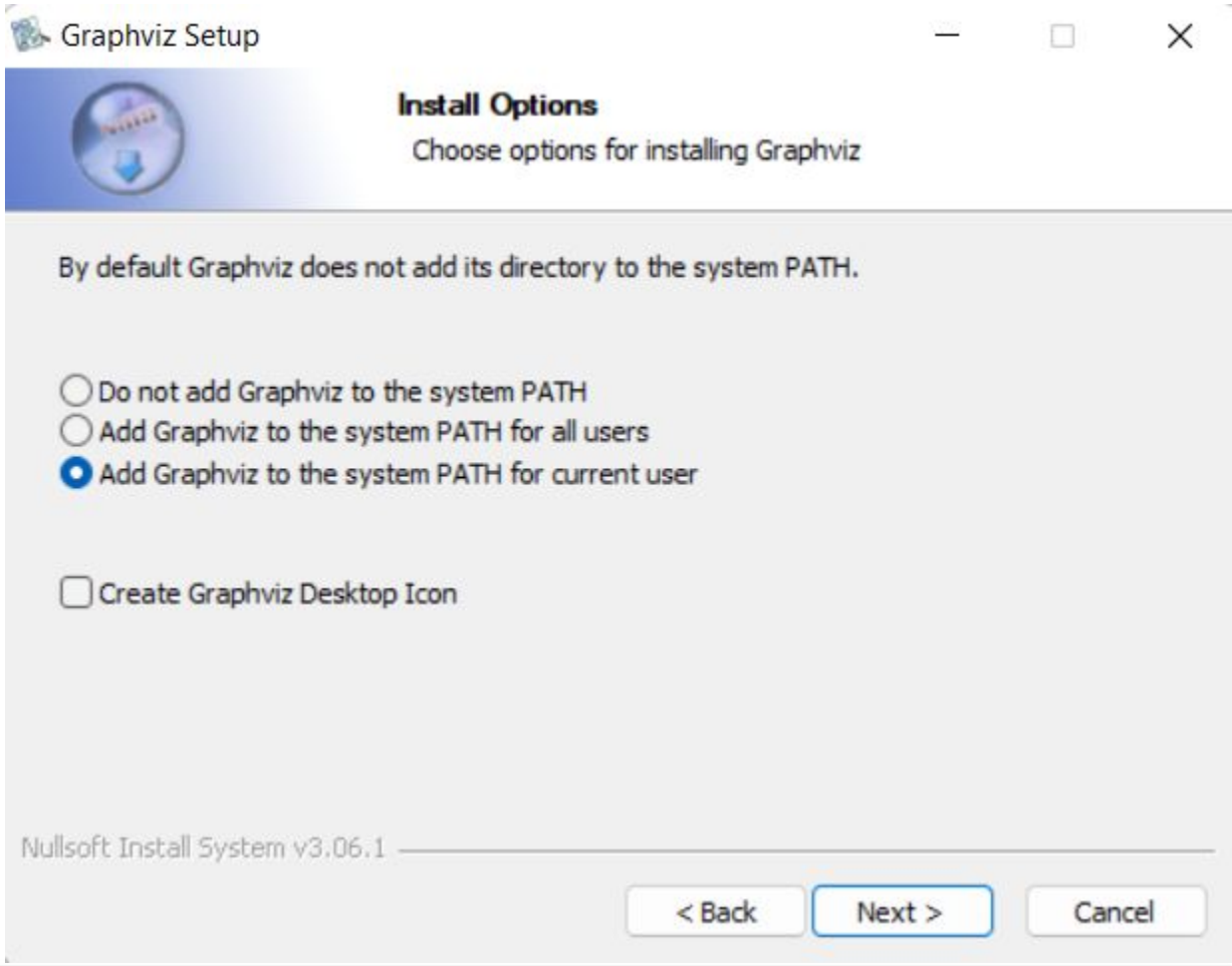
https://graphviz.org/download/



Install other packages: graphviz

Windows users:
download graphviz from

https://graphviz.org/download/



Install other packages: graphviz

Windows users:

Follow instructions here to add graphviz / dot to PATH

https://stackoverflow.com/questions/35064304/runtimeerror-make-sure-the-graphviz-executables-are-on-your-systems-path-aft

Install other packages: graphviz

Windows users:

(This is step 1, we'll do step 2 later)

26

Step 1: Install Graphviz binary

> **Windows:**

1. Download Graphviz from http://www.graphviz.org/download/
2. Add below to PATH environment variable (mention the installed graphviz version):
   - C:\Program Files (x86)\Graphviz2.38\bin
   - C:\Program Files (x86)\Graphviz2.38\bin\dot.exe
3. Close any opened Juypter notebook and the command prompt
4. Restart Jupyter / cmd prompt and test

> **Linux:**

1. sudo apt-get update
2. sudo apt-get install graphviz
3. or build it manually from http://www.graphviz.org/download/

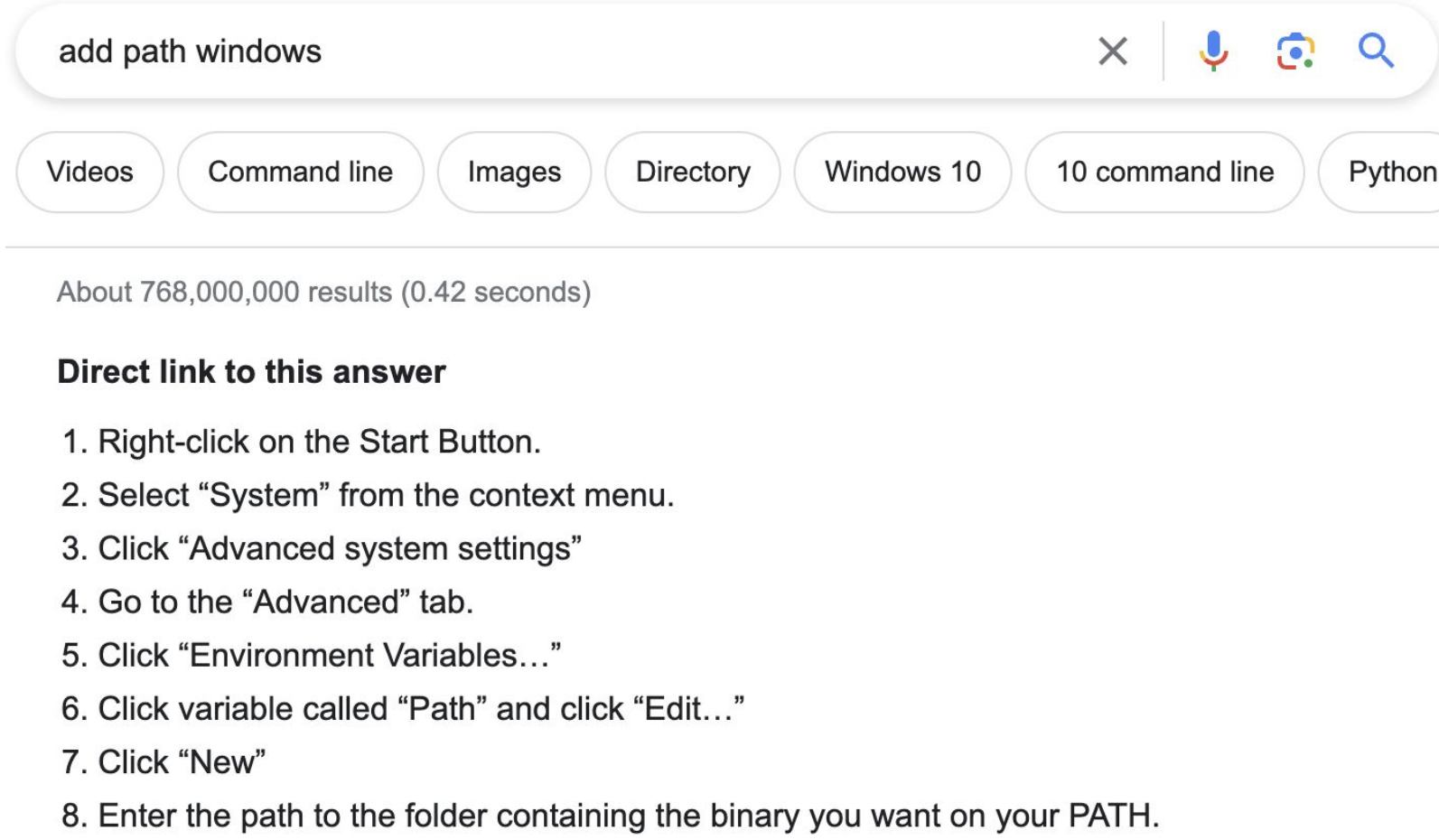
Step 2: Install graphviz module for python

> **pip:**

- pip install graphviz

Install other packages: graphviz

Windows users:



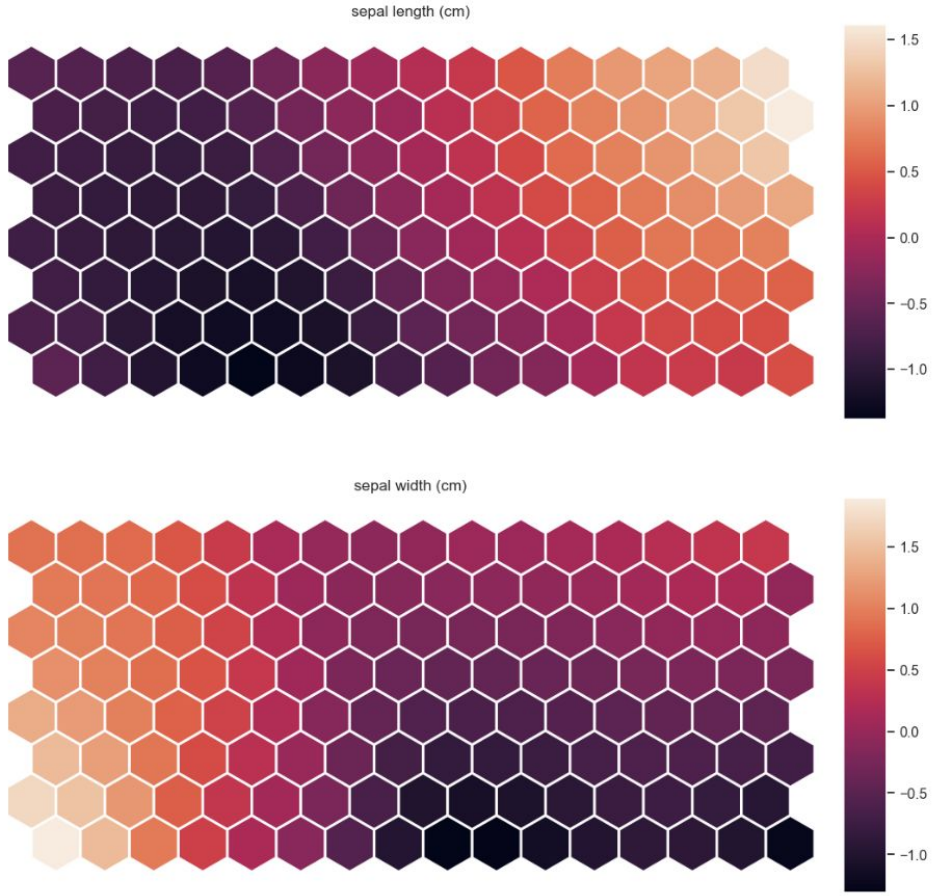
Does everyone have graphviz installed now?

Install Python packages: sompy

The sompy package allows us to train and visualize Self-Organizing Maps.

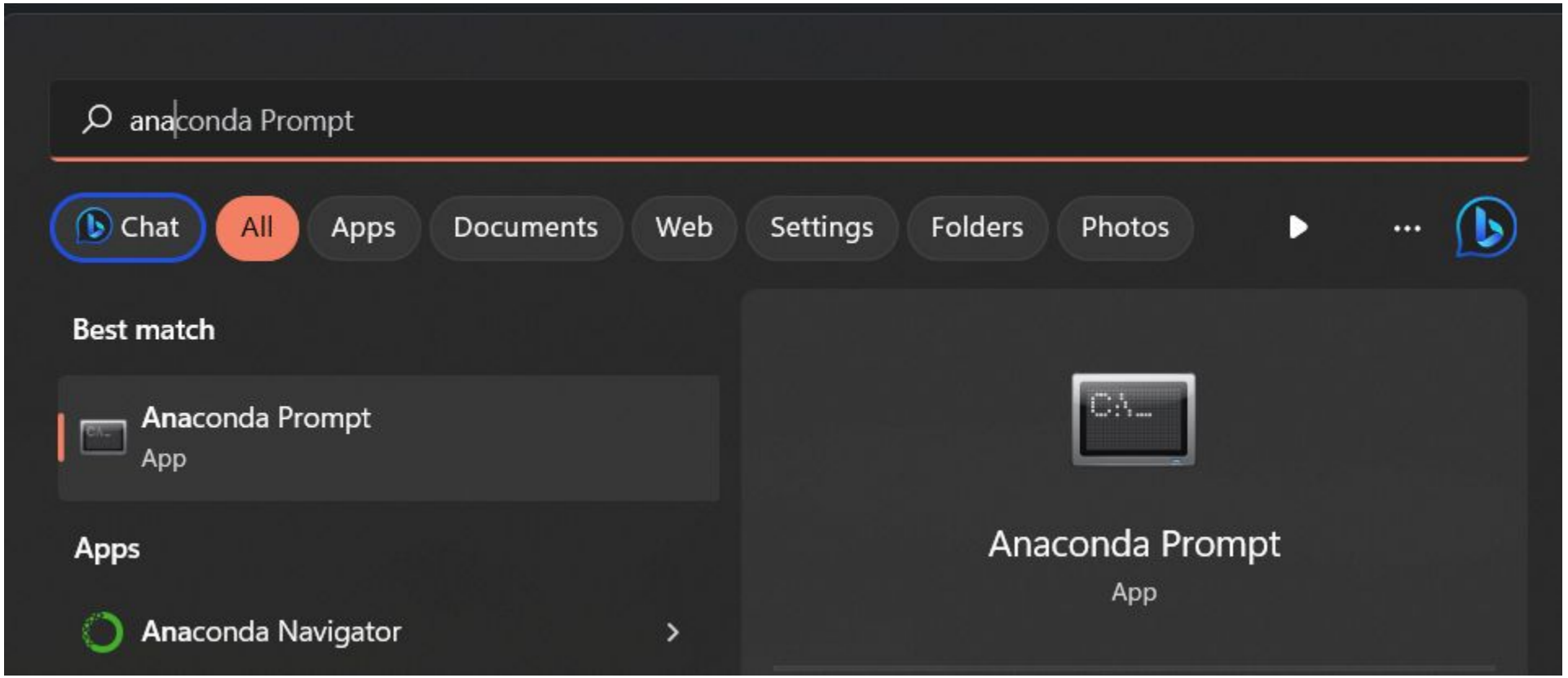
```
sns.set()
view2D = View2D(28, 46, "", text_size=150)
view2D.show(sm, col_sz=1, what='codebook')

plt.show()
```



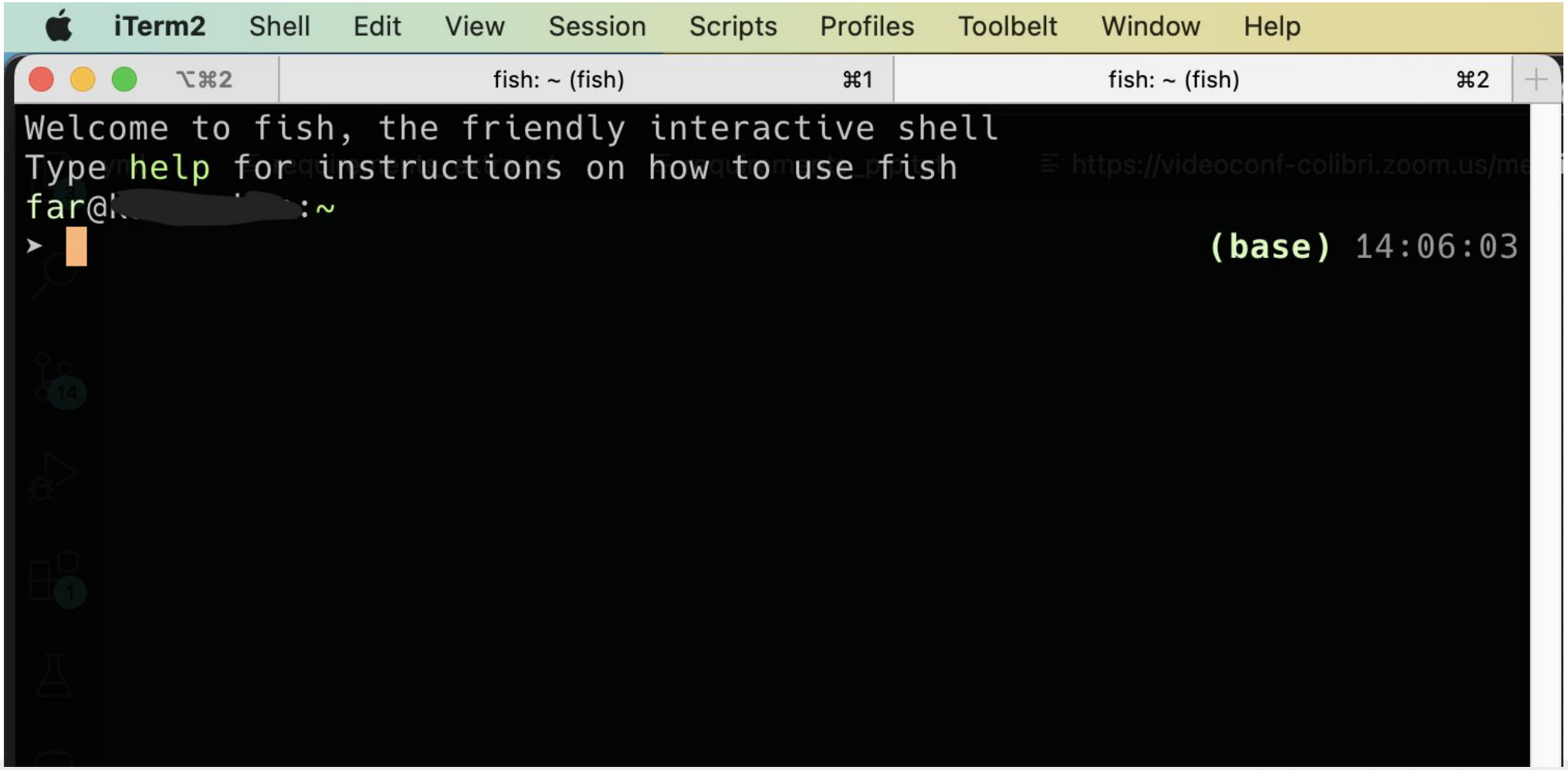
Install Python packages: graphviz/sompy

Windows: Open Anaconda Prompt



Install Python packages: graphviz/sompy

Mac / Linux: Open Terminal



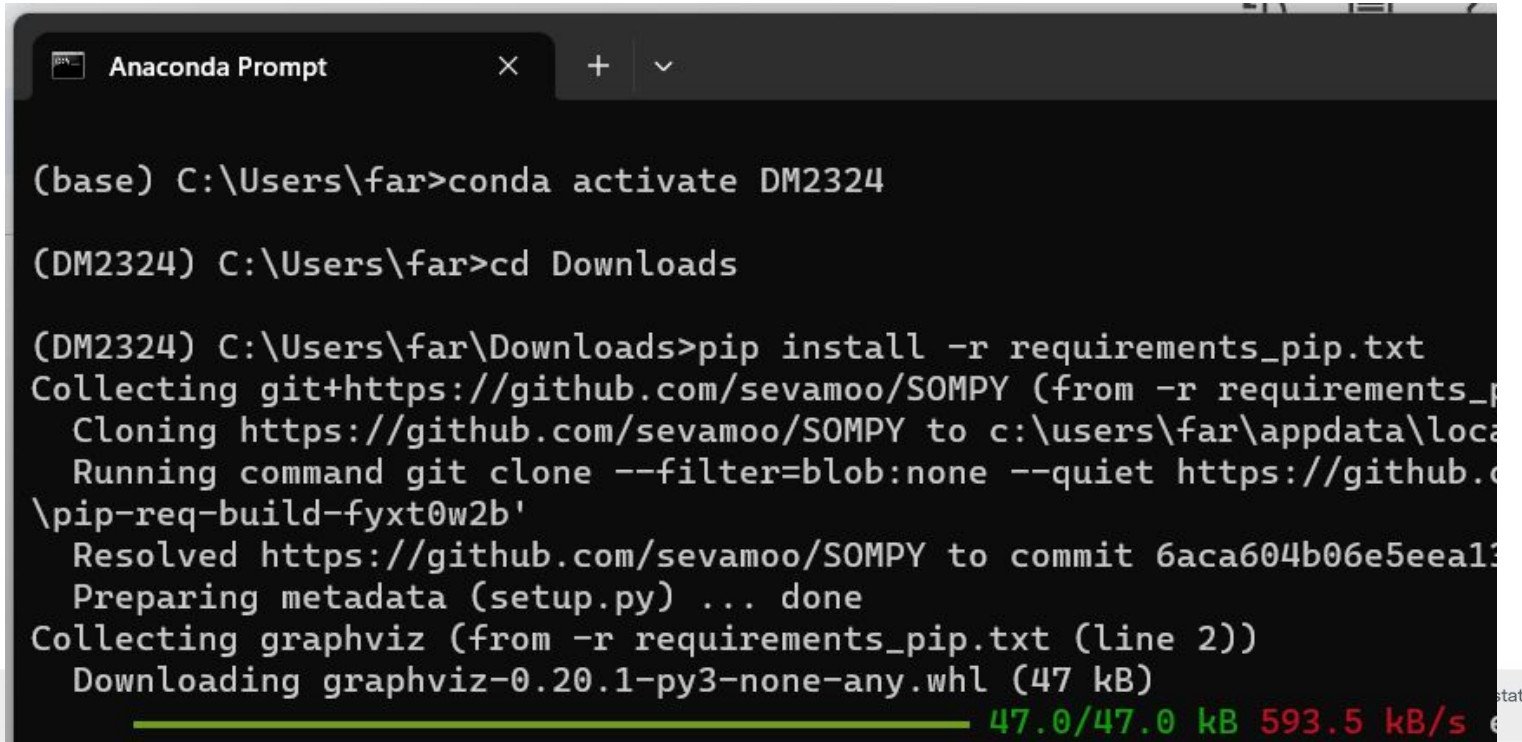
Install Python packages: graphviz/sompy

```
conda activate DM2324
cd Downloads
pip install -r requirements_pip.txt
```

Activate the environment we created earlier

Navigate to where you have the downloaded files

Install packages with pip



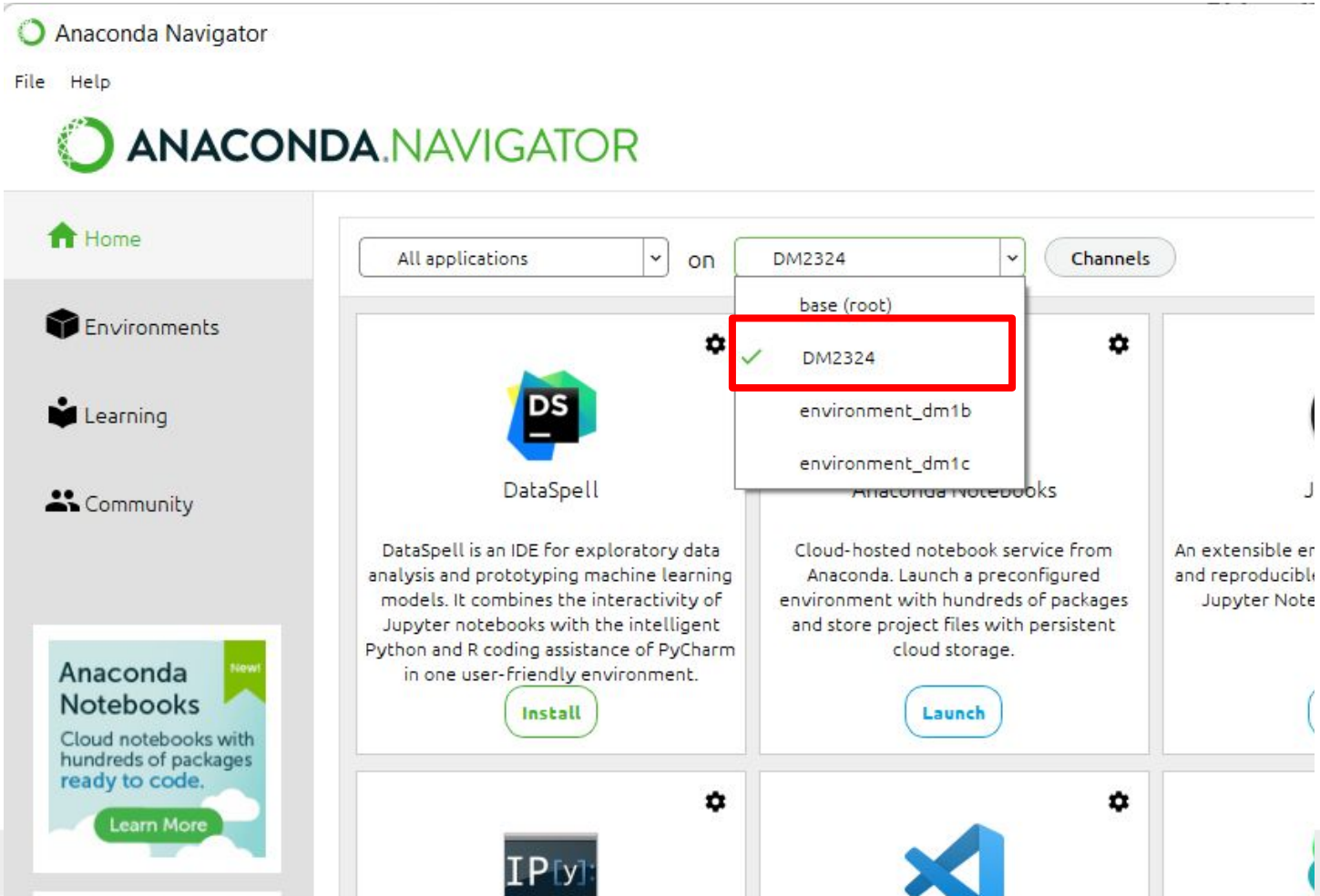
Test Jupyter notebook

Close all Terminal windows / Anaconda Navigator / Anaconda Prompt

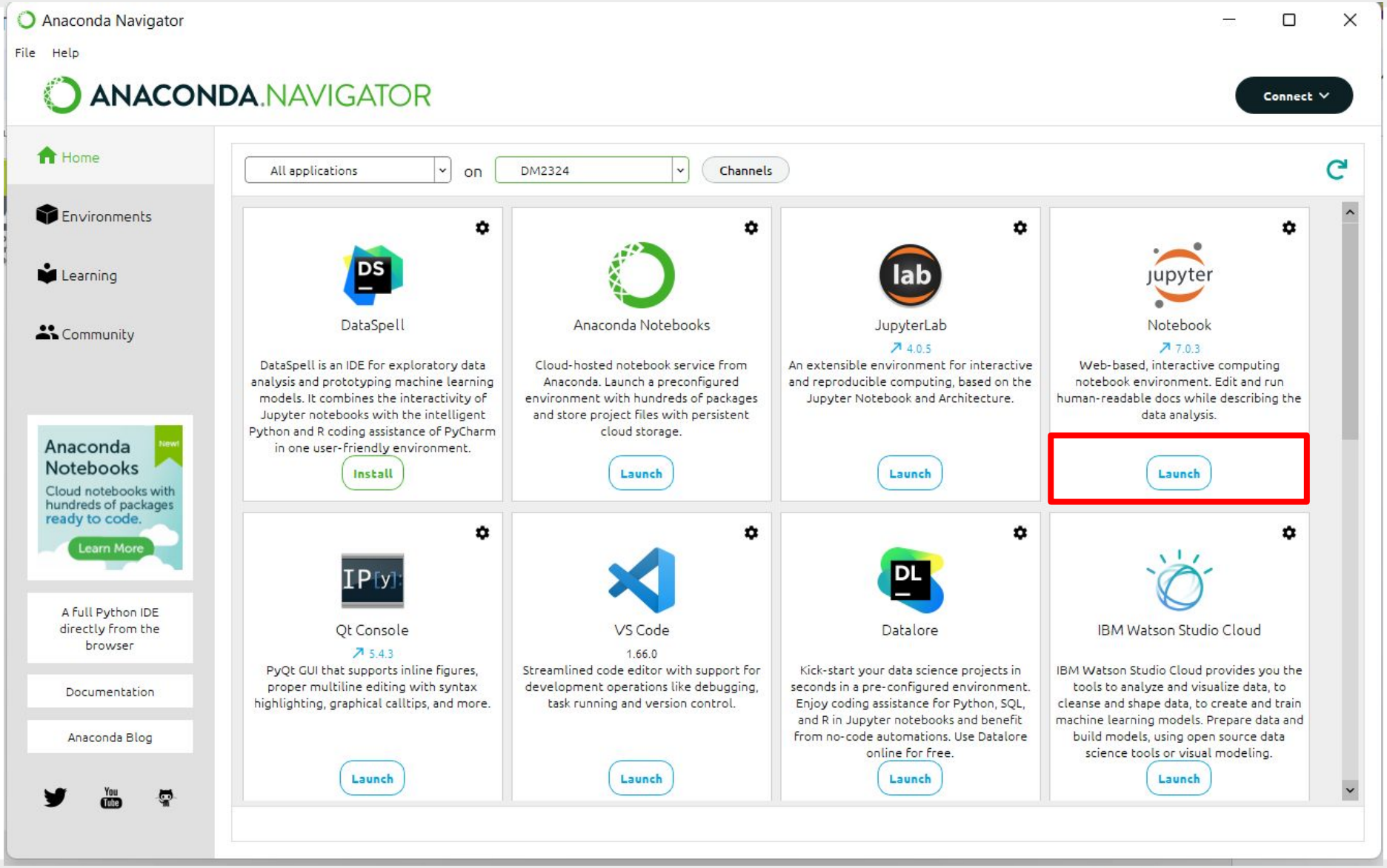
Test Jupyter notebook

Open Anaconda Navigator

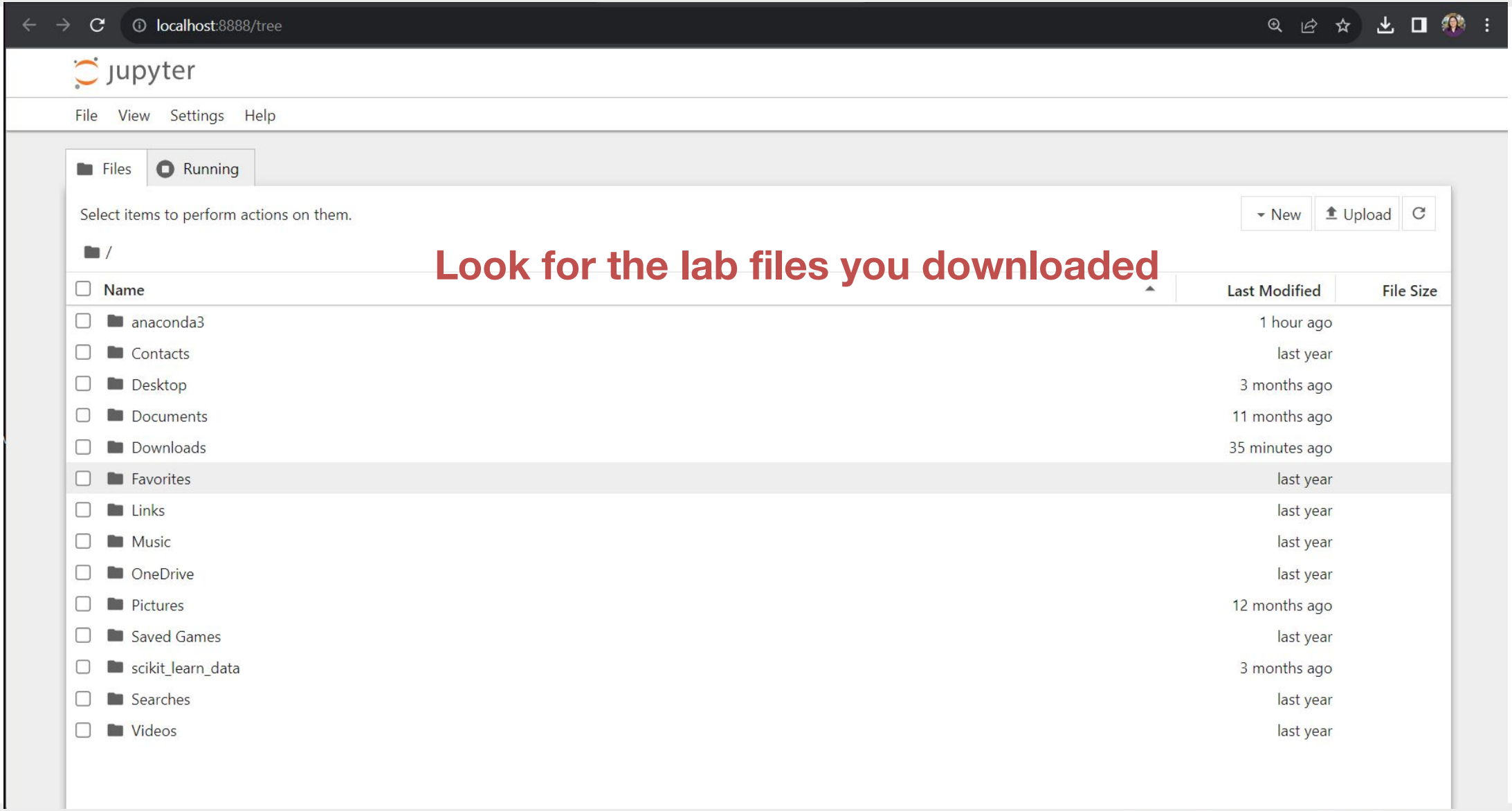
Make sure you select the environment we created



Test Jupyter notebook



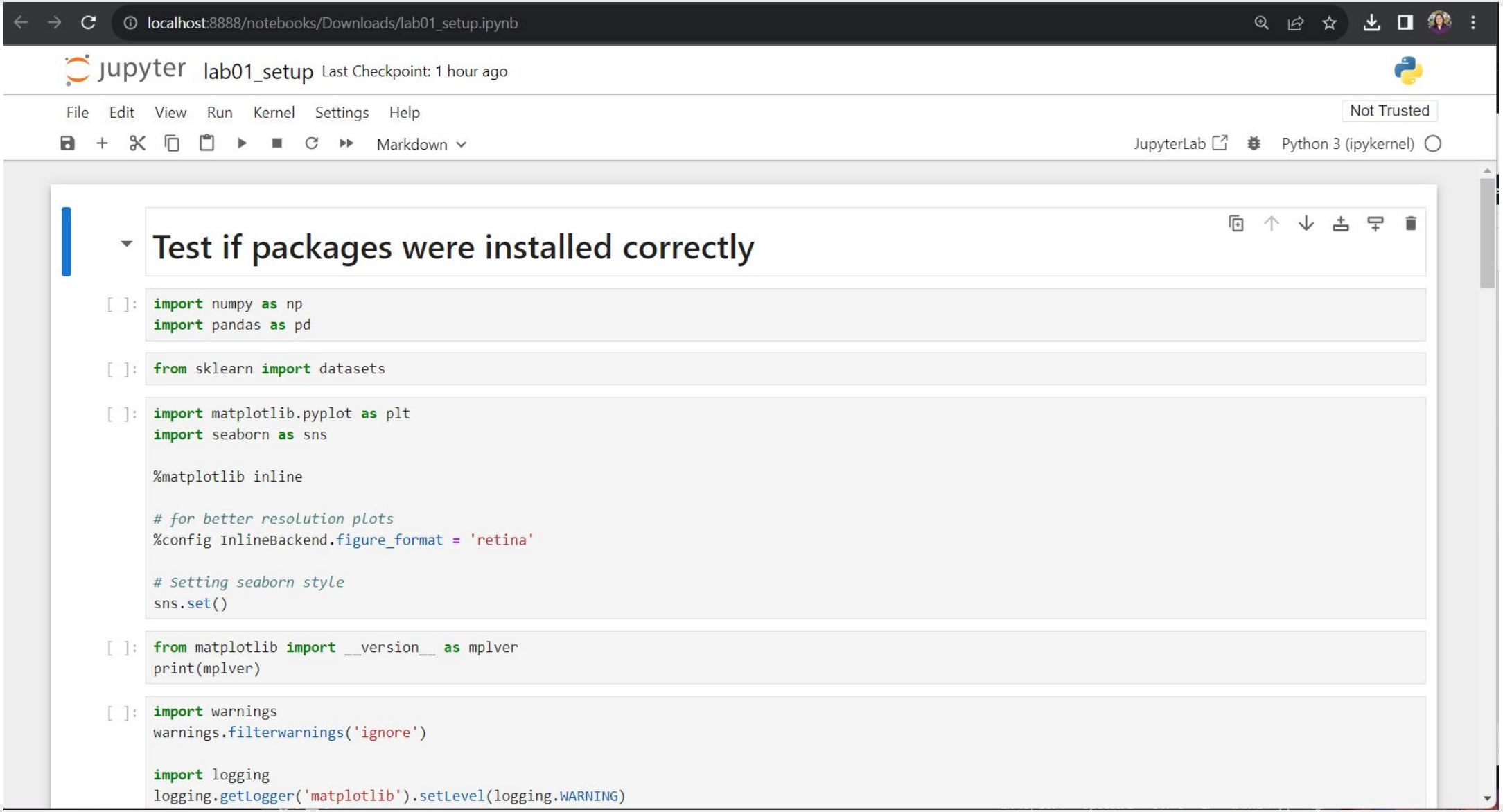
Test Jupyter notebook



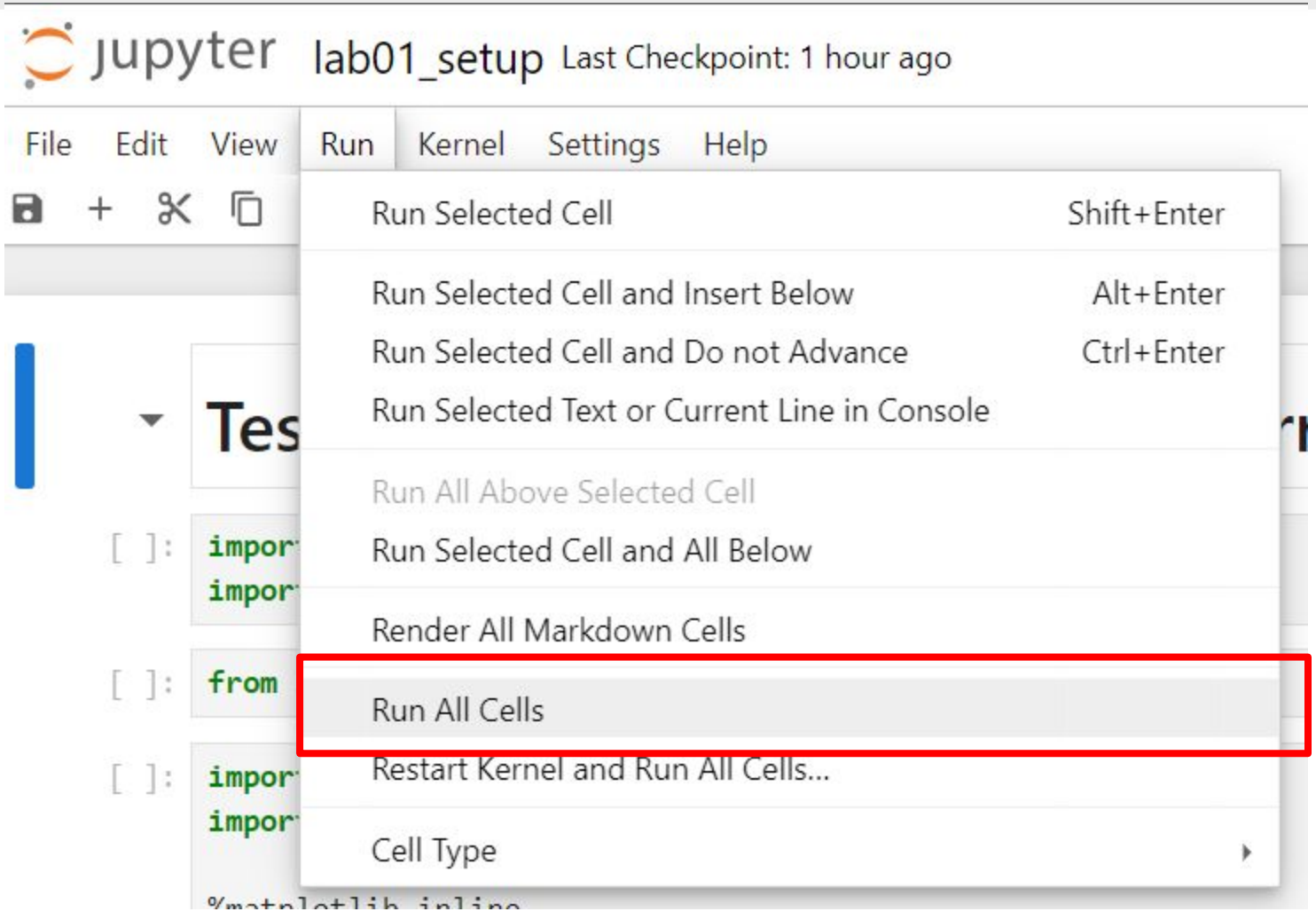
Don’t worry about understanding the code at this point

Right now we just want to make sure that all the packages we need are installed and work correctly

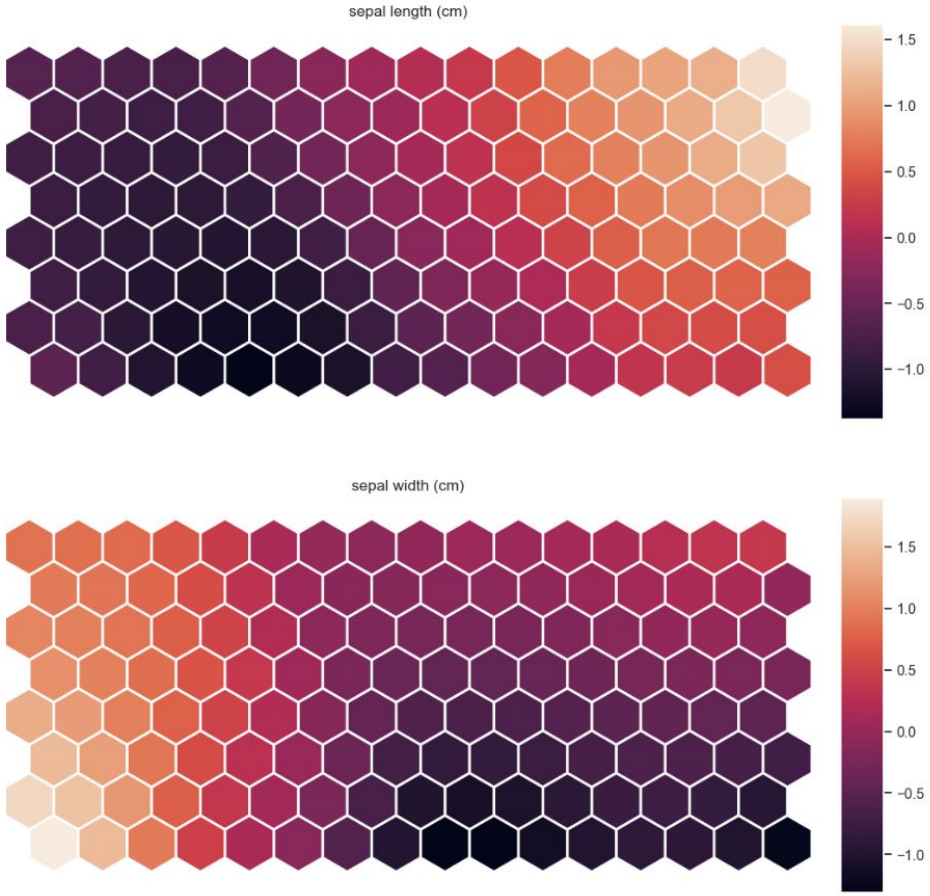
Load the notebook file

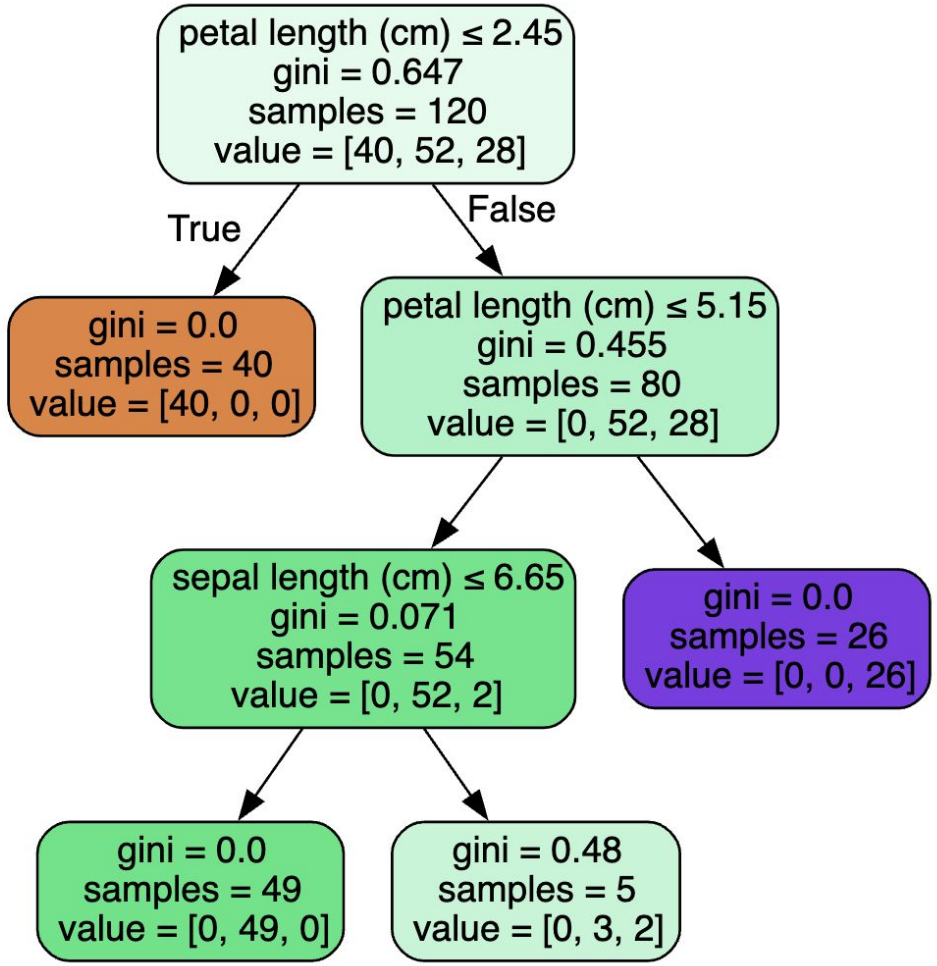


Run all cells



You should be able to see these visualizations





Does anyone still have any errors in the notebook?

Installing using command line

You can also use miniconda
instead of Anaconda Navigator

https://docs.anaconda.com/free/anaconda/getting-started/distro-or-miniconda/

Installing using the command line

Install miniconda

https://docs.conda.io/projects/miniconda/en/latest/#quick-command-line-install

Create environment using the command line

Windows: Open Anaconda Prompt (miniconda3)

Linux/Mac: Open Terminal

```
cd Downloads
conda env create -f environment_dm.yml
conda activate DM2324
```

Test Jupyter notebook + install other packages

Windows: Open Anaconda Prompt (miniconda3)
Linux/Mac: Open Terminal

```
conda activate DM2324
jupyter notebook
```

Follow the instructions in previous slides for testing Jupyter notebook and installing additional packages (**from slide 34**)

Let’s get started!
(for real)

Next:

Jupyter notebook

Distance Matrix

Questions?

Morada: Campus de Campolide, 1070-312 Lisboa, Portugal
Tel: +351 213 828 610 | Fax: +351 213 828 611

Acreditações e Certificações da NOVA IMS

Cofinanciado por