Github basic commands

In this course we will only use basics functionalities — for a full tutorial see this

TO GET STARTED:

1. create a github account at https://github.com/

2. go to the course repository and fork it

   - A *fork* is a copy of a repository in your own account. Forking a repository allows you to freely experiment with changes without affecting the original project.

Github basic commands

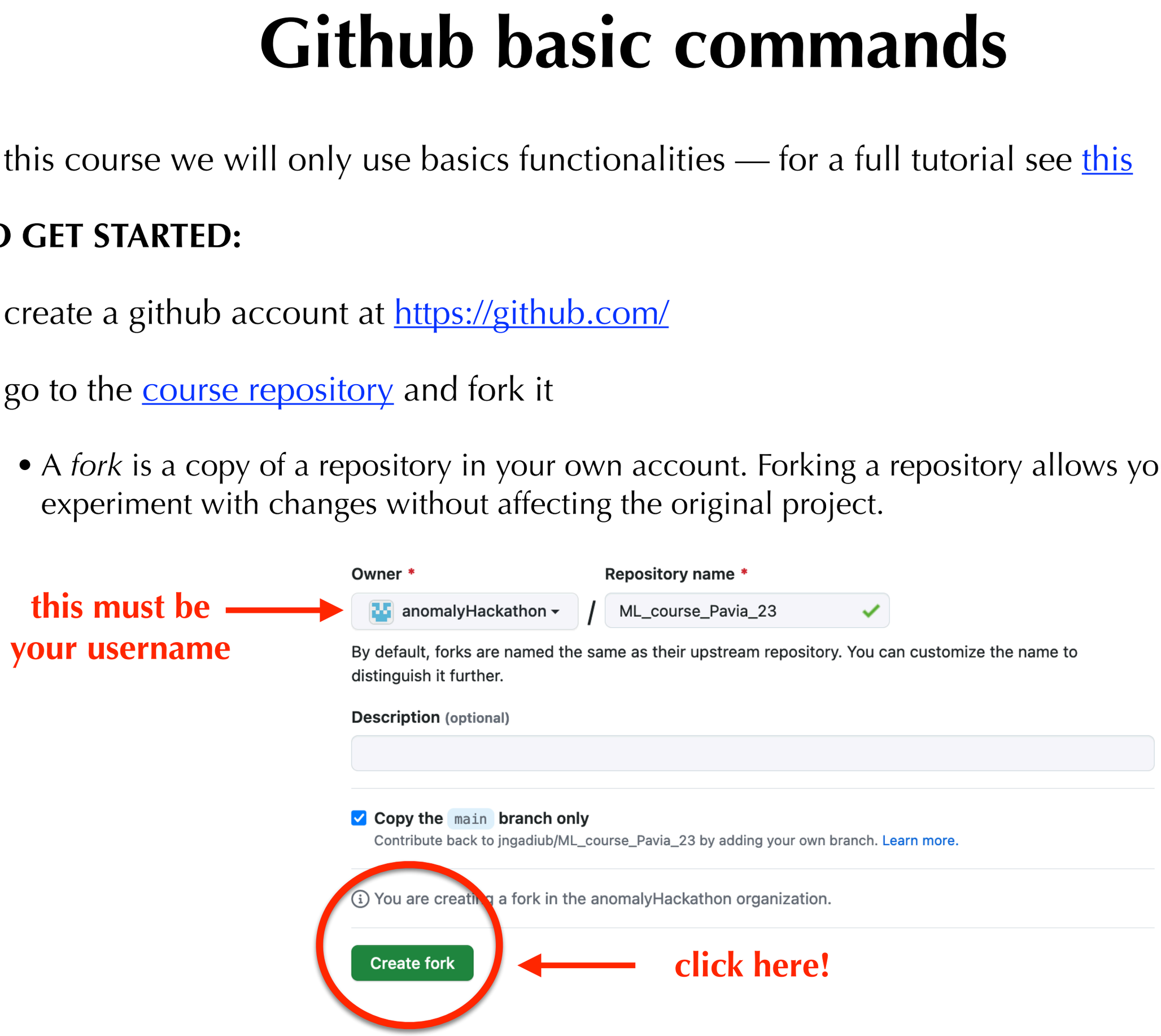
In this course we will only use basics functionalities — for a full tutorial see this

TO GET STARTED:

1. create a github account at https://github.com/

2. go to the course repository and fork it

   - A *fork* is a copy of a repository in your own account. Forking a repository allows you to freely experiment with changes without affecting the original project.



Github basic commands

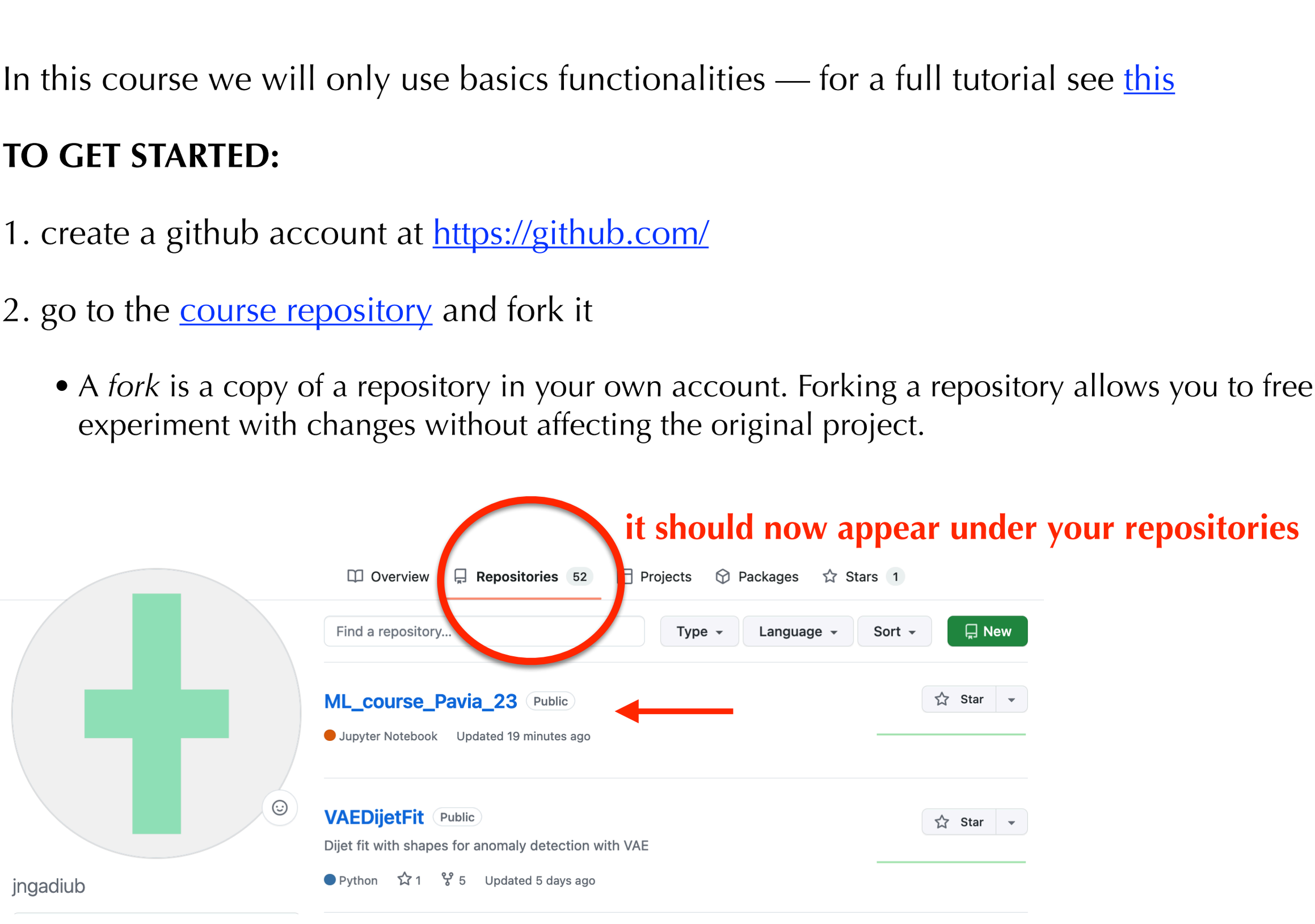
In this course we will only use basics functionalities — for a full tutorial see this

TO GET STARTED:

1. create a github account at https://github.com/

2. go to the course repository and fork it

   - A *fork* is a copy of a repository in your own account. Forking a repository allows you to freely experiment with changes without affecting the original project.



Github basic commands

- Before and throughout the course there will be changes to the original course repository and **you will need to keep your own fork of the repository up-to-date**

- To do that, you first need to install git on your computer (if it's not already installed) following these instructions

- Then, open a terminal, go to your preferred folder, and type the following commands:

```
git clone https://github.com/your-username/ML_course_Pavia_23.git
cd ML_course_Pavia_23
git remote add course https://github.com/jngadiub/ML_course_Pavia_23.git
git fetch course
git merge course/main
git push
```

- After the `git merge` command you will see a list of files that got changed in your local folder with respect to your remote (on github) repository

- The last `git push` command just pushes these changes to your remote repository

- At this point your local folder together with your remote repository should be fully synch with the original repository

Github basic commands

- At the end of the course you will be asked to do your own exercise which will require you to upload code and slides to your own fork of the repository

- Whenever you change or add a new file in your local ML_course_Pavia_23 folder and you want to save these local changes to your remote github repository (highly recommended) you go again to the terminal and type:

```
cd ML_course_Pavia_23
git status
git add fileX folderX …
git commit -m “whatever message explaining changes”
git push origin main
```

- With the `git status` command you can see the list of changes — be sure you add them all to the commit when using `git add`

- Keep in mind that your local/remote changes will merge with the changes in the original course repo when following the steps in previous slide 4 — this might raise conflicts and/or out-of-synch issues

  - highly suggested to do the steps in this slide only at the end of the course to give me access to your exercise

Setup

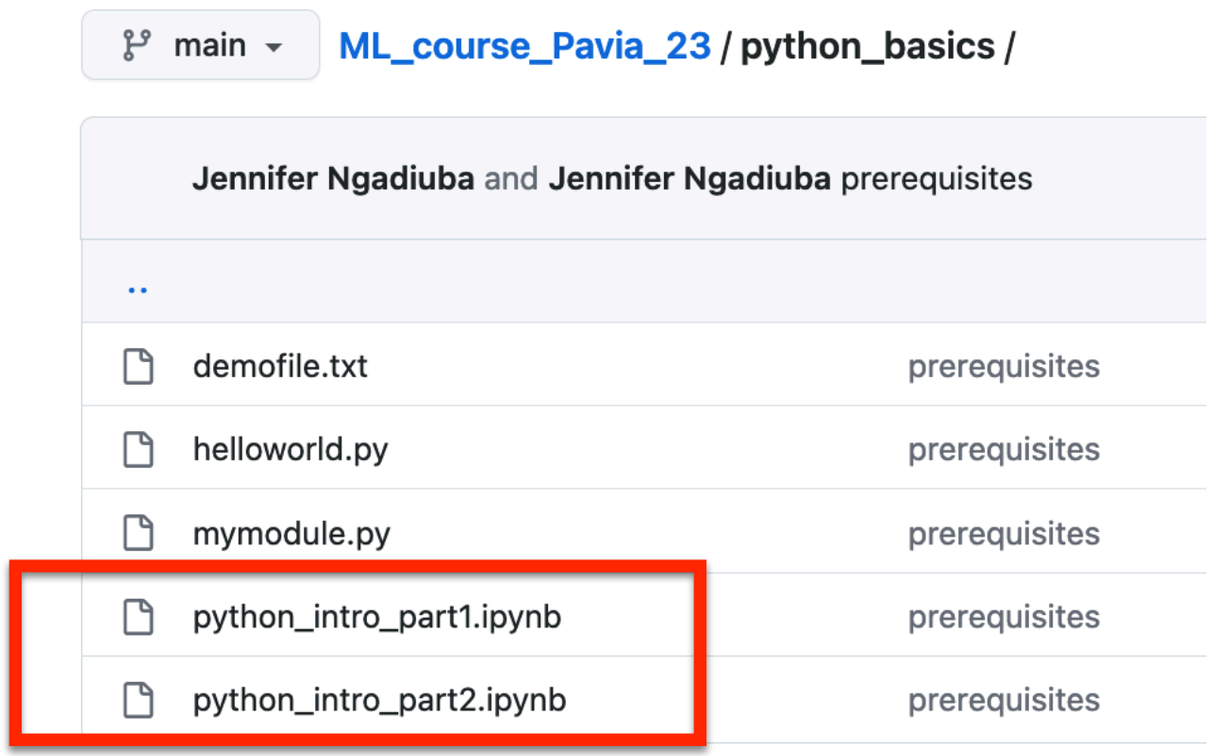
- We will be using **Colab** to run the hands-on part:

  - **Colab** is a free platform developed by Google to execute code on the cloud
    nb, you will need a google account

- In both setups: the interactive part is served with Python notebooks through **jupyter**

- If you're new to jupyter notebooks, select a cell and hit "shift + enter" to execute the code

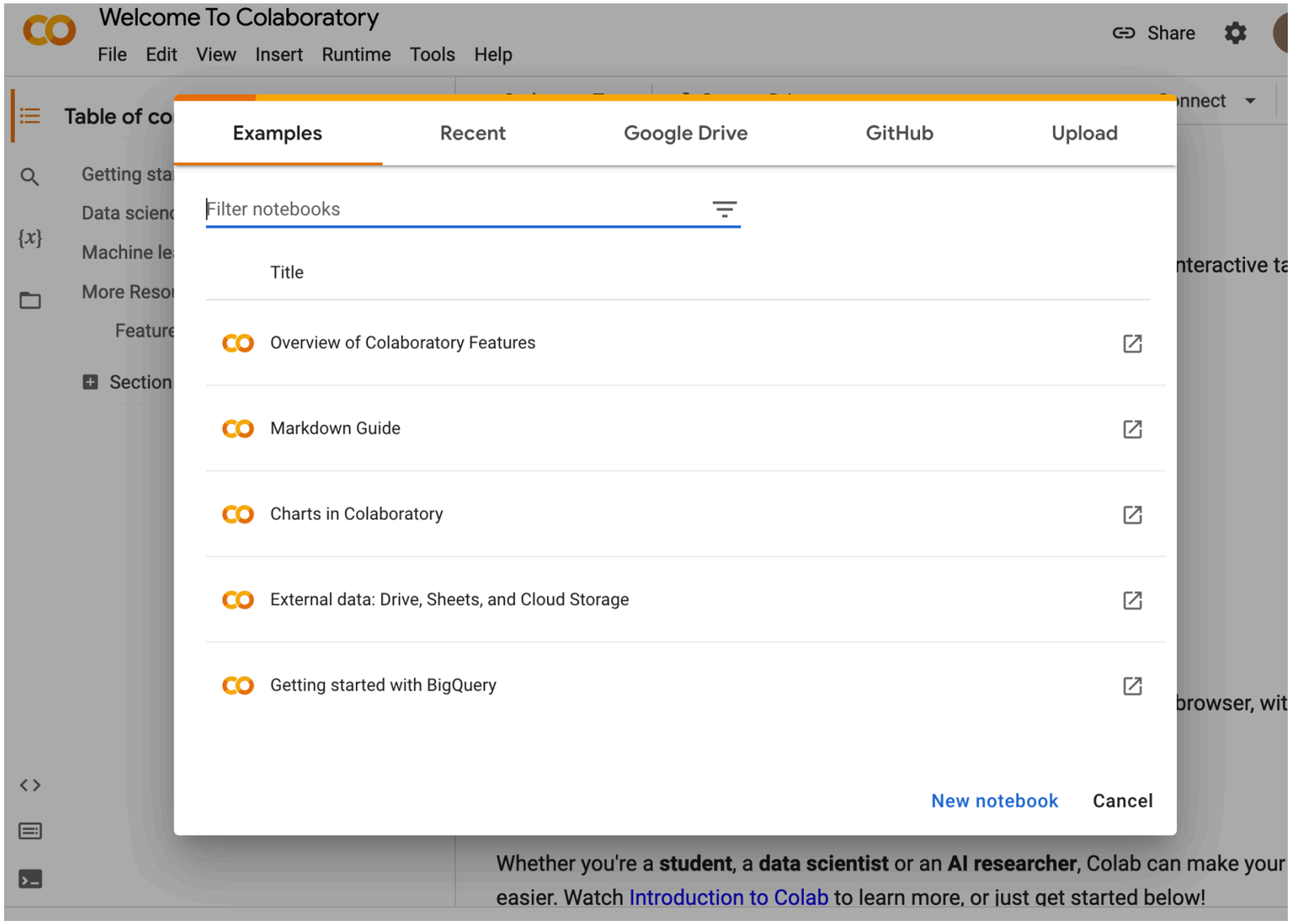
  - jupyter is rather intuitive but for a full tutorial see here



Running tutorial notebooks in Colab

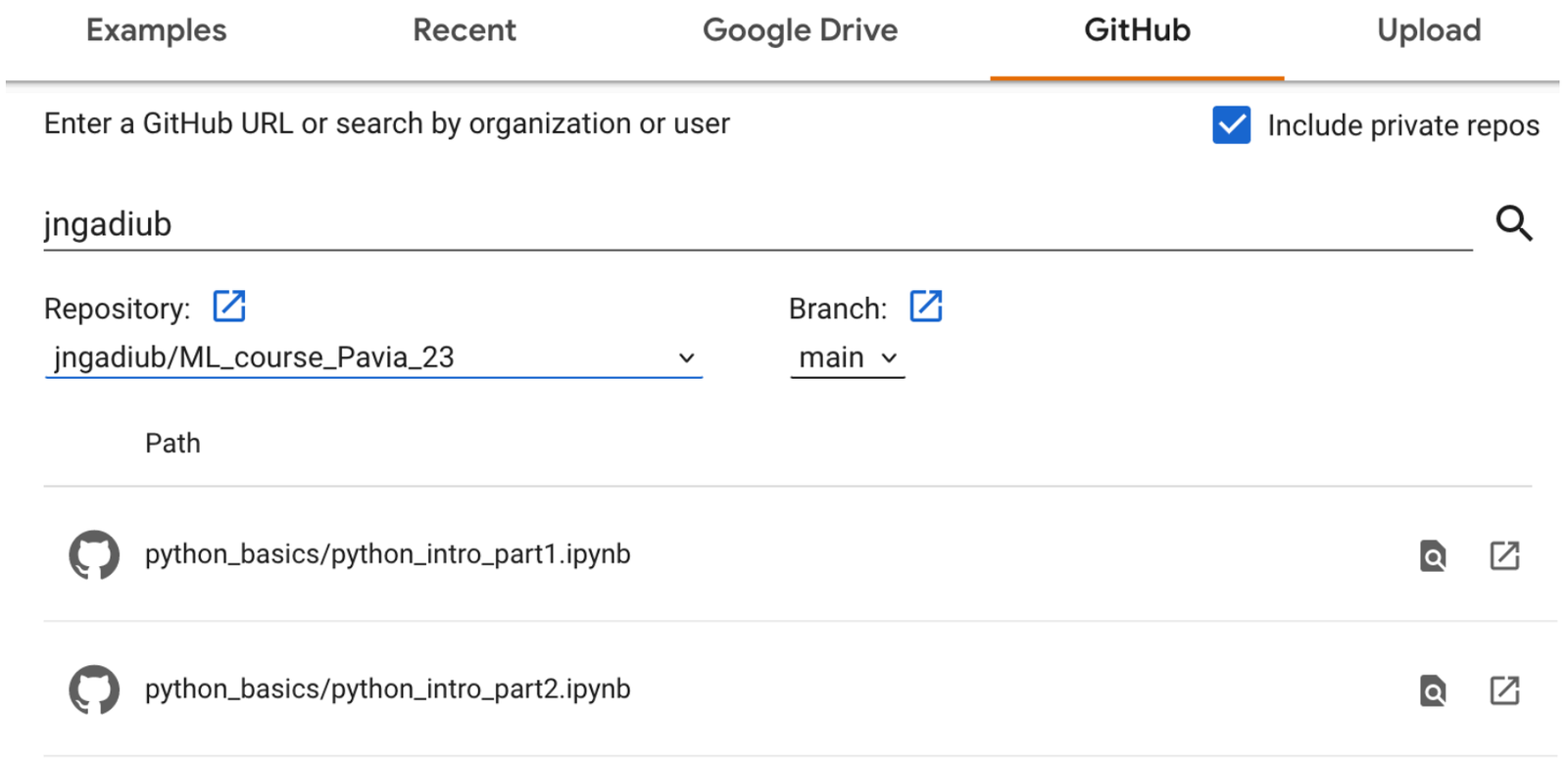
Step1: open notebook

- Make sure you have a Google account
- Go to: https://colab.research.google.com/

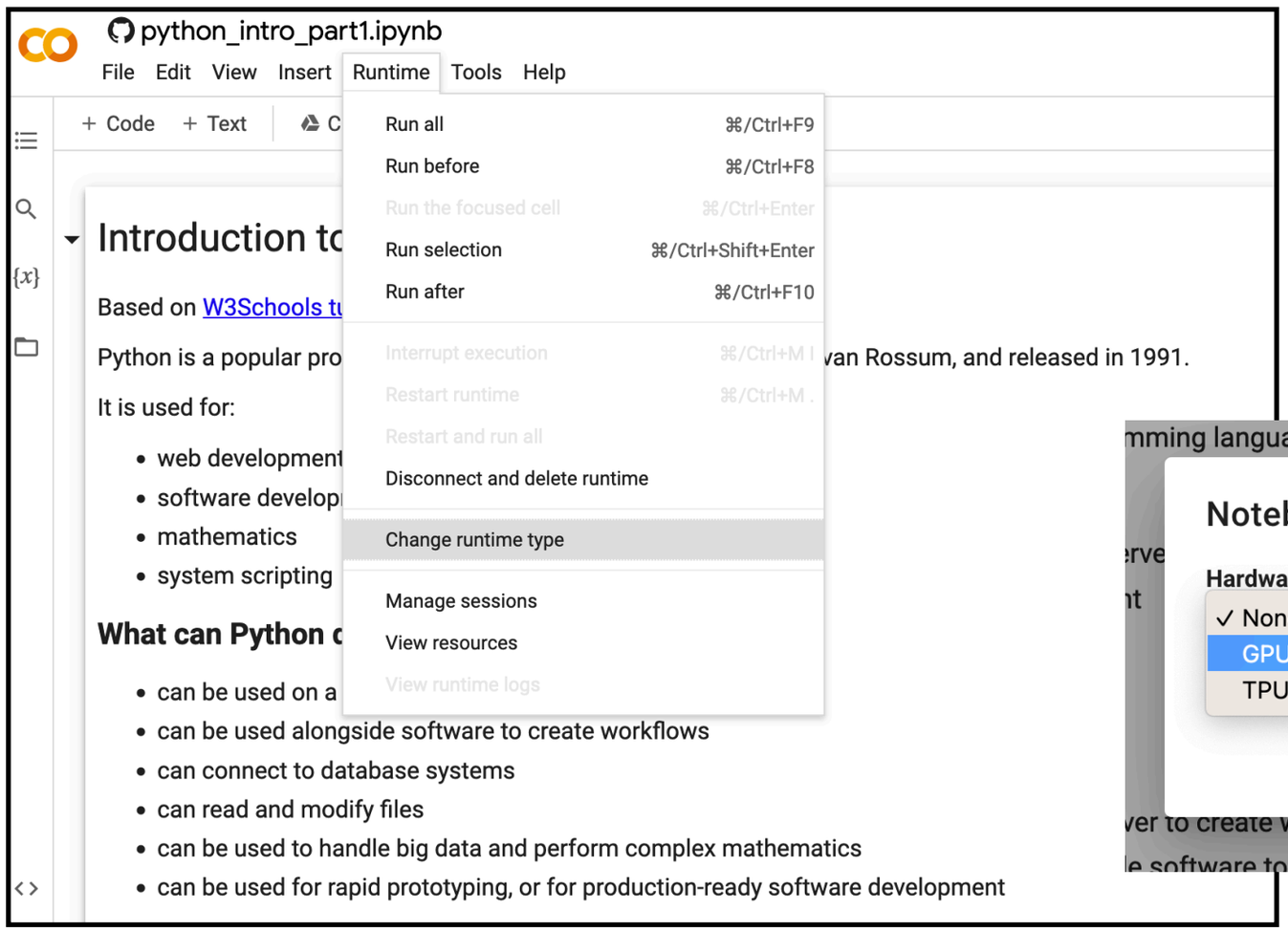


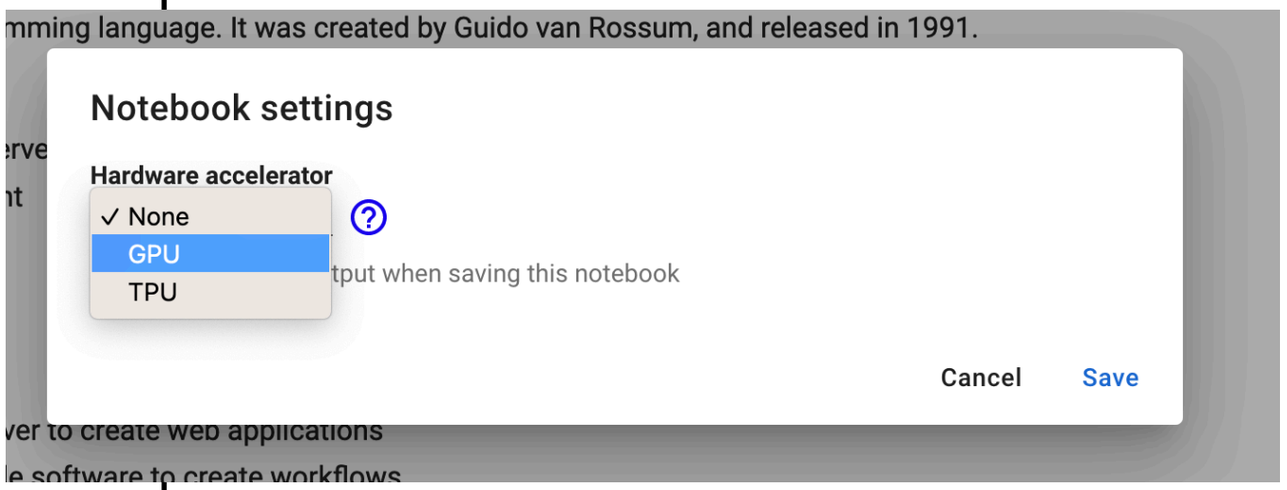
Step2: import from github

- Click on the `GitHub` tab
- Specify the repository `jngadiub/ML_course_Pavia_23`
  - if you are experienced with github you can instead specify your own fork such that you will be able to save any changes you might apply to the original course while being in synch
- Click on one of the `.ipynb` notebooks



Step3: use GPUs





Prerequisites notebooks

- We will use python for the whole course — if you are not familiar with it you must go through some of the basics functionalities by running these two notebooks in Colab (see previous slides) before the course:
  - `python_basics/python_intro_part1.ipynb`
  - `python_basics/python_intro_part2.ipynb`
- The notebooks also contain a few optional simple exercises to help you getting more familiar with it (let me know if you have issues)