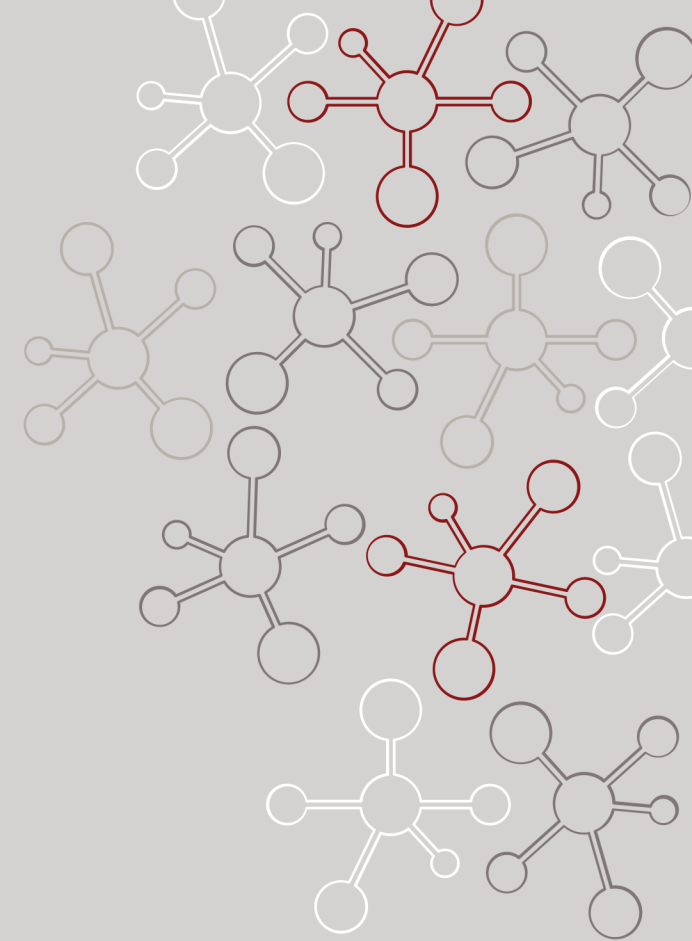


Getting Started

Michael Kagan
SLAC



Github

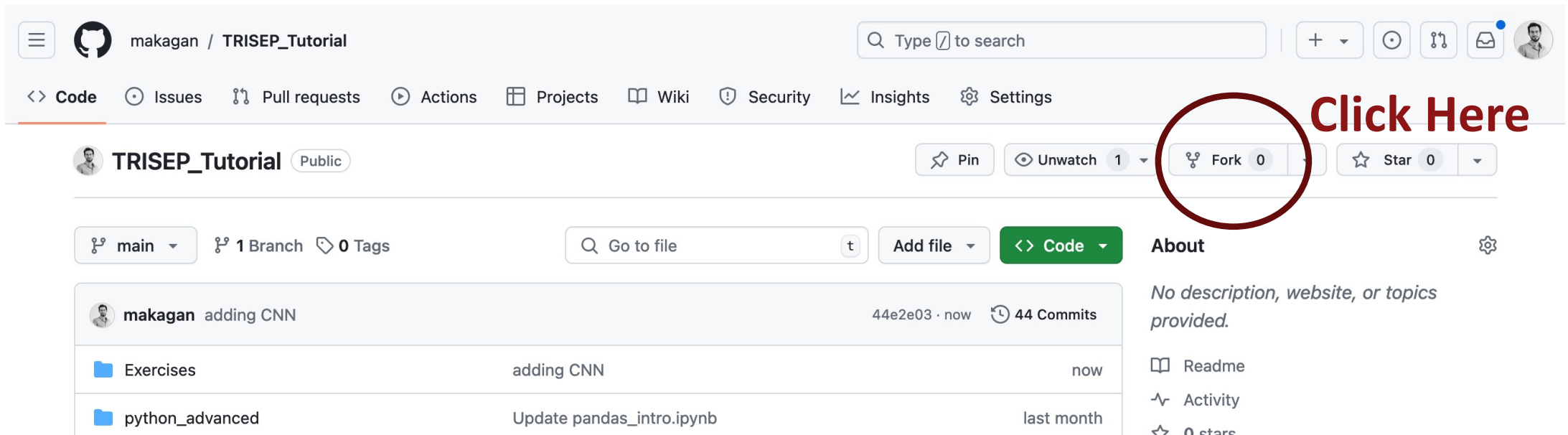
Github Basics

3

We will use Github for accessing the project code. For a full tutorial, see [here](#)

To get started:

1. Create Github account: <https://github.com/>
2. Got to the [project repository](#) and fork it
 - A fork is a copy of the repository (repo) in your own account, allowing you to edit and experiment without changing original repo.



The screenshot shows the GitHub interface for the repository 'makagan / TRISEP_Tutorial'. The repository is public and has 1 branch and 0 tags. The 'Fork' button is circled in red, and the text 'Click Here' is written in red next to it. The repository has 44 commits and 0 stars. The repository description is 'No description, website, or topics provided.' The repository contains two folders: 'Exercises' and 'python_advanced'. The 'Exercises' folder has a commit 'adding CNN' from 'makagan' 44e2e03 · now. The 'python_advanced' folder has a commit 'Update pandas_intro.ipynb' from 'makagan' last month.

makagan / TRISEP_Tutorial

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

TRISEP_Tutorial Public

Pin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

makagan adding CNN 44e2e03 · now 44 Commits

Exercises	adding CNN	now
python_advanced	Update pandas_intro.ipynb	last month

About

No description, website, or topics provided.

Readme Activity 0 stars

Github Basics

4

We will use Github for accessing the project code. For a full tutorial, see [here](#)

To get started:


1. Create Github account: <https://github.com/>
2. Got to the [project repository](#) and fork it

Create a new fork

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

Required fields are marked with an asterisk (*).

Owner * Repository name *

Choose an owner ▼ / TRISEP_Tutorial 

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

☒ Copy the `main` branch only

Contribute back to makagan/TRISEP_Tutorial by adding your own branch. [Learn more.](#)

Create fork

Click Here

Must be your username



Github Basics

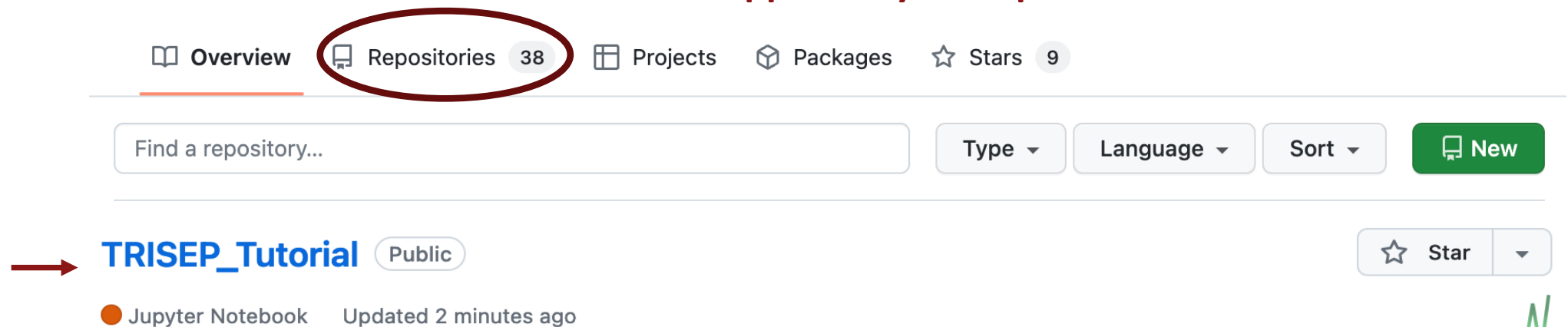
5

We will use Github for accessing the project code. For a full tutorial, see [here](#)

To get started:

1. Create Github account: <https://github.com/>
2. Got to the [project repository](#) and fork it

Should now appear in your repositories



Github Commands I

6

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 repo up-to-date**

You first need to install git on your computer (if it's not already installed) following [these](#) instructions

Open a terminal, go to your preferred folder, and type the commands:

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

Github Commands II

7

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 Commands III

8

Whenever you change or add a new file in your local `SSI_Projects` 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 SSI_Projects
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

Google Colab

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: **you will need a google account**

The project materials are served with Python notebooks through **jupyter**

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

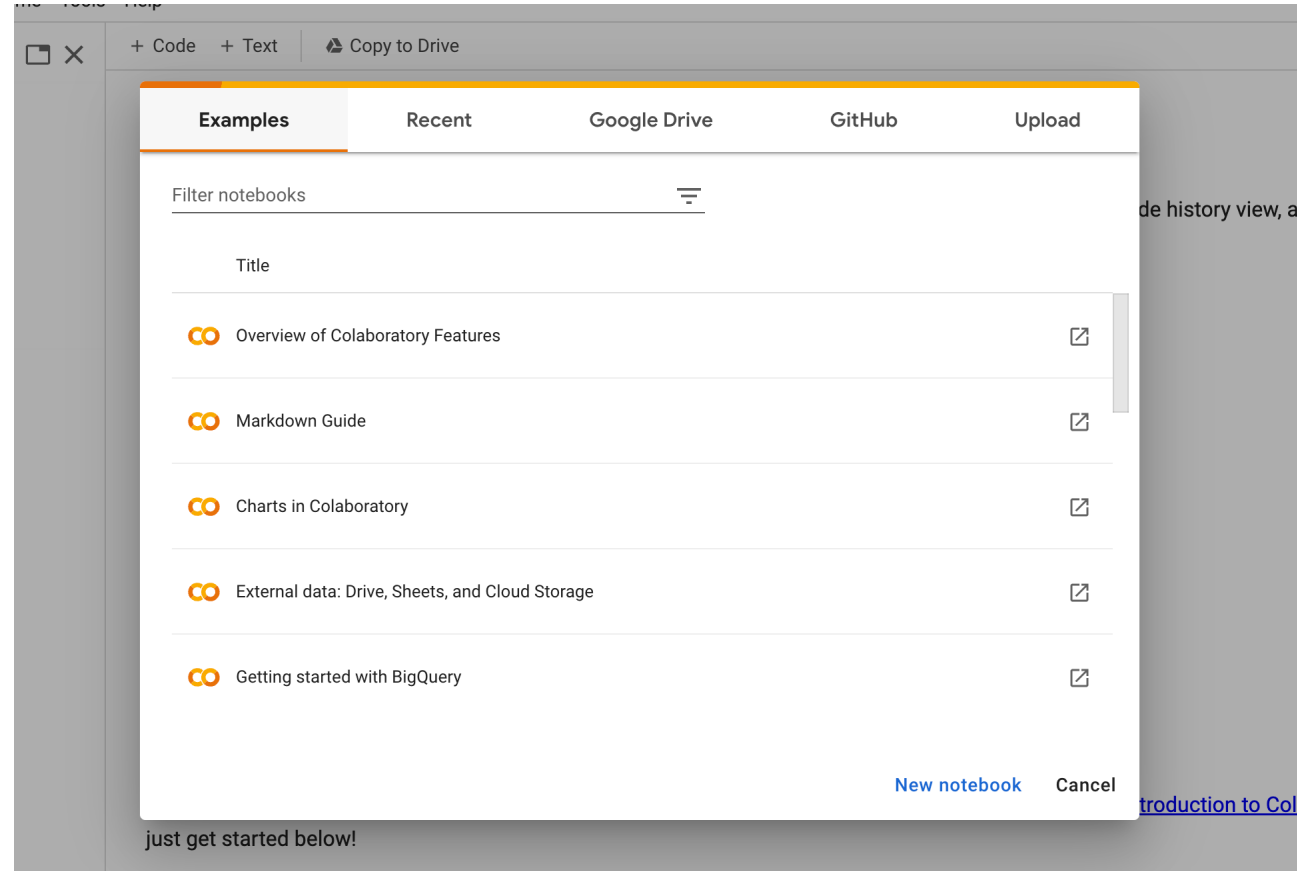
For a full jupyter tutorial see [here](#)

Step 1

11

Make sure you have a Google account

Go to: <https://colab.research.google.com/>



Step 2: Import from Github

12

Click on the `GitHub` tab

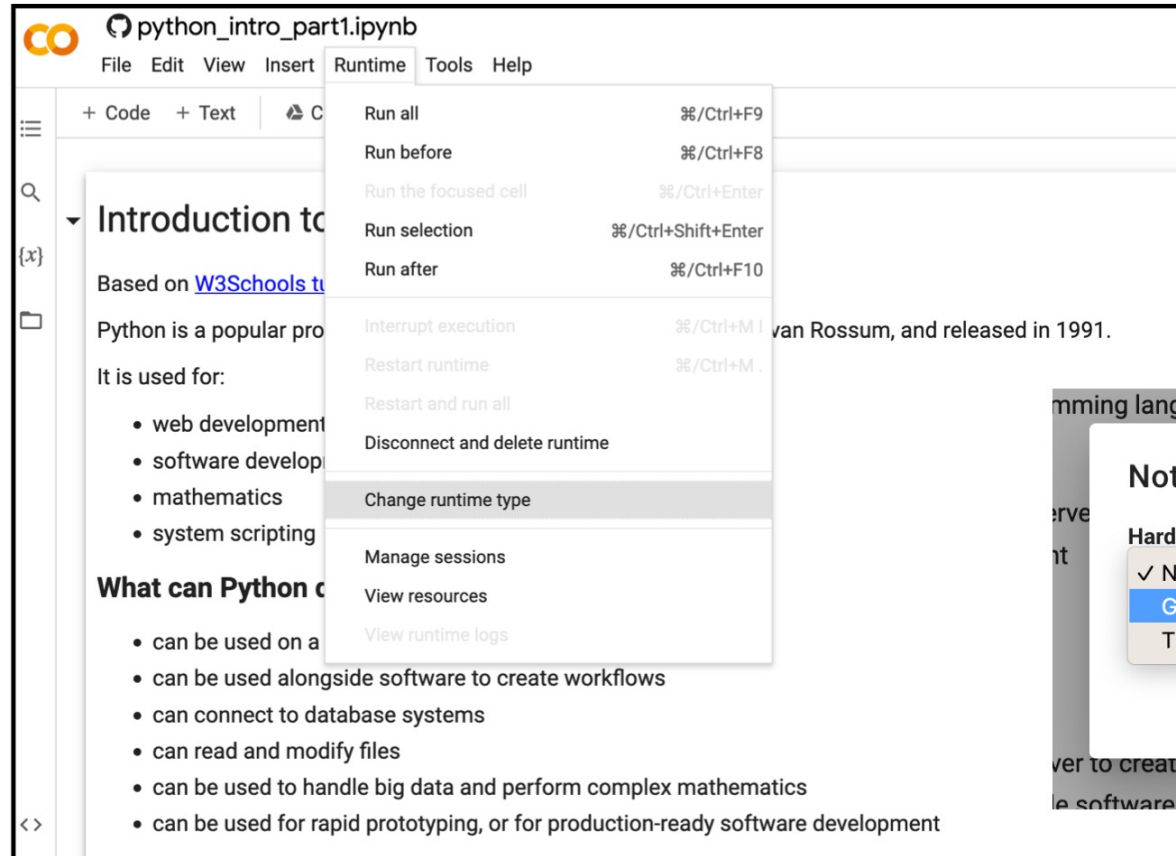
Specify the repo: `makagan/TRISEP_Tutorial`

- NOTE: you can specify your own fork of the repo, so you can save changes

Click on one of the `.ipynb` notebooks

Step 3: Use GPU

13



The screenshot shows a Jupyter Notebook titled "python_intro_part1.ipynb". The "Runtime" menu is open, displaying various execution options with their corresponding keyboard shortcuts. The options include "Run all", "Run before", "Run the focused cell", "Run selection", "Run after", "Interrupt execution", "Restart runtime", "Restart and run all", "Disconnect and delete runtime", "Change runtime type", "Manage sessions", "View resources", and "View runtime logs". The "Change runtime type" option is highlighted. The notebook content is partially visible, showing an introduction to Python.

python_intro_part1.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text

Introduction to Python

Based on [W3Schools tutorial](#)

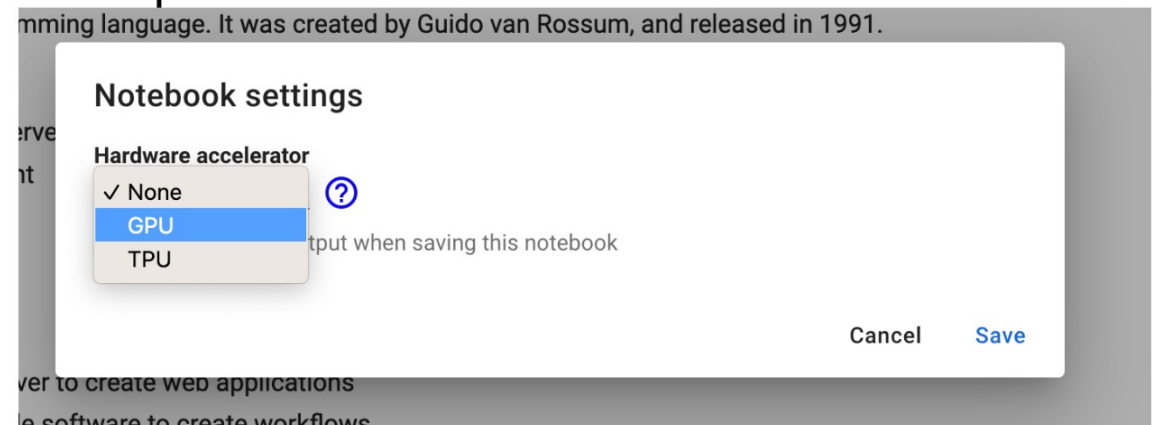
Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development
- software development
- mathematics
- system scripting

What can Python do?

- can be used on a wide variety of platforms
- can be used alongside software to create workflows
- can connect to database systems
- can read and modify files
- can be used to handle big data and perform complex mathematics
- can be used for rapid prototyping, or for production-ready software development



The screenshot shows the "Notebook settings" dialog box. The "Hardware accelerator" section is expanded, showing three options: "None" (selected with a checkmark), "GPU", and "TPU". A question mark icon is next to the "GPU" option. The "Save" button is highlighted in blue.

Notebook settings

Hardware accelerator

- ✓ None
- GPU
- TPU

Cancel Save

Prerequisites

We will use python for the projects

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)