NOVA IMS Information Management School

Data Mining

Practical Session #1

Fall Semester 2024-2025
Master in Data Science and Advanced Analytics

Pedro Ferreira – pnferreira@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

Practical Session Preparation

Download Anaconda Navigator

- https://www.anaconda.com/download
- Documentation: https://docs.anaconda.com/free/anaconda/install/

Download notebook and other materials

- Moodle / Practical Sessions / Lab 01
- https://github.com/fpontejos/Data-Mining-24-25

Optional: Download GitHub Desktop

- https://desktop.github.com/
- Sign up for GitHub account: https://github.com/signup

About Us

Pedro Ferreira

Academic Background:

- BSc in Physics (2022)
- MSc in Data Science & Advanced Analytics (2022 – Oct 2024 – NOVA IMS)

Professional Experience:

- Data Science & Analysis in the Banking Sector

Research Experience:

- Machine Learning for Astrophysics
- Supervised Machine Learning for Football Performance Evaluation

About Us

Farina Pontejos

Academic Background:

- BSc in Mining Engineering (2011)
- MSc in Data Science & Advanced Analytics (2021-2023 – NOVA IMS)
- PhD in Information Management (since 2023 [Ongoing] – NOVA IMS)

Professional Experience:

- Mining Engineer; Web Developer
- IT Consulting (Systems Analysis; ERP; Data Science)

Research Experience:

- Natural Language Processing; Large Language Models; Software Engineering Applications

Resources

- Bibliography
- Data Mining Github repo:
  - https://github.com/fpontejos/Data-Mining-24-25
- 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

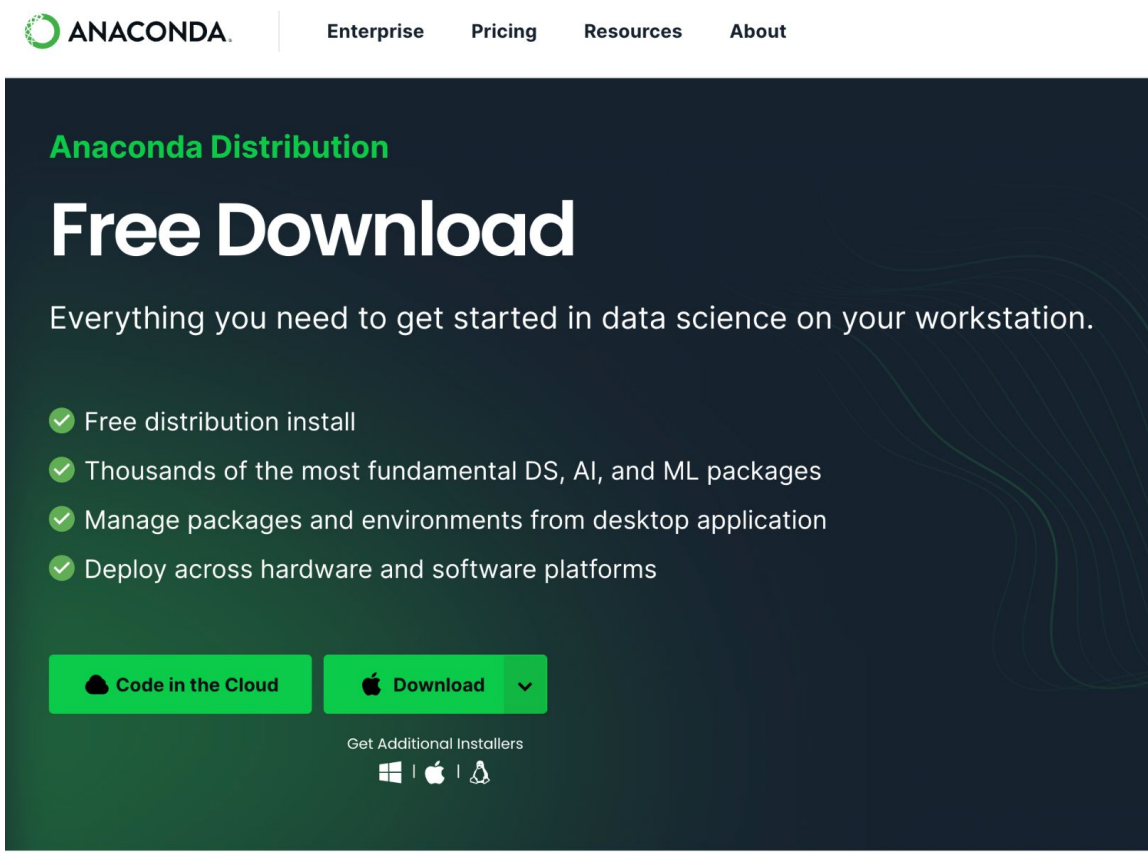
www.anaconda.org

- 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

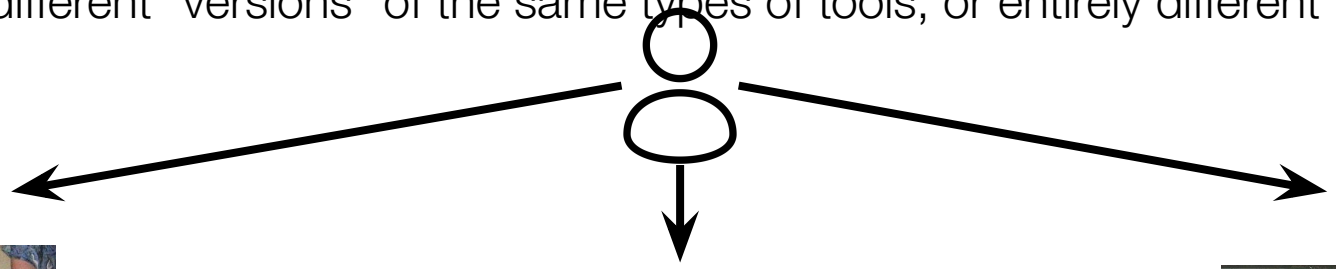
https://docs.conda.io/projects/conda/en/latest/user-guide/concepts/environments.html

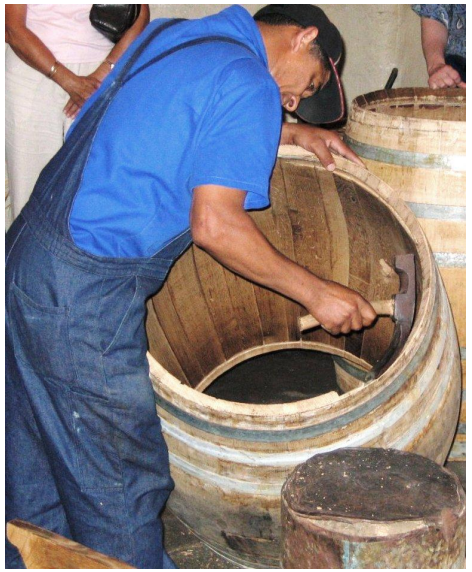
https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html

- 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-24-25

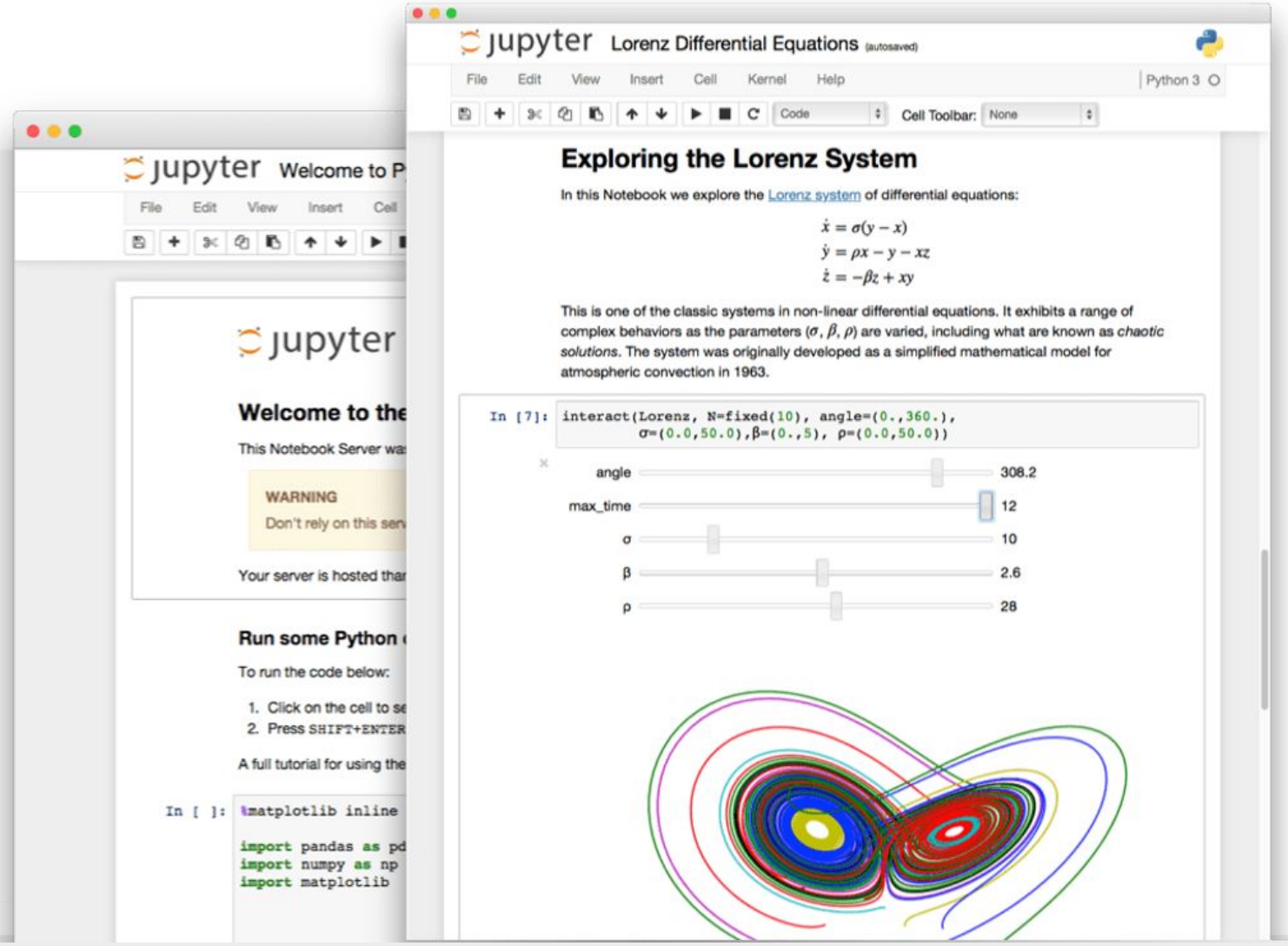
Main ways to access Python

- 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):
  - Software that helps you build code

Jupyter notebooks

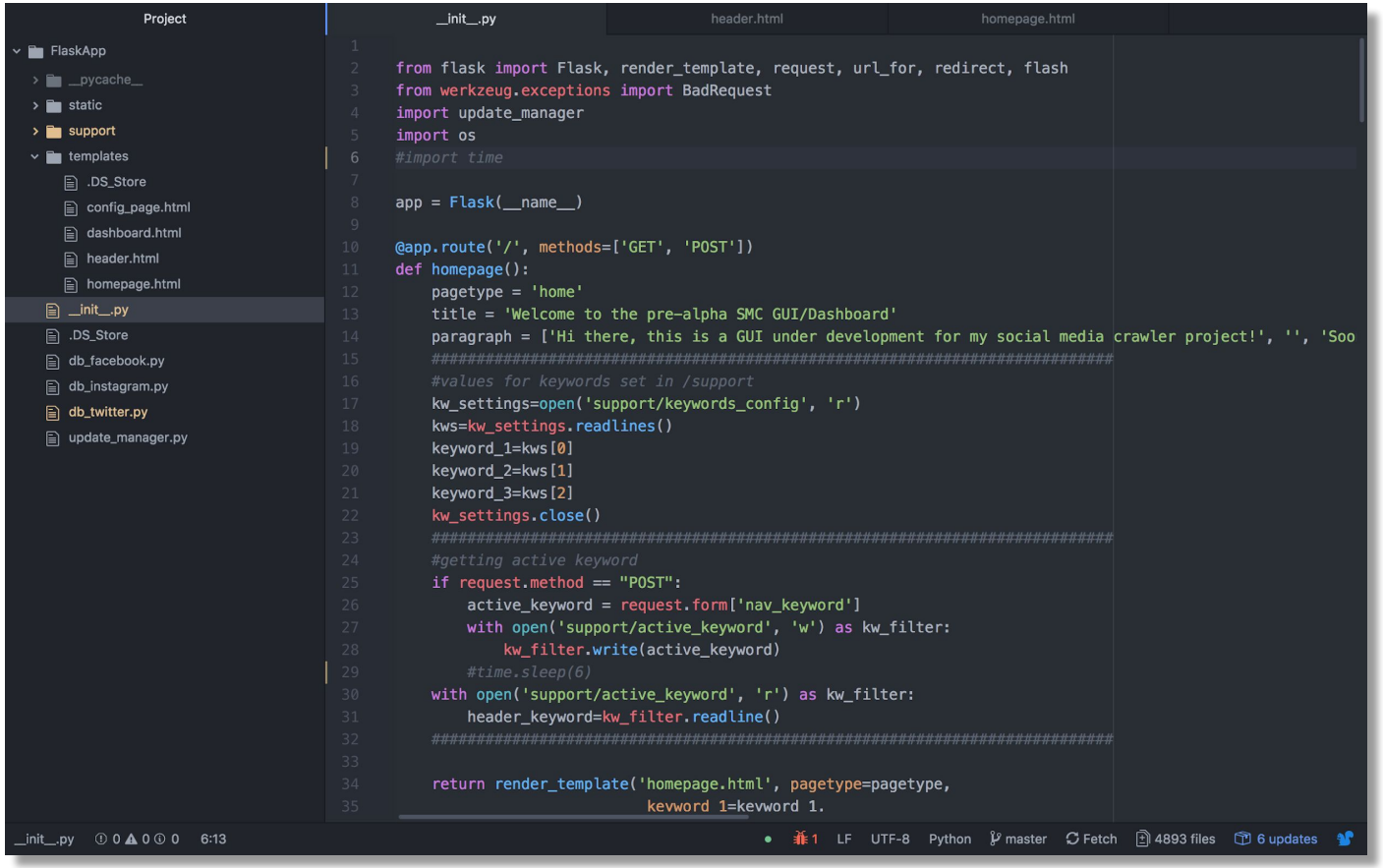
https://jupyter.org/

We will be using Jupyter notebooks for our practical sessions.



Text Editors

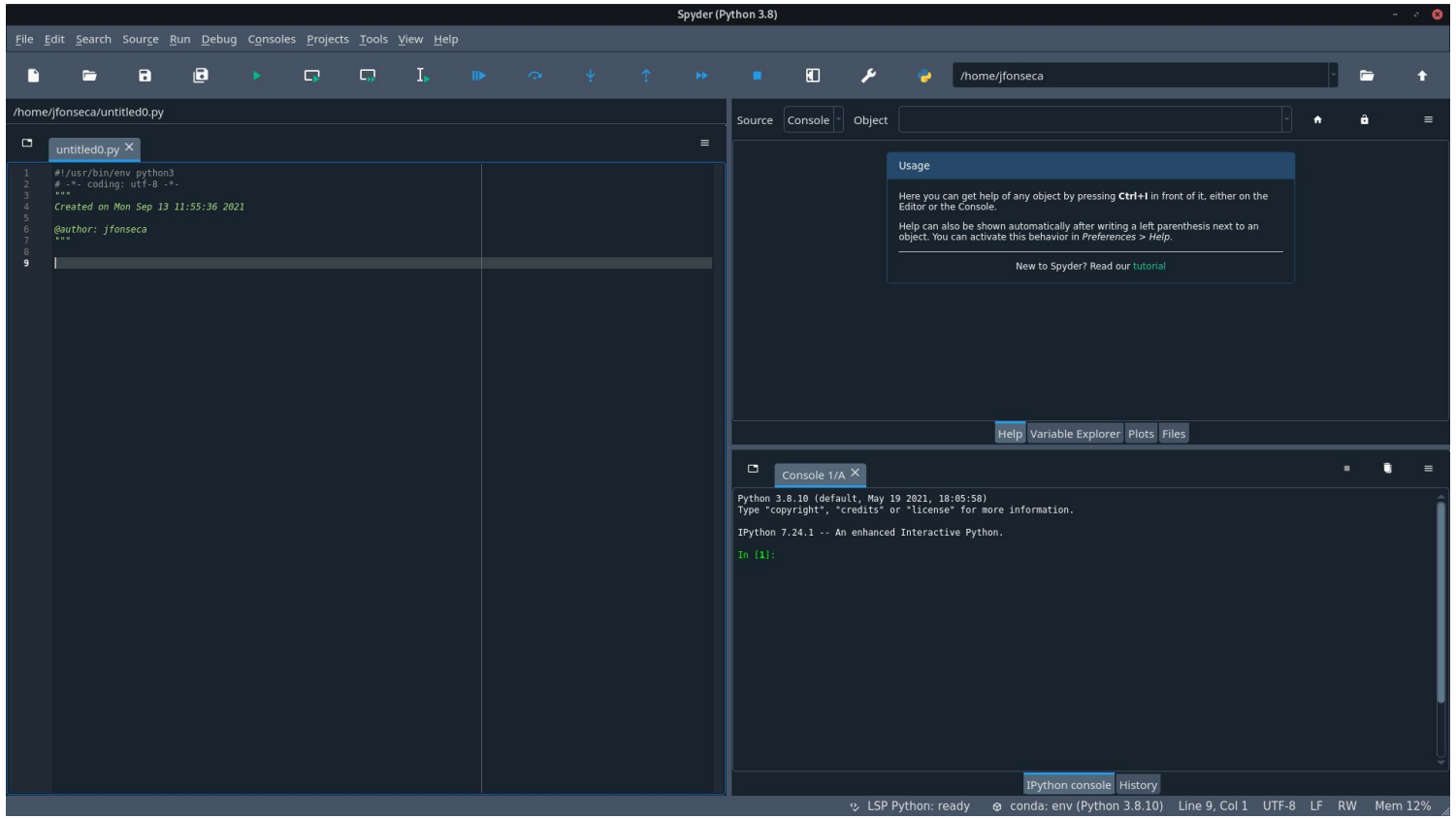
- Another method to write python scripts is using text editors
- Some popular text editors:
  - Vim (Linux terminal text editor)
  - Atom (popular open source editor)
  - Sublime Text (popular proprietary text editor)
  - Notepad ++ (Windows only)
- Usually highly customizable



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

NOVA IMS Information Management School

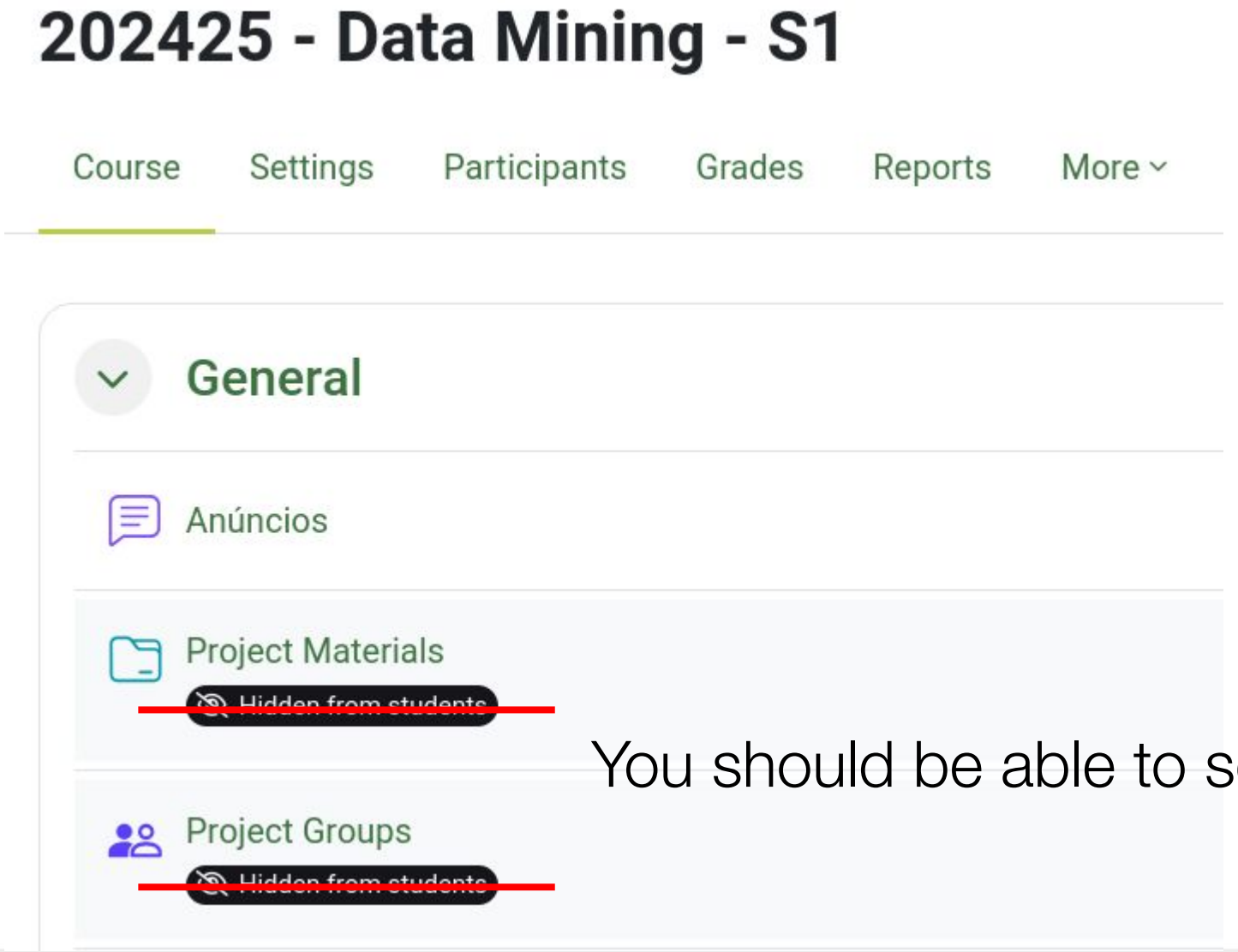
Data Mining Project

Data Mining Project

- Form groups on Moodle **(up to 4 students)**
- Project guidelines will be released on Moodle
- Anonymized data from a real-world scenario

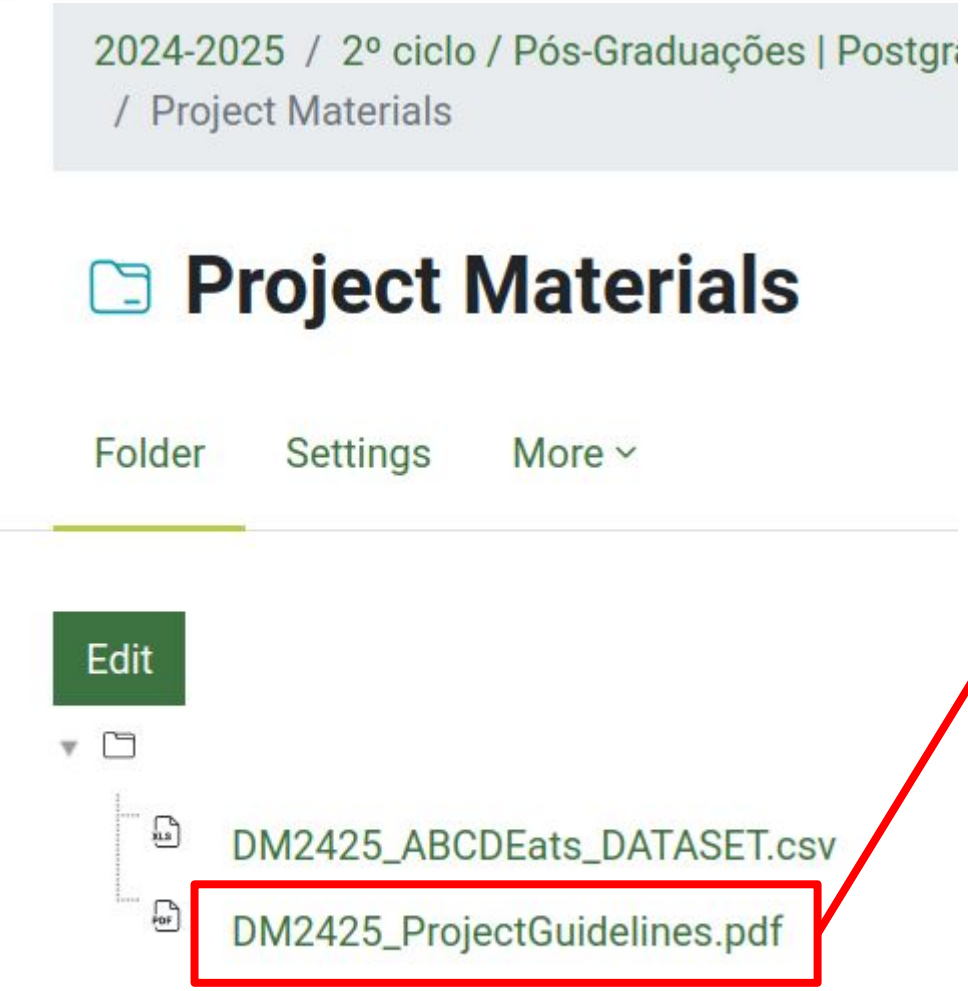
- Your goal is to develop a **Customer Segmentation** in such a way that it will be possible for the Marketing Department to better understand all the different Customers' Profiles.

Everything is on Moodle



You should be able to see this now :)

Please read the guidelines



ABCDEats Inc.

Data Mining Project Guidelines

version: v6.2024.09.09

Fall Semester 2024-2025

Master in Data Science and Advanced Analytics

NOVA Information Management School

Project Part 1: EDA

Project Evaluation

The project will be evaluated based on three components. All components are mandatory.

Part 1: Exploratory Data Analysis (20%)

Expected Outcomes

- Conduct an in-depth exploration of the dataset. Summarise key statistics for the data, and discuss their possible implications.
- Identify any trends, patterns, or anomalies within the dataset. Explore relationships between features.
- Create new features that may help enhance your analysis.
- Use visualisations to effectively communicate your findings.

Submission deadline

- 04 November 2024 21h00
- 10% penalty for each day of delay

Deliverables

- Written report, maximum of 5 pages.
- Python code, in a Jupyter notebook with cells already executed.
- See Project Deliverables section for details

Project Part 1: EDA

Academic paper example:

M. Maphosa, W. Doorsamy and B. S. Paul, "Student Performance Patterns in Engineering at the University of Johannesburg: An Exploratory Data Analysis," in *IEEE Access*, vol. 11, pp. 48977-48987, 2023, doi: 10.1109/ACCESS.2023.3277225.

https://ieeexplore.ieee.org/abstract/document/10128127

Project Part 2: Final Report

Part 2: Final Report (70%)

Expected Outcomes

- Preprocess the data. Invest time into evaluating your preprocessing pipeline, explaining the choices you made and the advantages and disadvantages of different decisions.
- Justify the clustering approach. Determine and provide a rationale for the clustering solution, including the number of clusters, that you decide to use.
- Explain the clusters in your final segmentation. Analyse and describe the characteristics of each group, taking into account the perspectives you used. Create profiles that highlight the distinguishing features of each cluster.
- Suggest business applications. Based on your insights, define general marketing approaches for each cluster.

Submission deadline

- 03 January 2025 21h00
- 10% penalty for each day of delay

Deliverables

- Written report, maximum of 10 pages.
- Python code, in a Jupyter notebook with cells already executed.
- See Project Deliverables section for details

Project Part 3: Discussion

Part 3: Project Discussion (10%)

After submitting the final reports, each group will be scheduled for a 15-20 minute discussion with one of the instructors. During this discussion, students will answer questions about their analysis and explain their methodologies.

Each student will receive an individual grade based on their contribution to the discussion. This aims to assess each student's level of engagement with, and comprehension of the delivered work.

Target schedule: 21-25 January 2025. **This is not yet final** and will be confirmed later on Moodle.

Project Bonus **(Optional)**

Optional: Opportunity for Bonus Points (max 20%)

Note: *This does not replace the requirement to deliver the other parts of the project.*

As an optional part of the project, you may develop an interactive application that allows ABCDE to explore and interact with the EDA and customer segmentation analysis performed previously. The application should be user-friendly, visually intuitive, and allow the user to gain insights from the customer segmentation results.

This is intentionally open-ended; some suggestions are provided below, but the kinds of visualisations or features to include is limited only by your imagination and skill.

The only restriction is that it must be developed using Python.

Possible application features:

- **Cluster Exploration**: The application must provide an interactive interface where users can explore the different customer segments. Each cluster should be clearly defined, with key demographic, behavioral, and preference-based characteristics displayed.
- **Visualisation Tools**: Implement visual tools such as charts, graphs, or scatter plots that help users visualize the clustering results. Consider using heatmaps, pie charts, or bar graphs to represent segment data.
- **Filter Functionality**: Allow users to filter clusters based on various attributes (e.g., age, location, purchase history) and explore how these filters affect the composition of the clusters.
- **Cluster Comparison**: Enable users to compare different clusters side-by-side in terms of their key characteristics, providing detailed insights into their similarities and differences.
- **Personalized Insights**: Include an option where users can input specific customer attributes (e.g., age, region, payment method) and get insights into which cluster the customer is most likely to belong to.

Submission deadline

- 07 January 2025 21h00
- **Please inform me by email if you intend to deliver this optional part.**

Let’s get started!

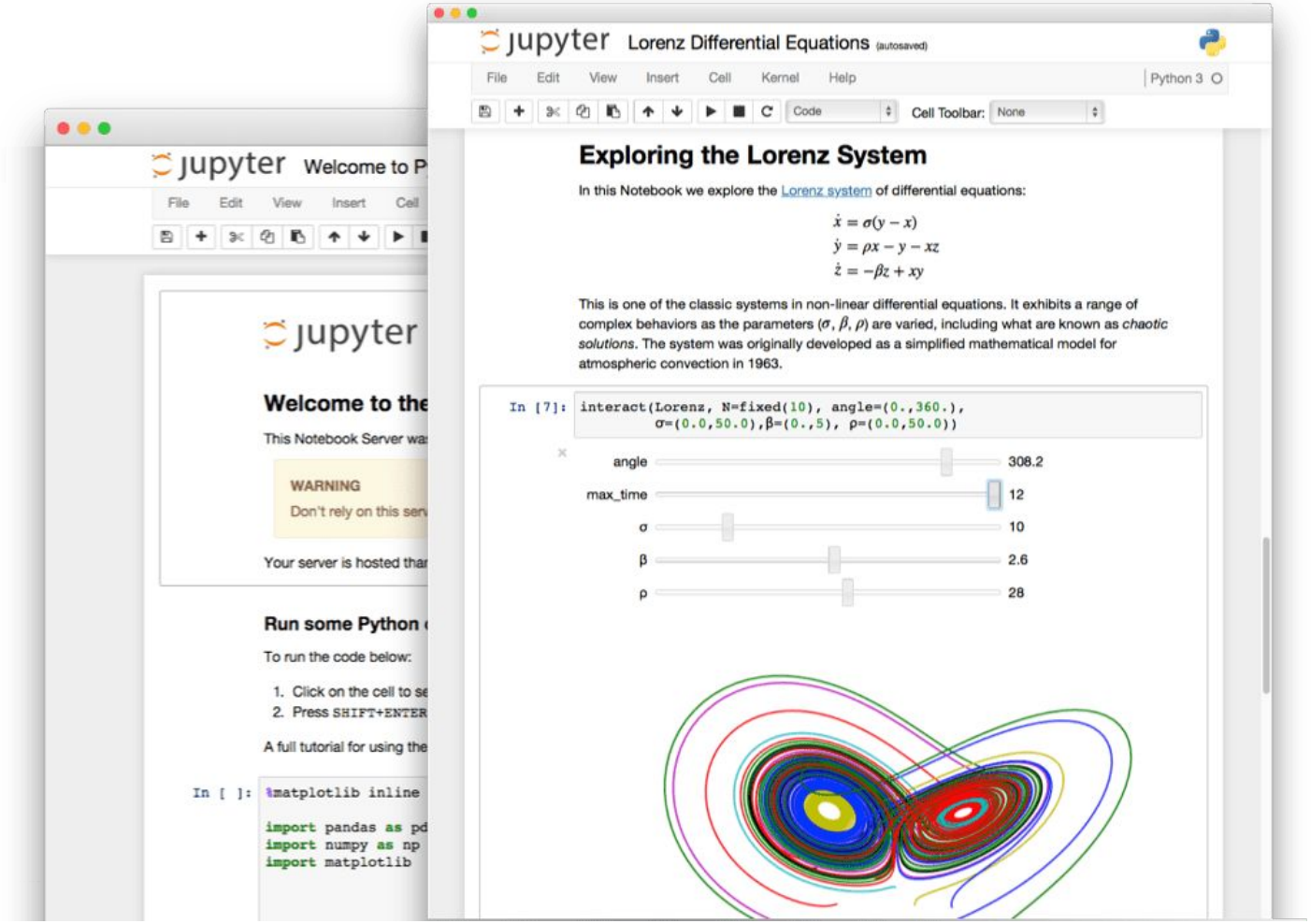
Next: Setting up our tools

The Jupyter Notebook

http://jupyter.org/

Let's try it out!

- Install and Open Anaconda Navigator
- Start Jupyter Notebook



Setting up our tools

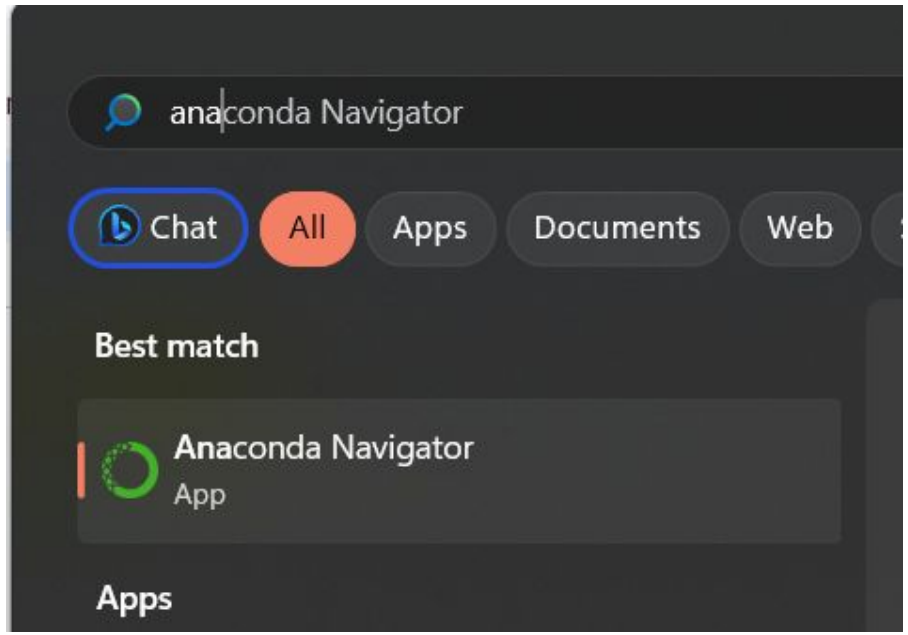
Two options:

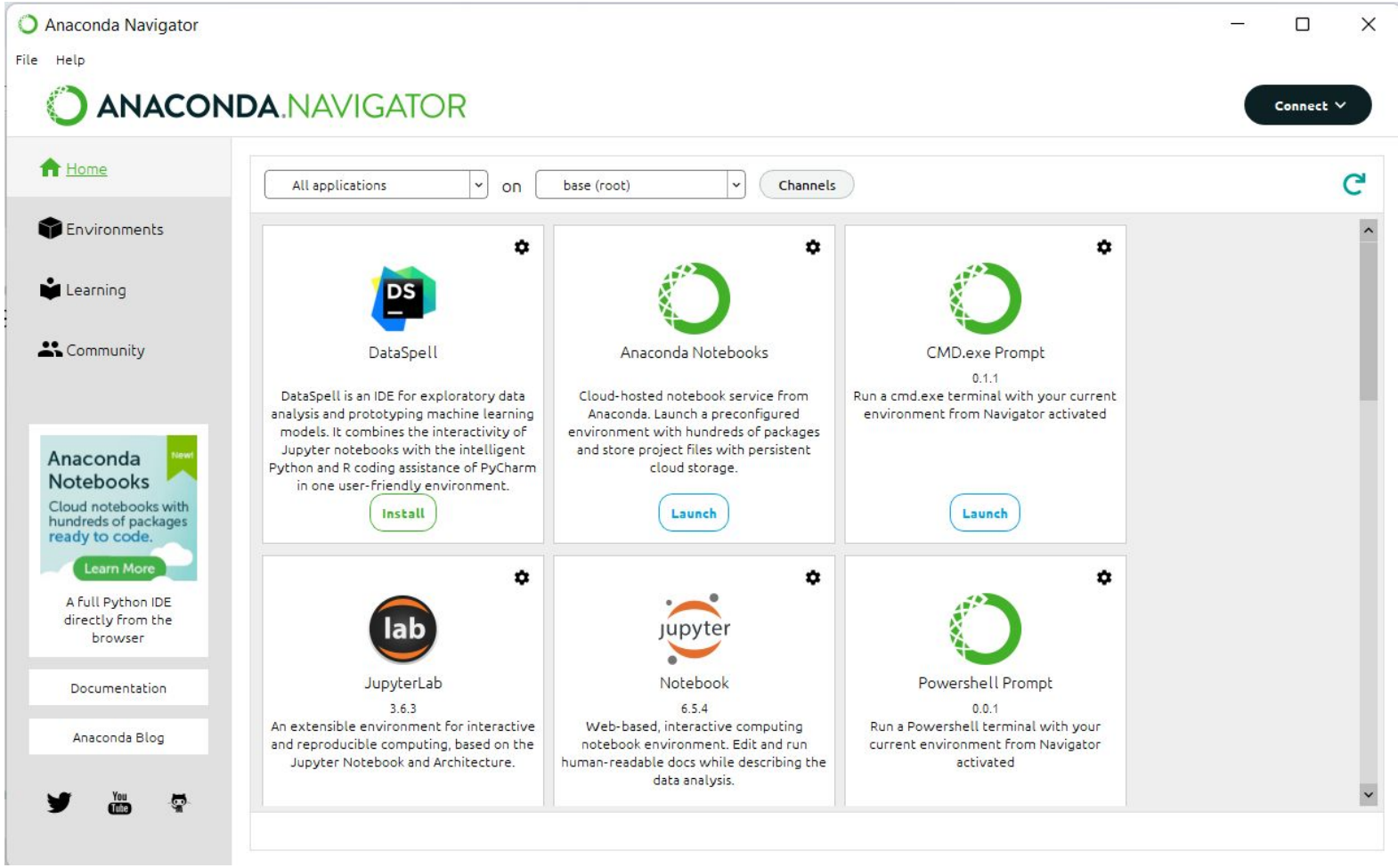
1. **With GUI (Anaconda Navigator)**

2. Command line (miniconda) (skip to slide 64)

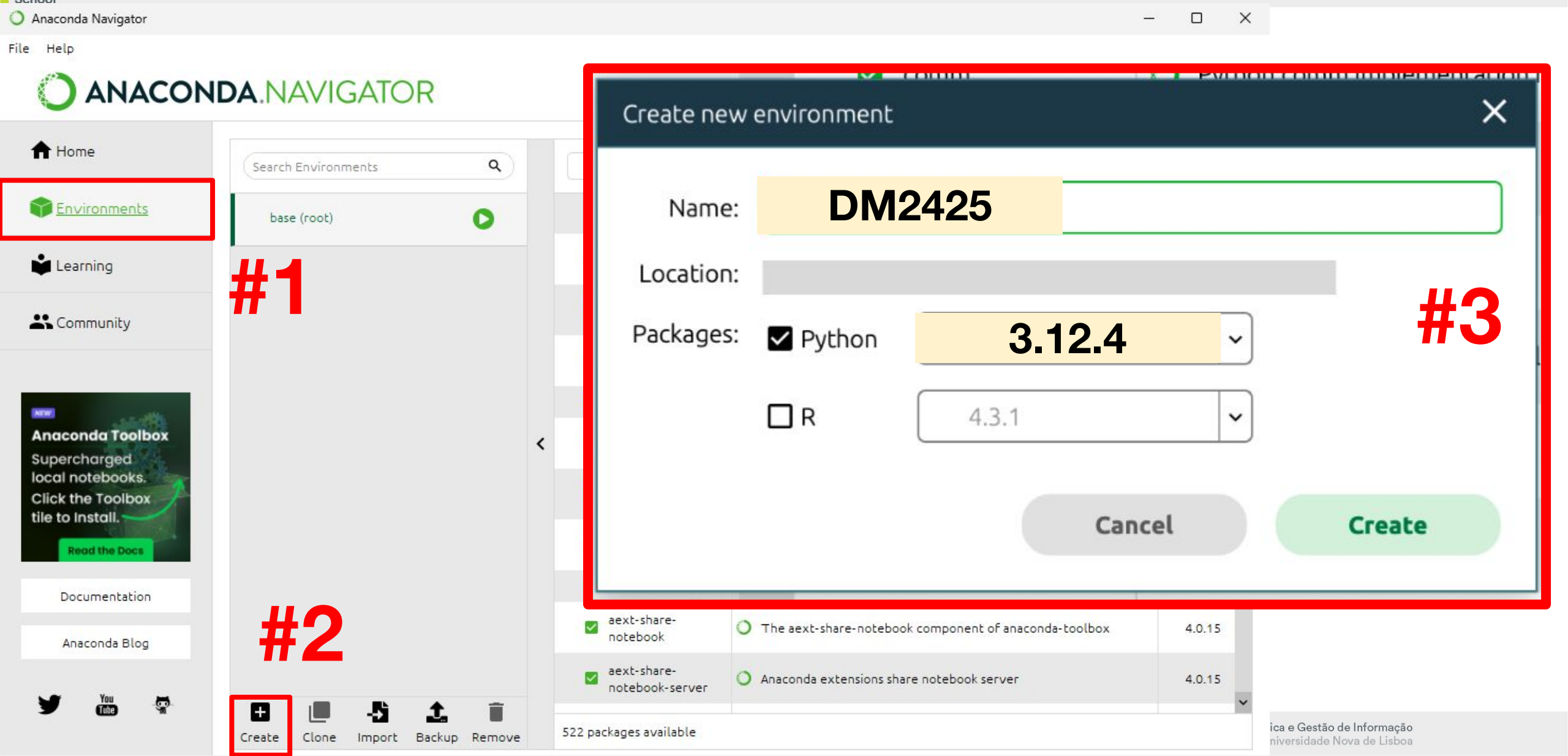
Setting up our tools

Load Anaconda Navigator

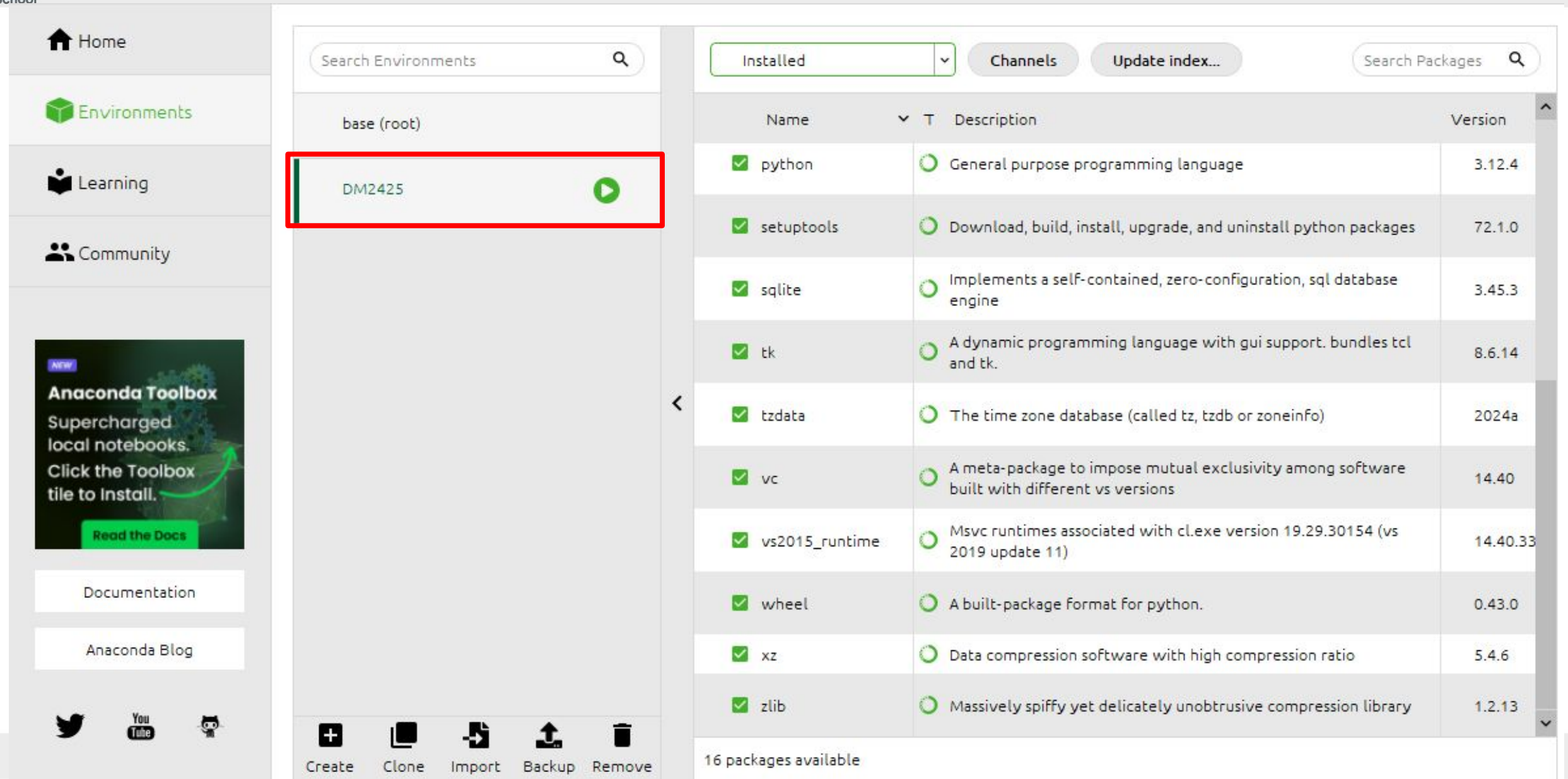




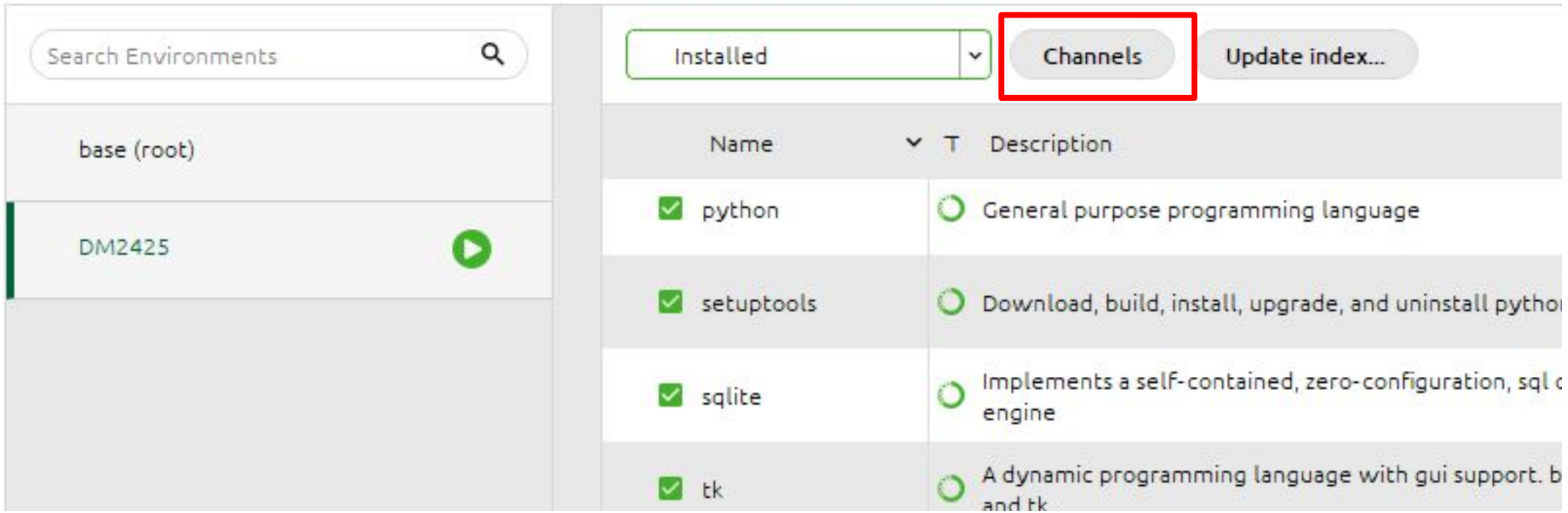
Create a new environment: DM2425



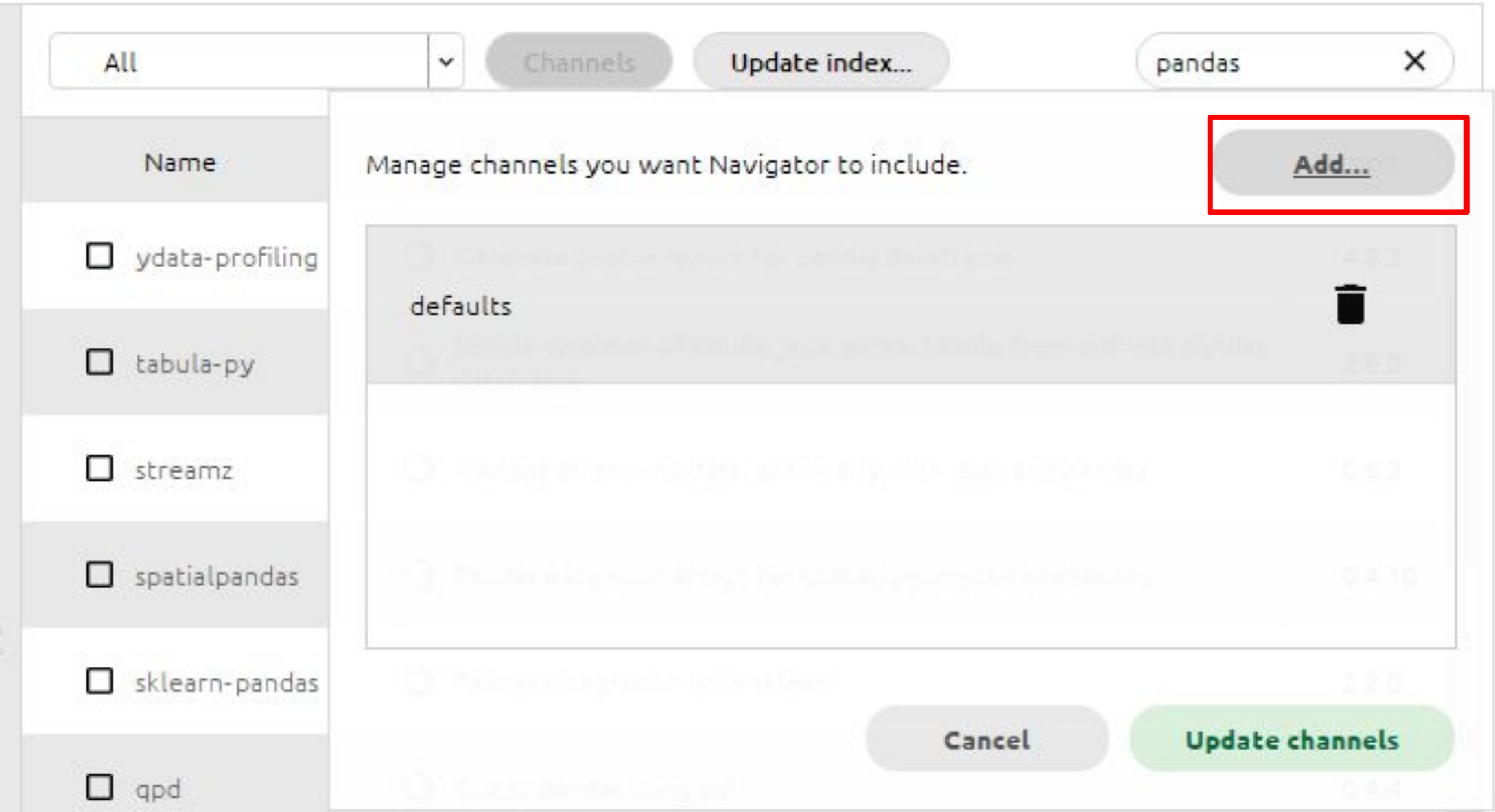
Create a new environment: DM2425



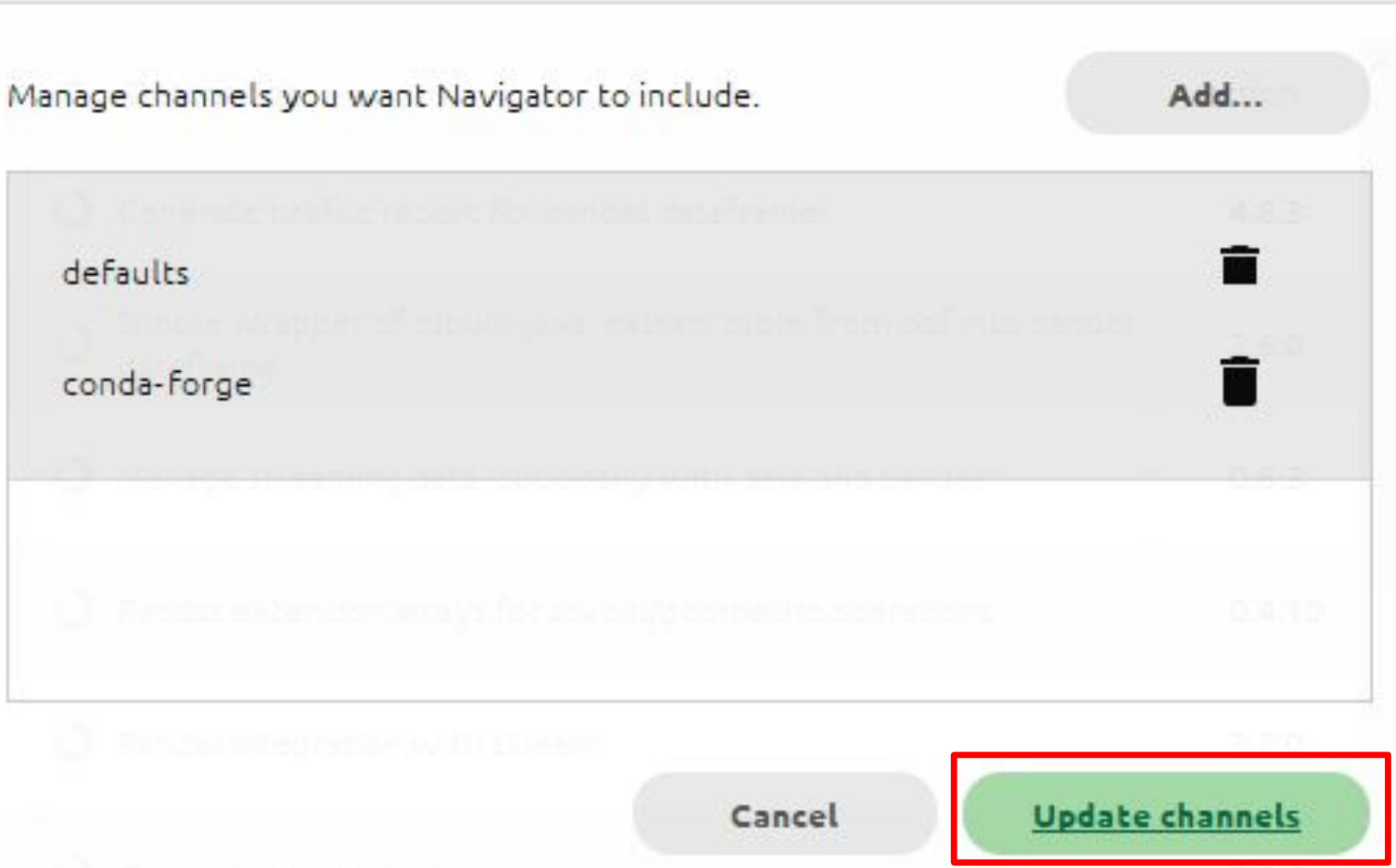
Add conda channel: conda-forge



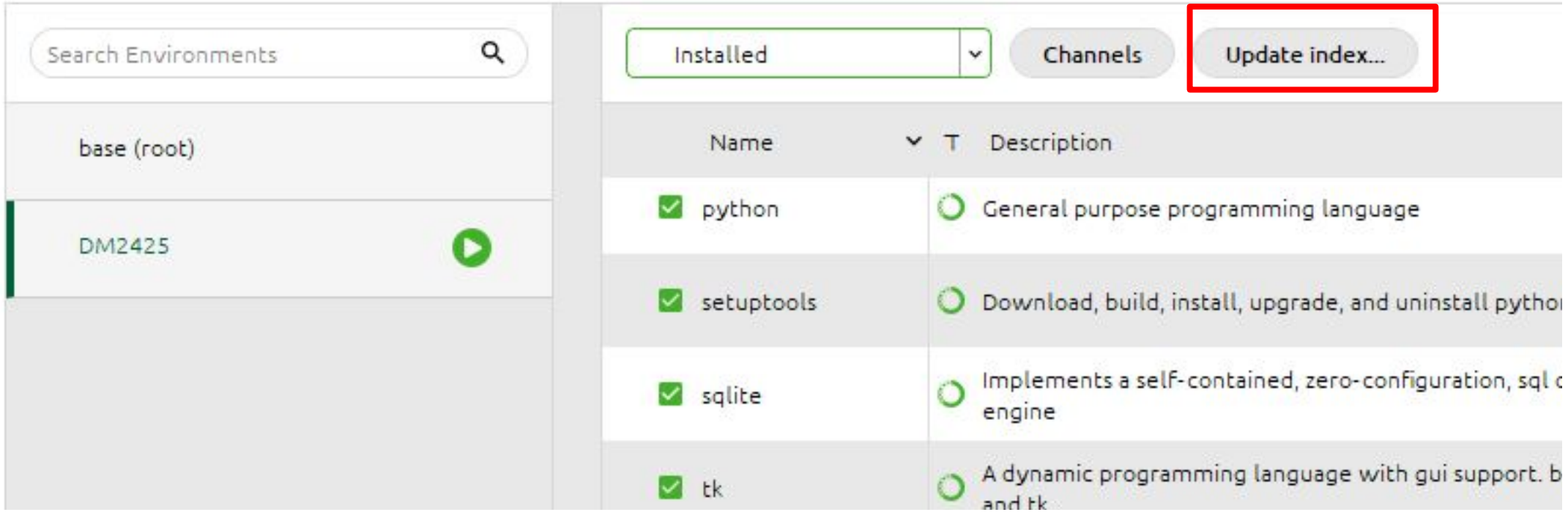
Add conda channel: conda-forge



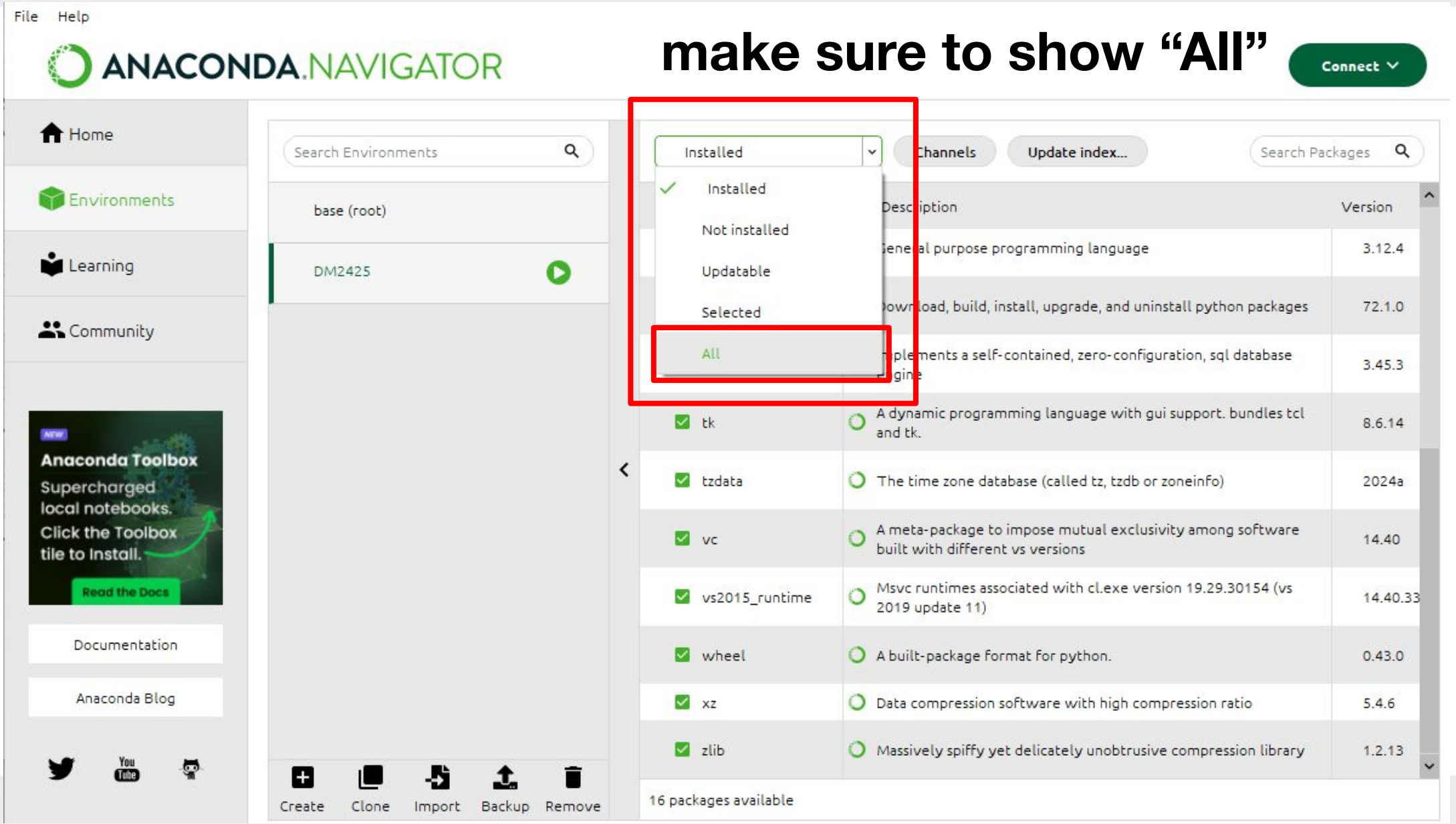
Add conda channel: conda-forge



Add conda channel: conda-forge

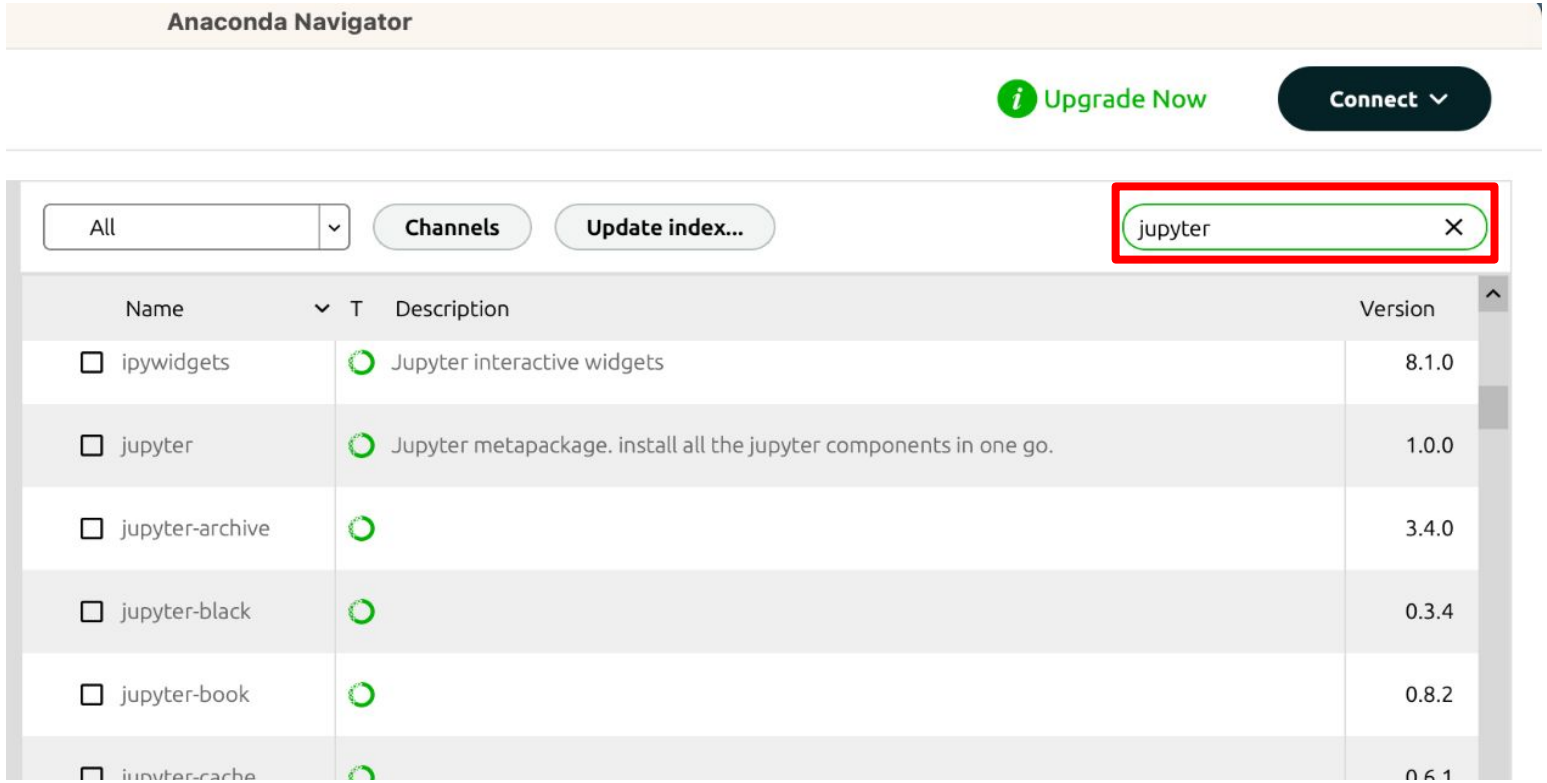


Install libraries



Install libraries

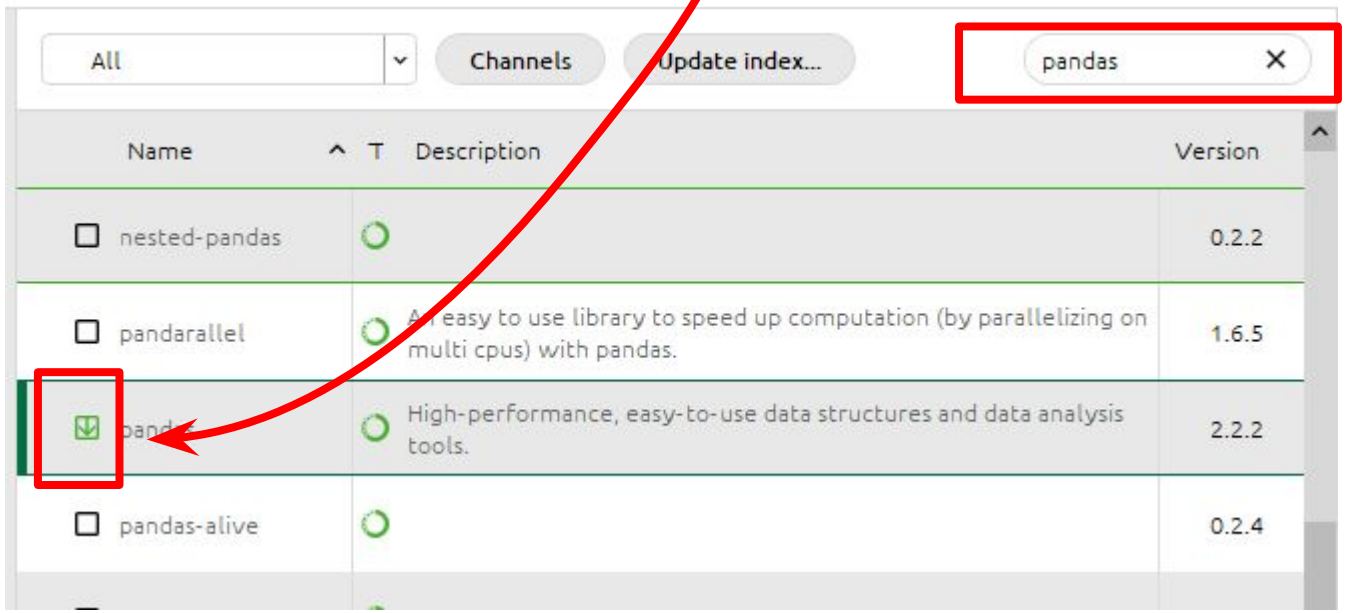
search for the libraries we need



Install libraries

search for the libraries we need

- search package name, then select the check box on the left of the name



- after selecting, you can search the next library name, then select that, and so on

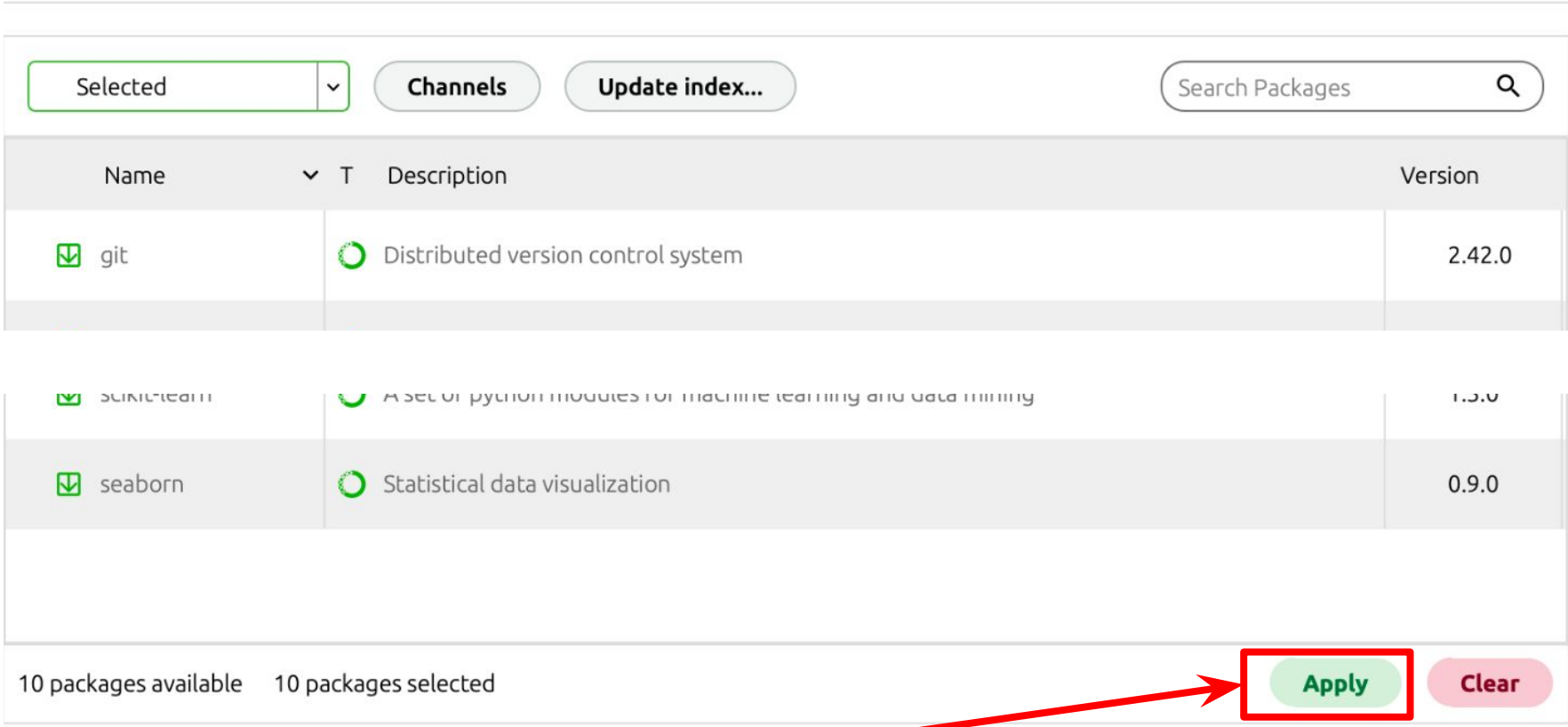
1. `jupyter`
2. `scikit-learn`
3. `scikit-image`
4. `numpy`
5. `pandas`
6. `ipywidgets`
7. `matplotlib`
8. `seaborn`
9. `minisom`
10. `ydata-profiling`

Install libraries

after searching and selecting everything on the list,

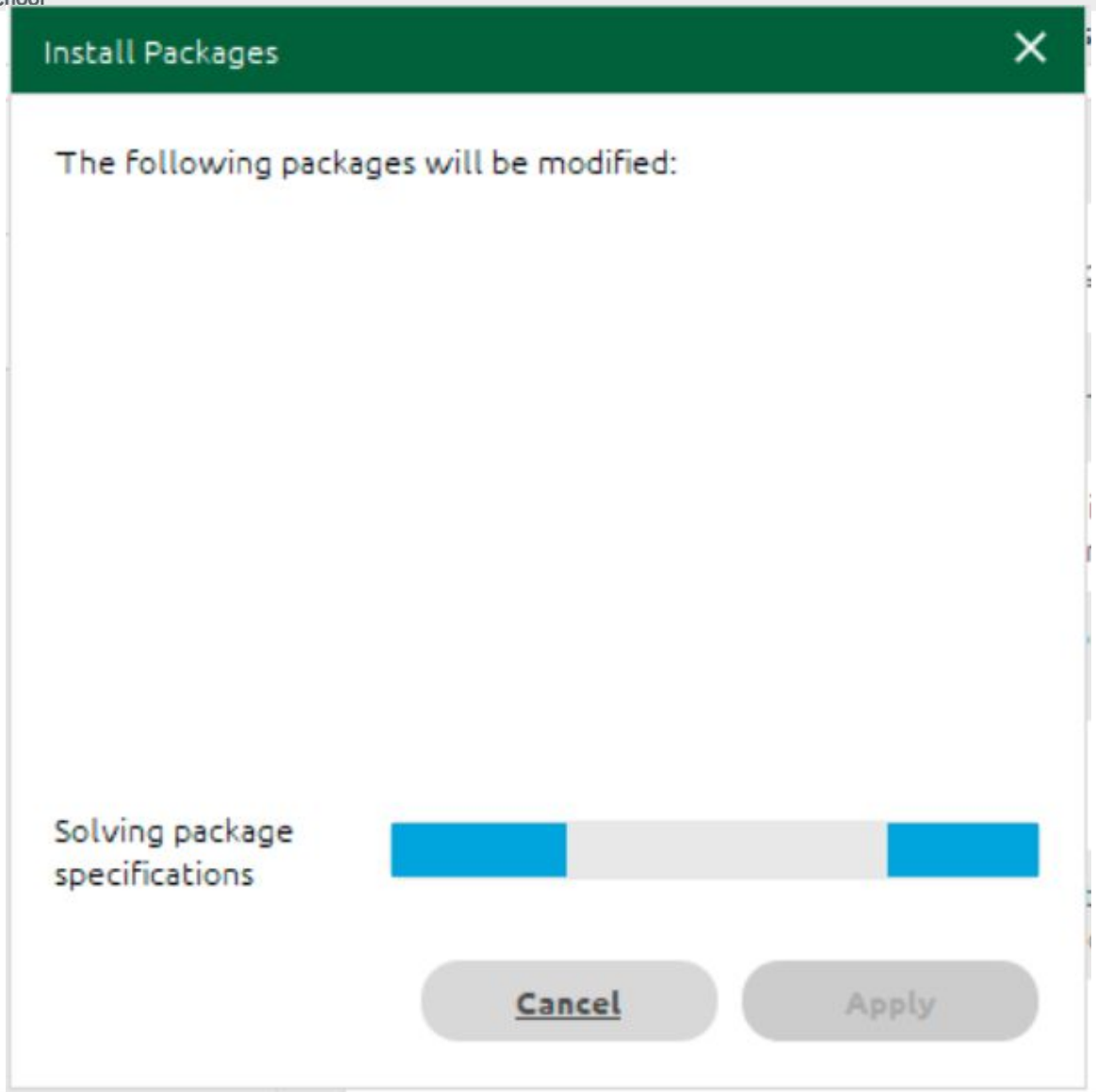
you can clear the search bar by clicking the ‘x’

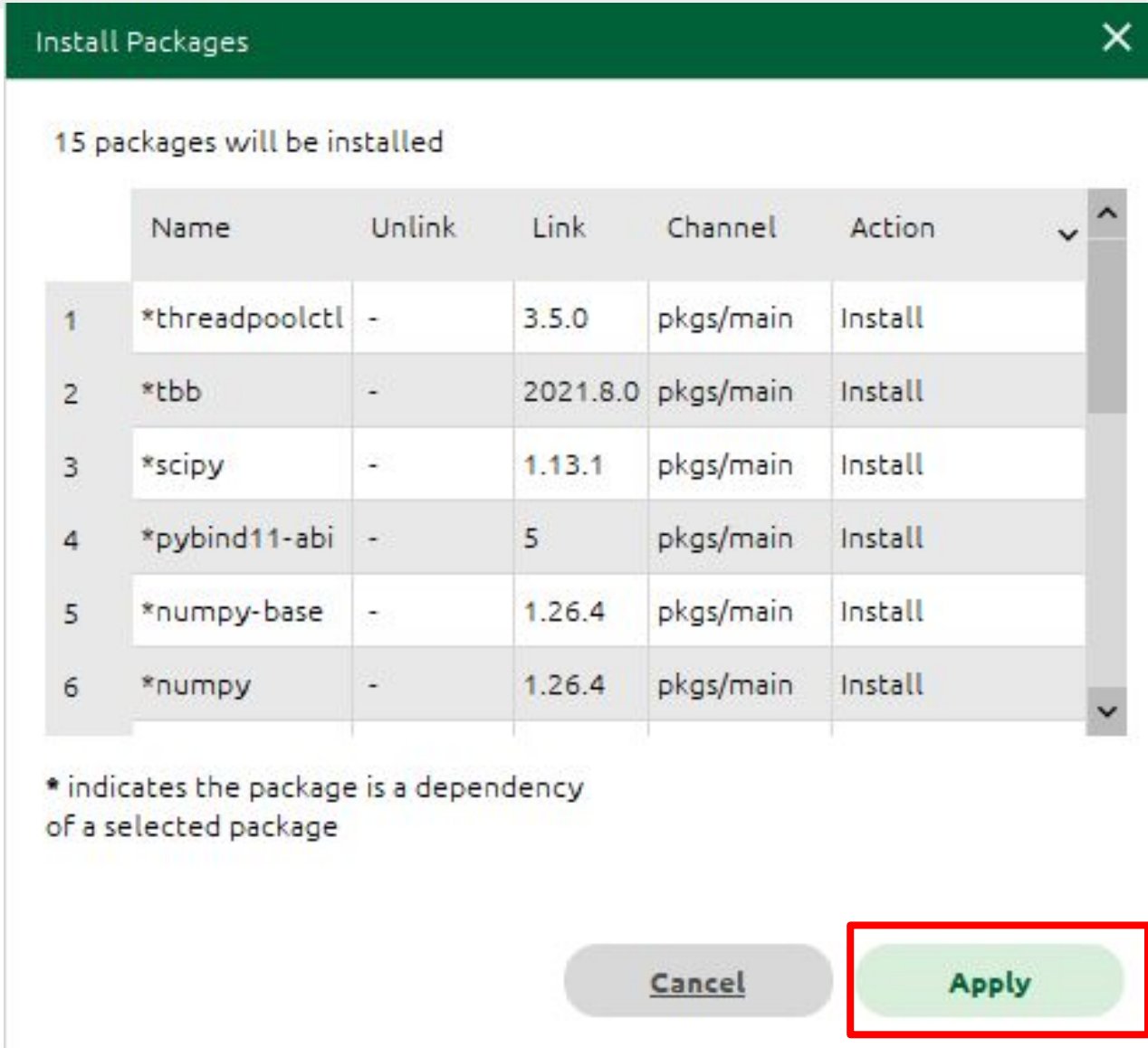
then filter the “Selected” option to see if you got everything



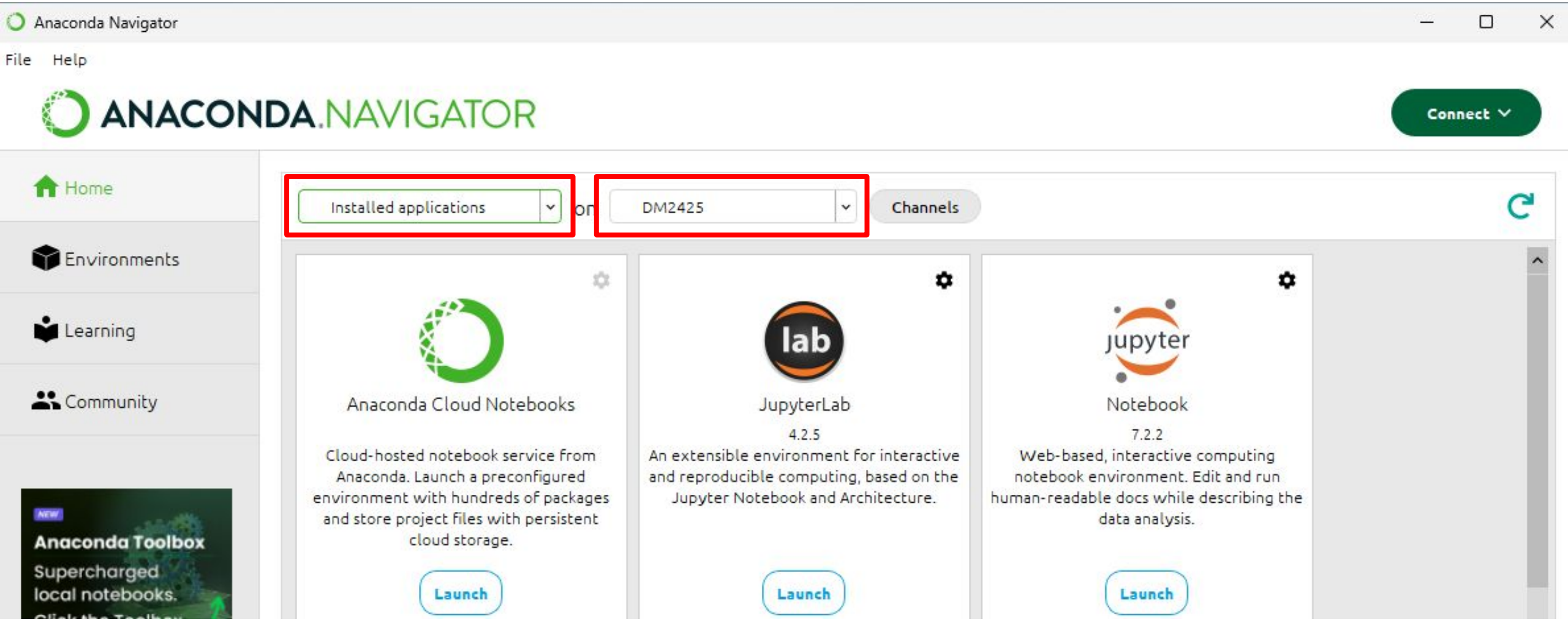
then click Apply

Install libraries





Go back to home



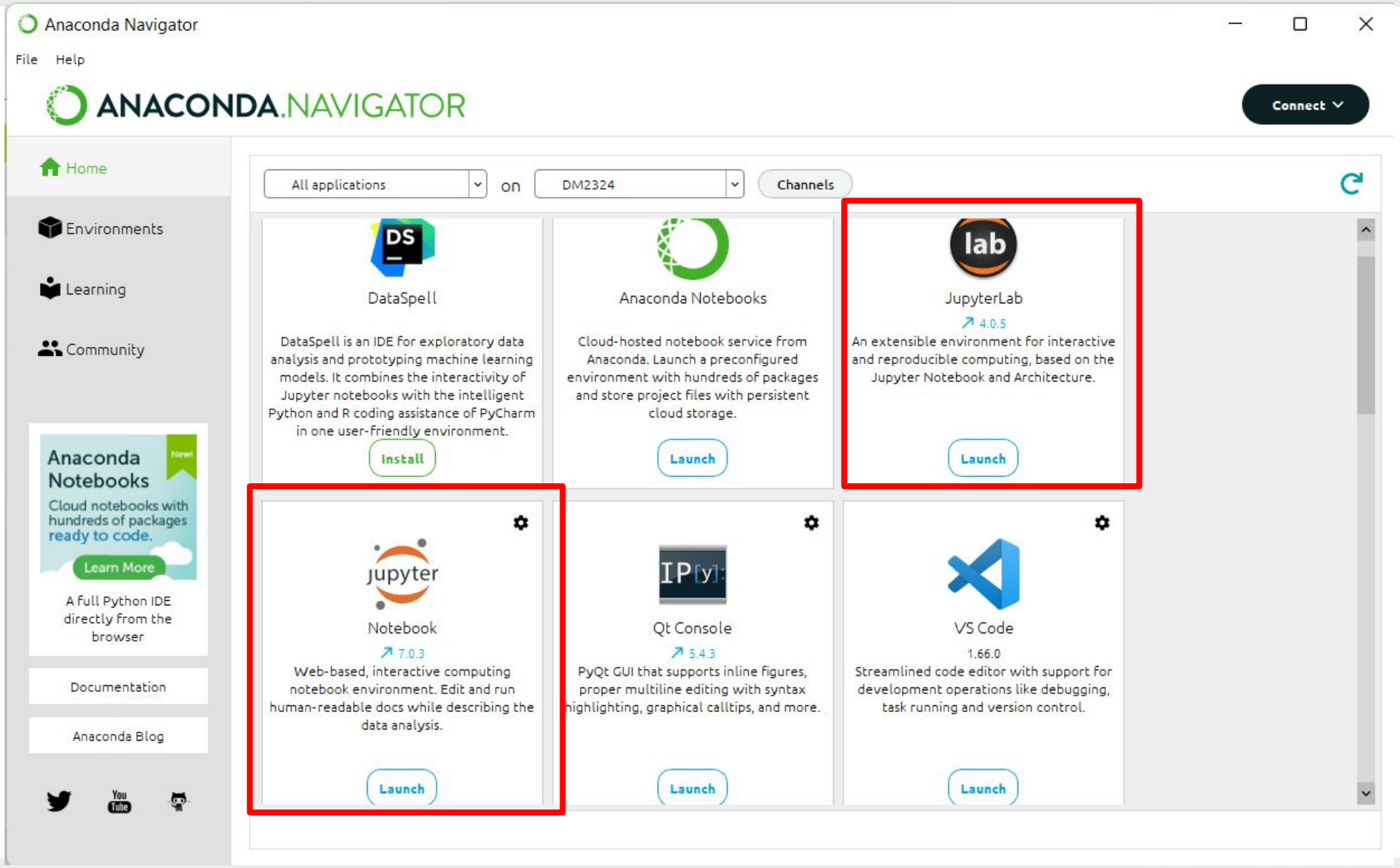
NOVA IMS Information Management School

Test Jupyter Notebook

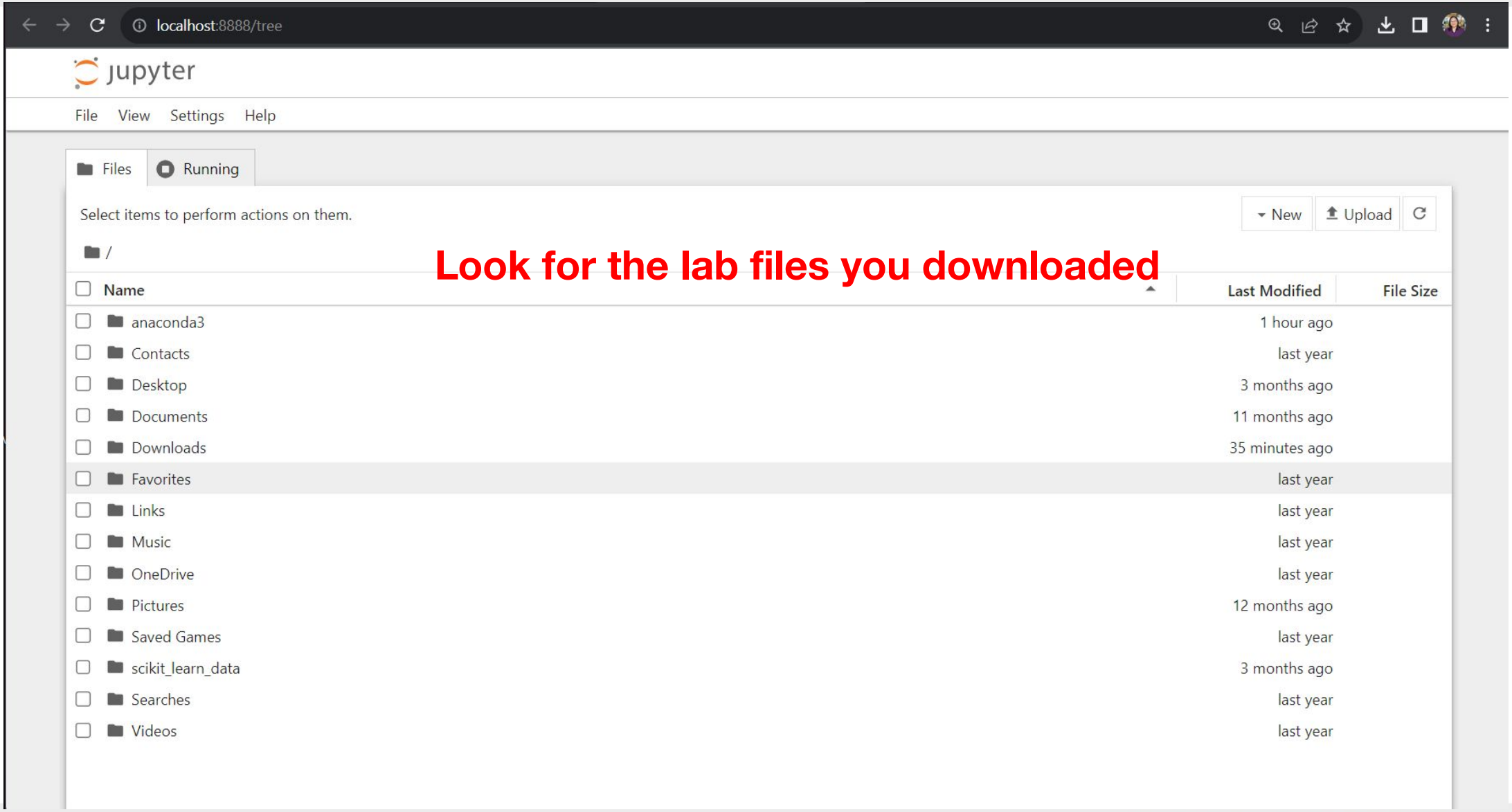
Test Jupyter notebook

Close all Terminal windows / Anaconda Navigator / Anaconda Prompt

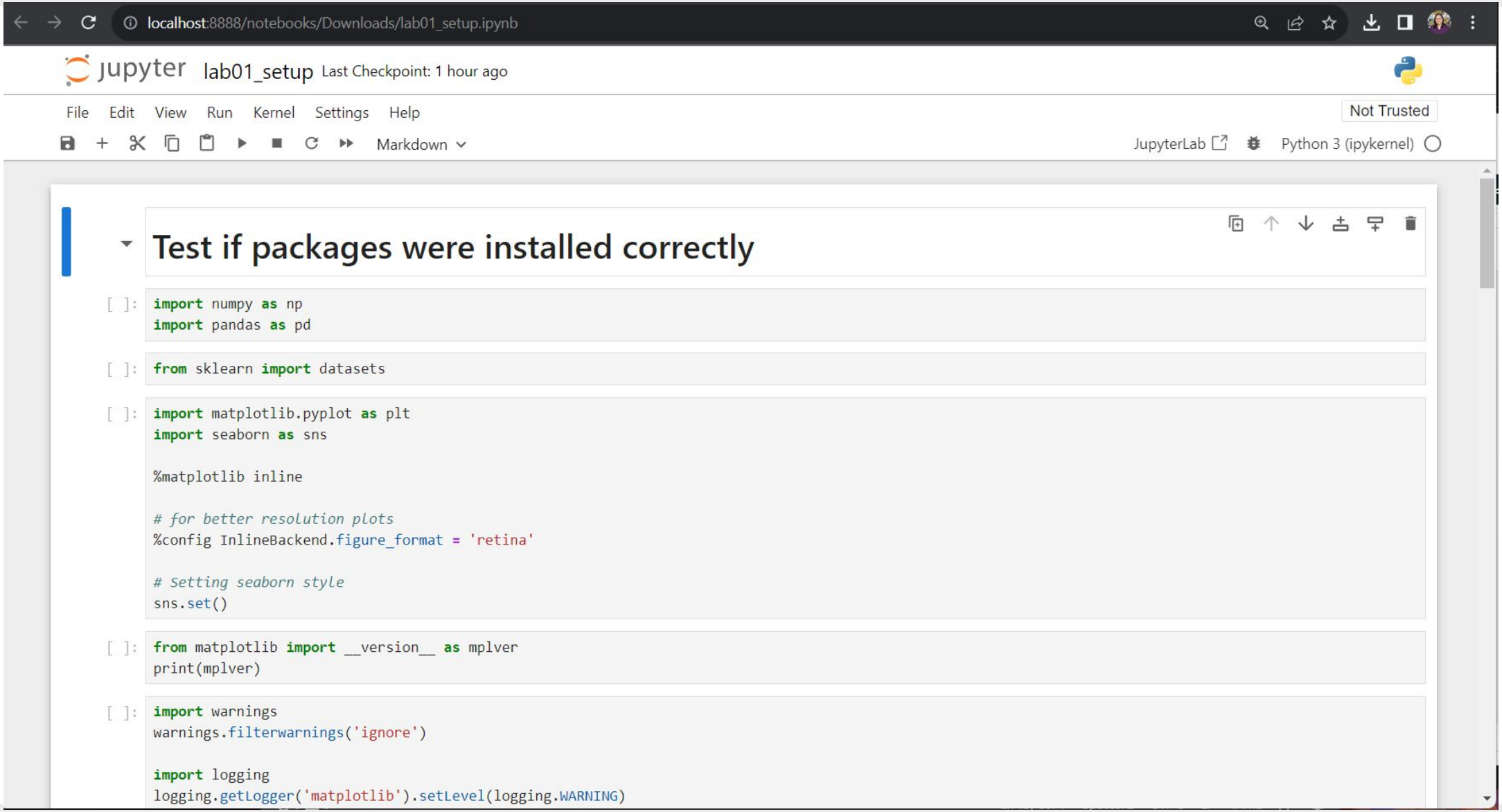
Test loading Jupyter notebook



Test loading Jupyter notebook



Load the lab01_setup.ipynb notebook file



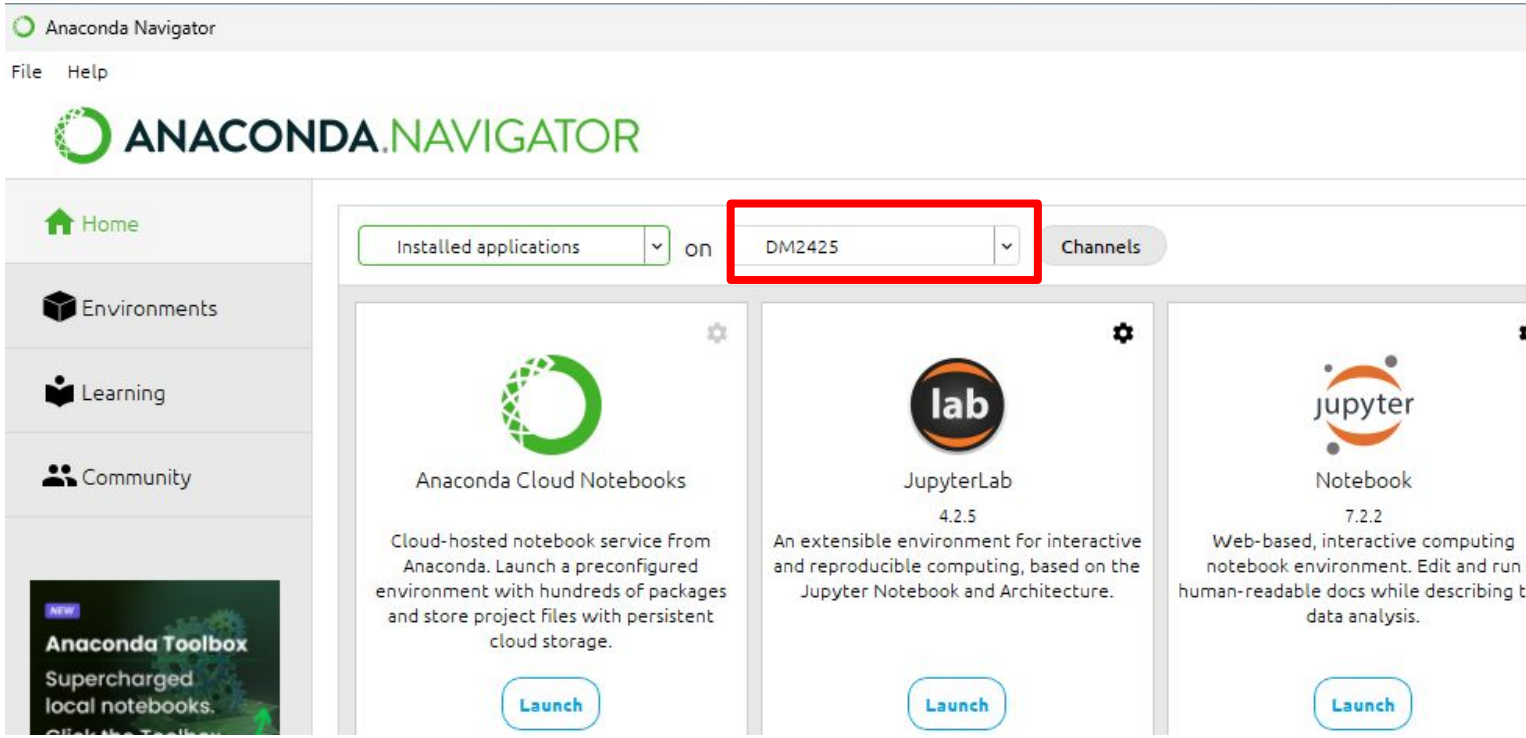
Don’t worry about understanding the code at this point

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

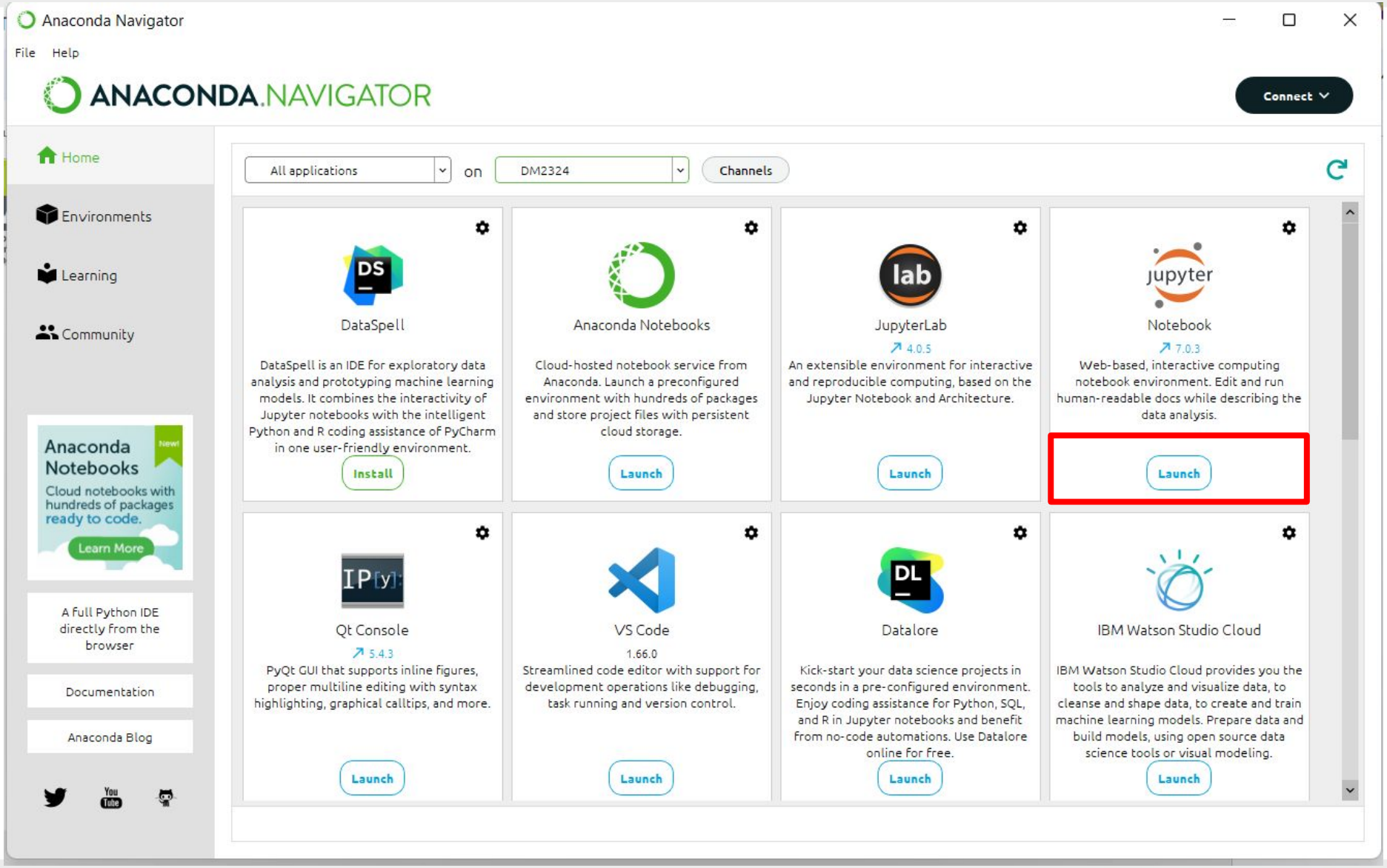
Test Jupyter notebook

Open Anaconda Navigator

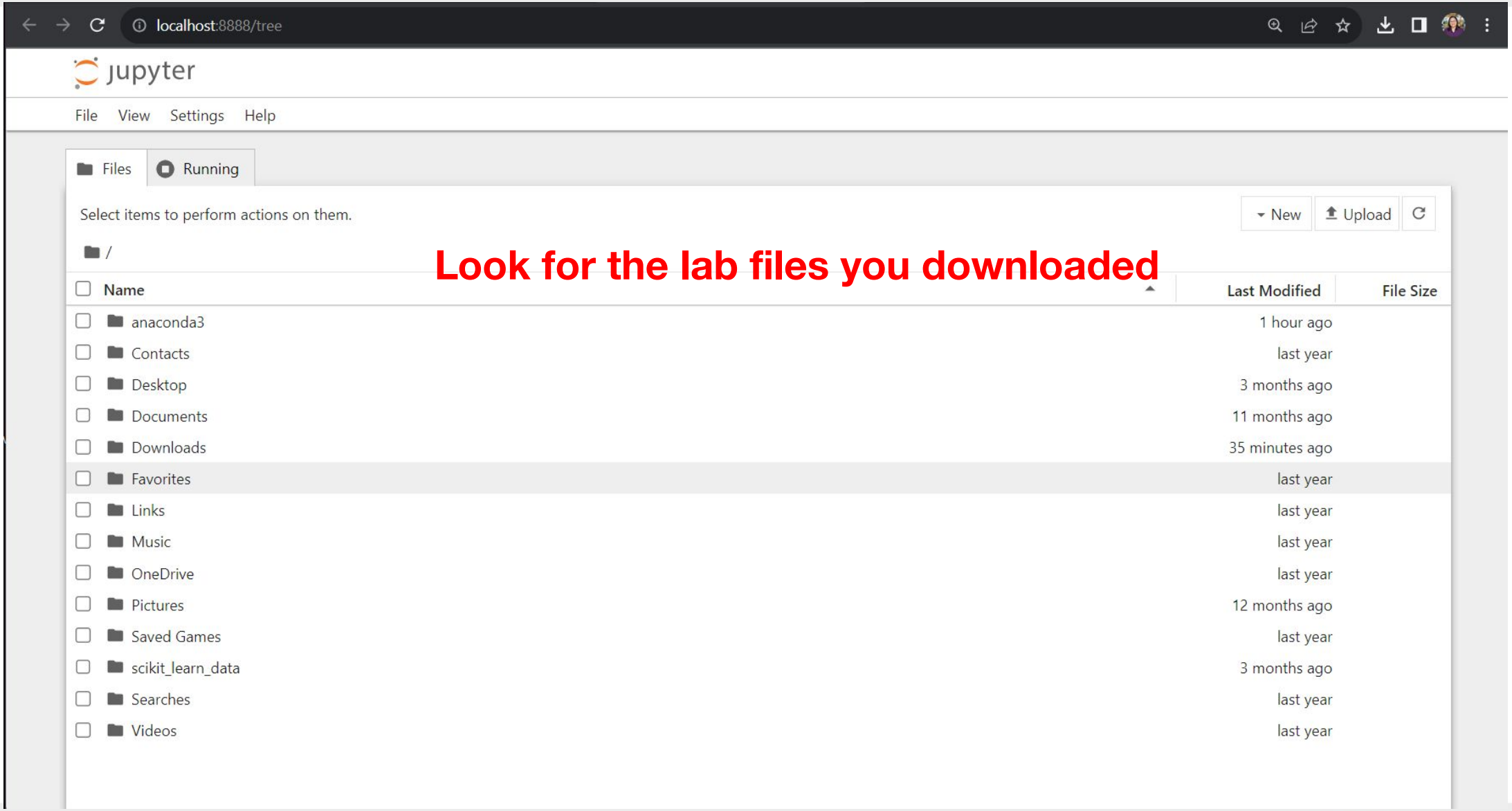
Make sure you select the environment we created



Test Jupyter notebook



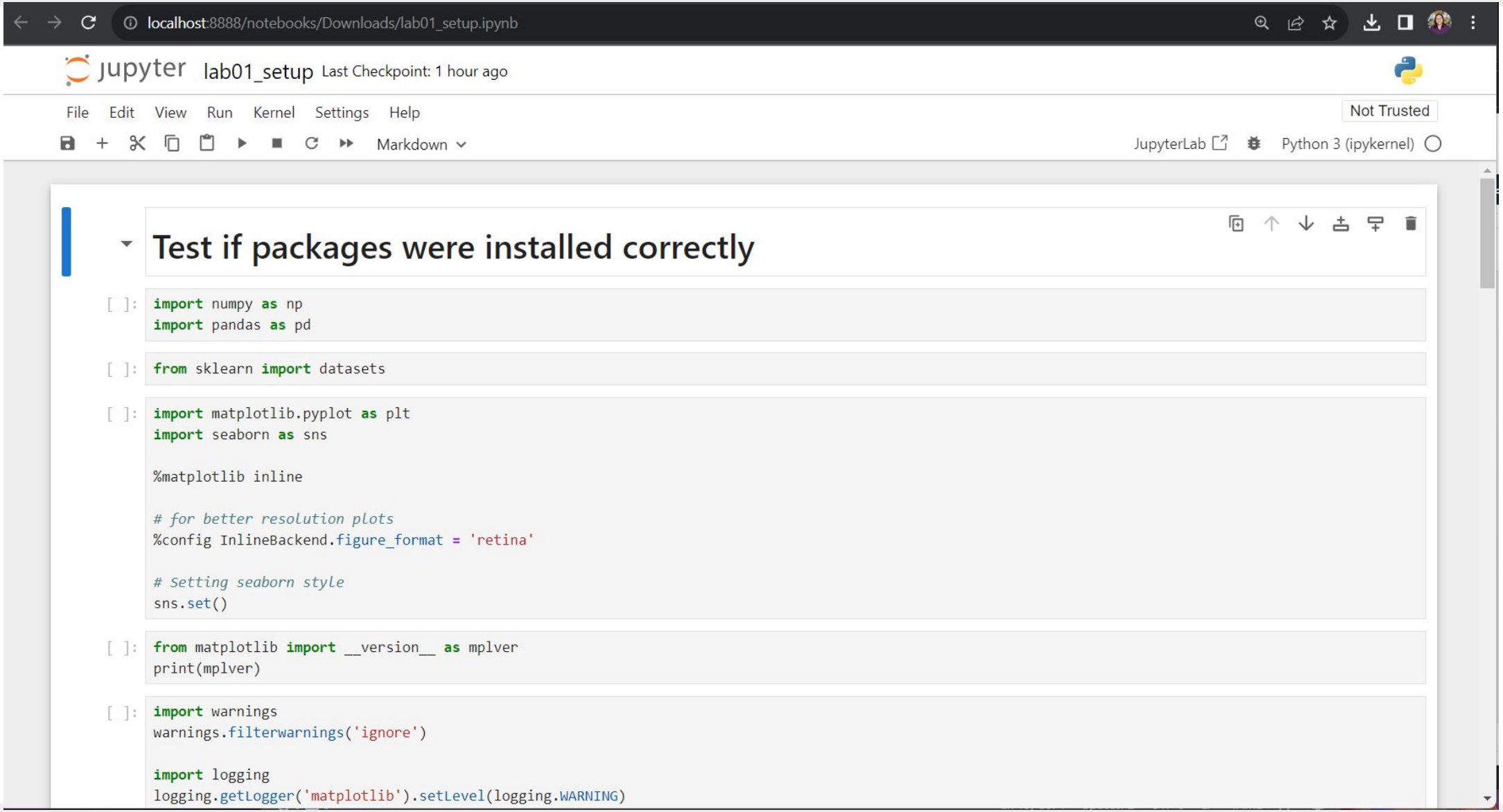
Test Jupyter notebook



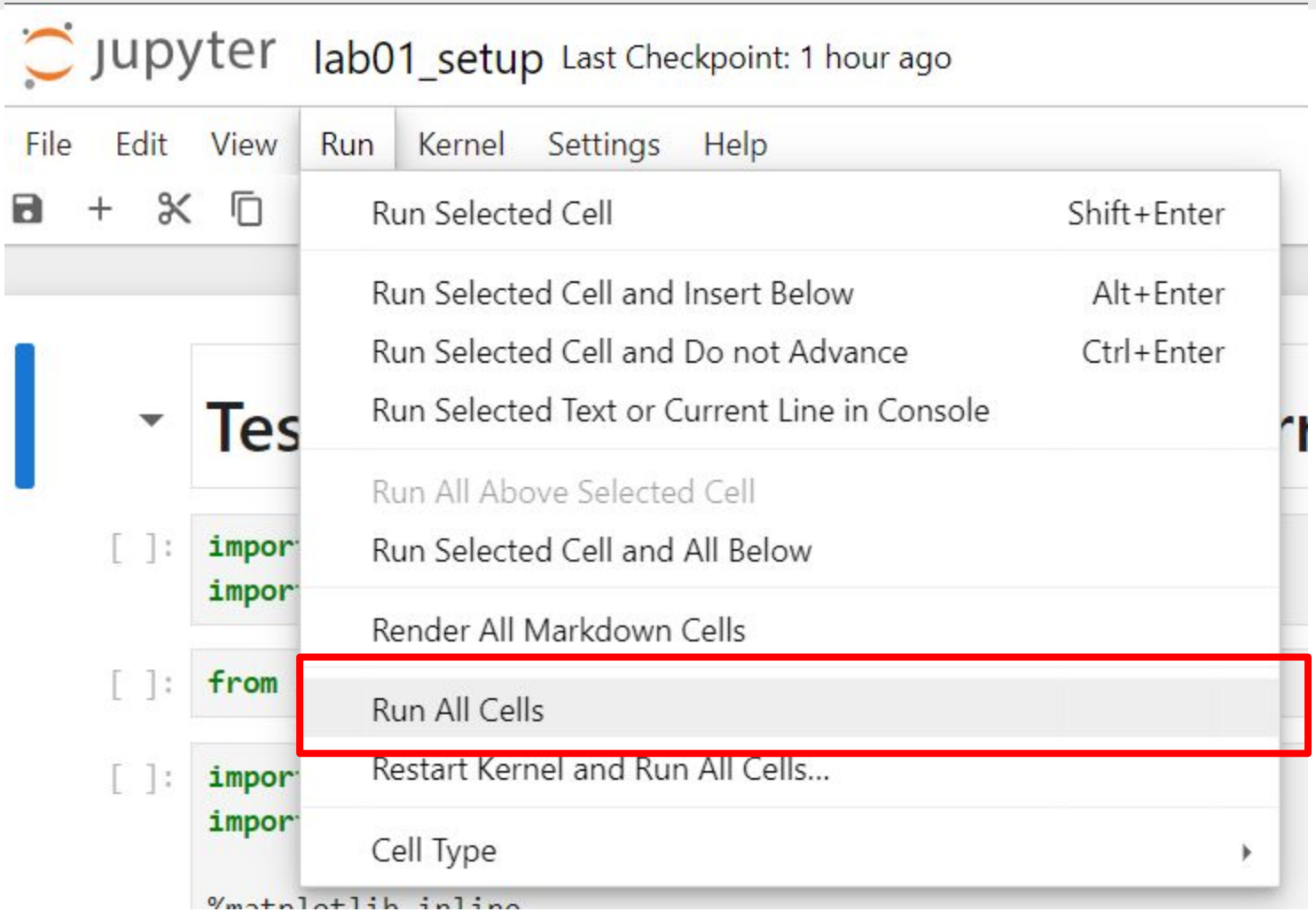
Don’t worry about understanding the code at this point

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



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 dm2425_env.yml
conda activate DM2425
```

Test Jupyter notebook + install other packages

Windows: Open Anaconda Prompt (miniconda3)

Linux/Mac: Open Terminal

```
conda activate DM2425
jupyter notebook
```

Follow the instructions in previous slides for testing Jupyter notebook and installing additional packages

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