BIOS-584 Python Programming (Non-Bios Student)

Week 01 – Understand Git and GitHub

Instructor: Tianwen Ma, Ph.D.

Department of Biostatistics and Bioinformatics,
Rollins School of Public Health,
Emory University

Git

- Open source and free source control management (SCM)
- Allow tracking changes in your codebase, collaborate with others, and manage your projects
- Standard for version control
- A powerful tool but a bit challenging to use for beginners
- GitHub kicks in to help.

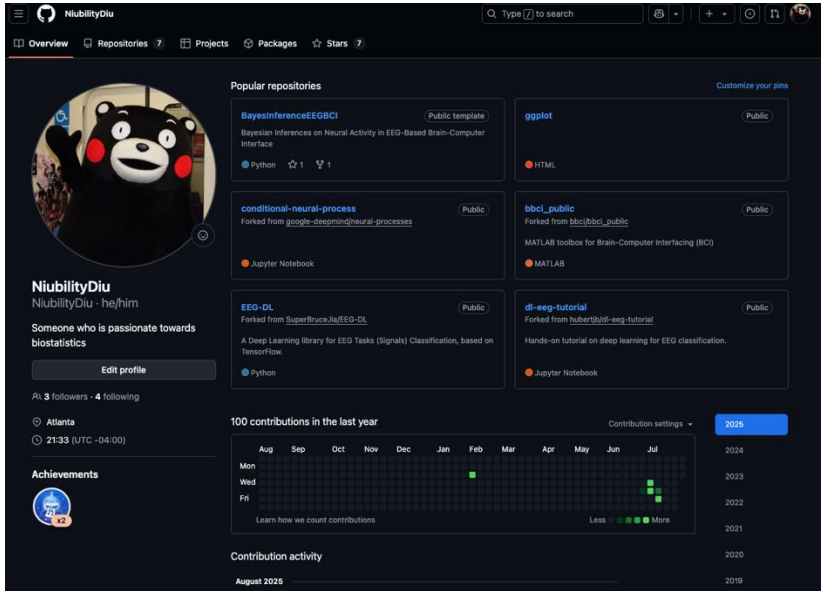
GitHub

- A web-based platform to provide hosting for Git repositories, allowing you to store different versions of your code in the cloud and collaborate with others.
- Other features including issue tracking, project management, and continuous integration.
- It can even host your website for free.
- Social network for developers, where you can follow other developers, star their projects, and contribute to open-source projects.

Get Started with GitHub

Create a GitHub Account

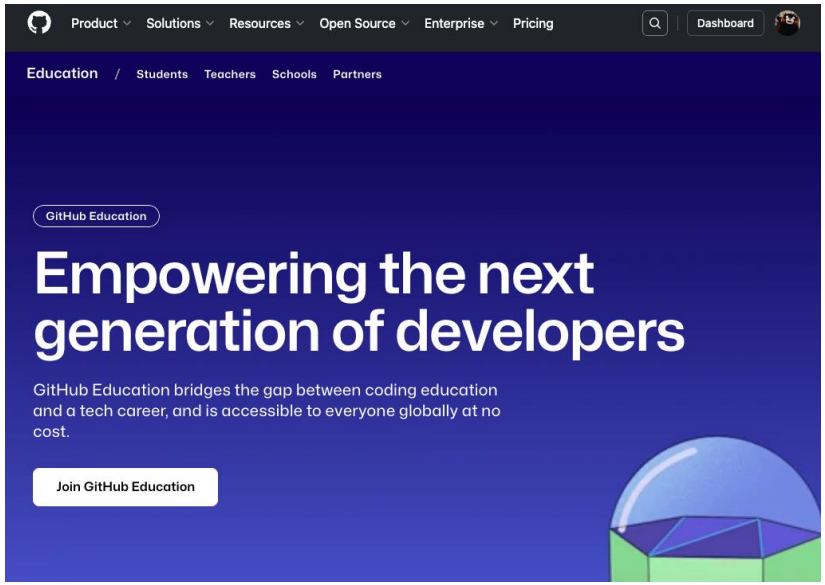
- https://github.com/
- Free educational account with Emory email address
- Be careful with your username as it is visible to the public



This is my profile page.

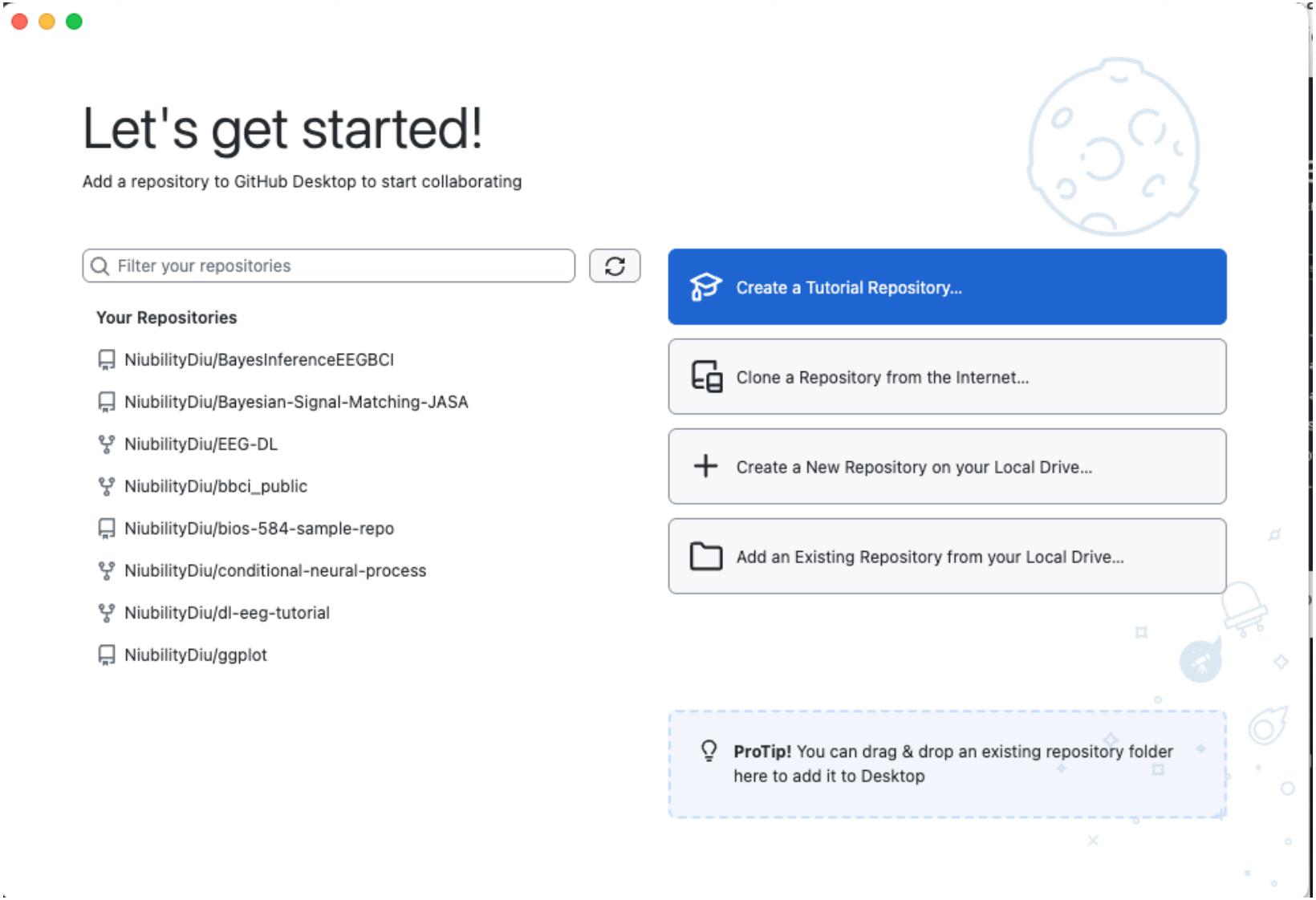
Apply to GitHub Education

- https://github.com/education for GitHub Student Developer Pack.
- Free domain name, cloud hosting, etc.
- Verify your student status



Download and Install GitHub Desktop

- https://desktop.github.com/
- Helpful graphic user interface (GUI) for Git
- Available for Windows and macOS.



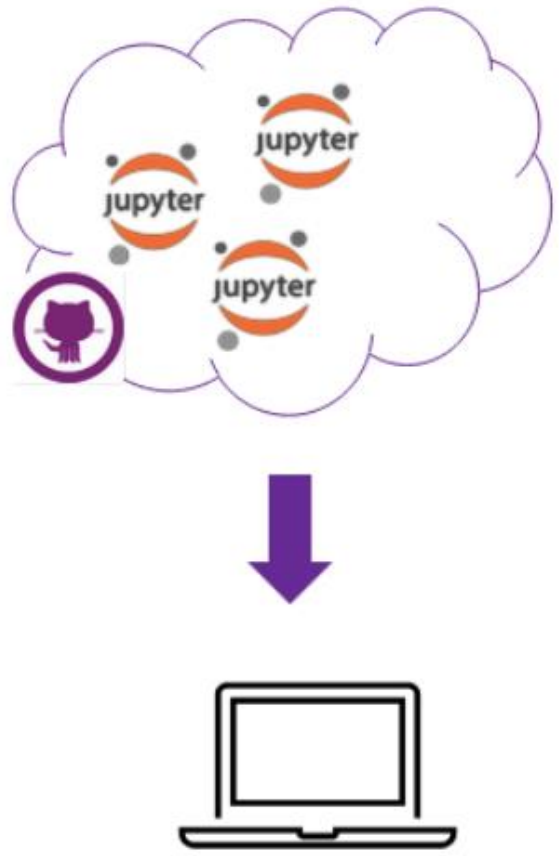
You should see a similar page to this when you first click on the GitHub application.

Access and Edit Remote Repo

- Access: Download a remote GitHub repository to your local computer to work on it
- Edit: Make local changes to the repository and update them to the cloud
- Two ways:
  - Cloning
  - Forking – Use forking in this course

Cloning

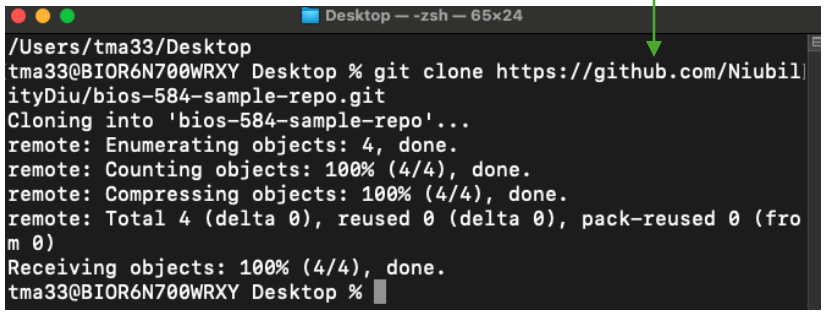
- Make a local copy of a repository
- Download a repository to your local computer
- Unlike forks, clones reference the original repository.
  - you cannot push changes to it unless you have permissions from the owner



Clone Example

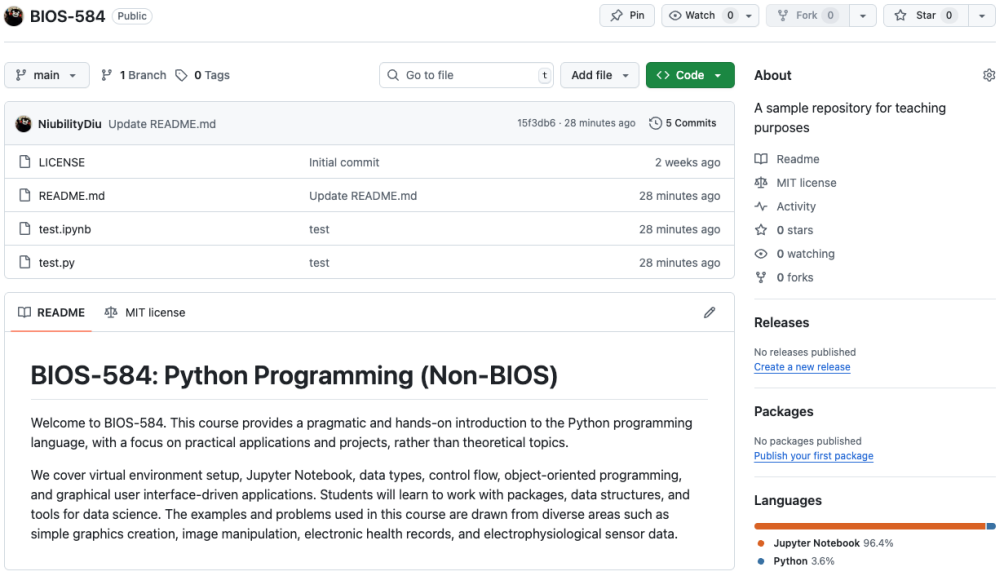
- Example repo:
  - https://github.com/NiubilityDiu/BIOS-584
- Can clone the repo using GitHub desktop
  - Does not create a remote copy of the repo in my GitHub page

Sorry that I changed the name.

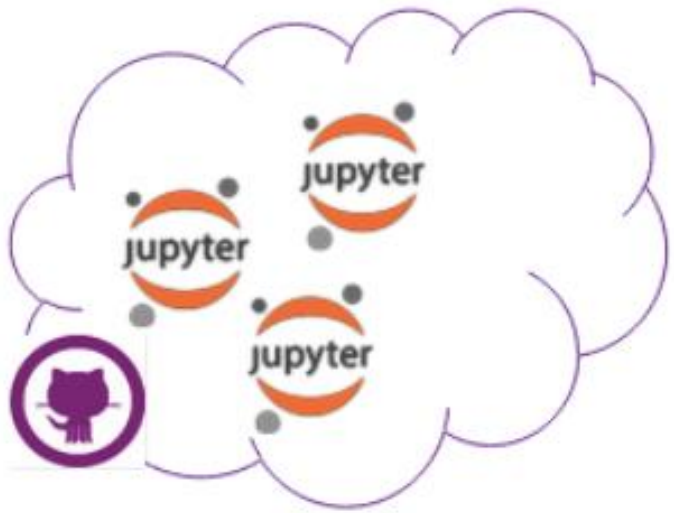


Clone Example

- If user cloning the repo have write access, they can replace files and folders in the remote GitHub repo
- The purpose of cloning is to contribute directly to the original repo or work offline

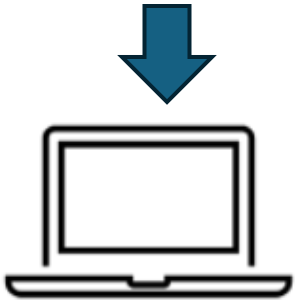


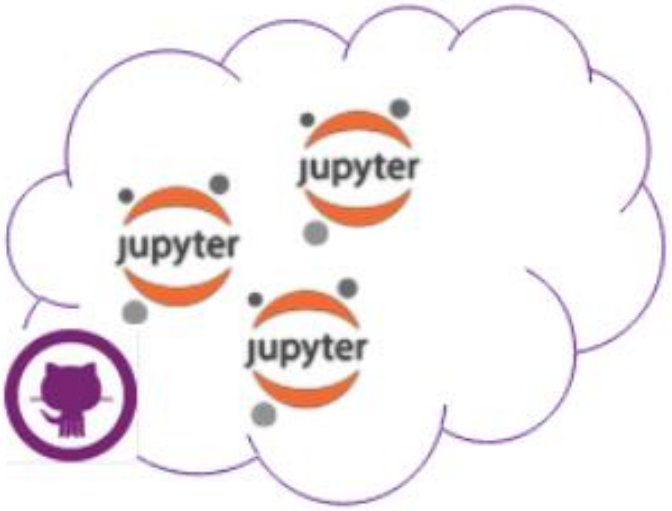
Cloning Example



github.com/student1/bios-584-sample-repo

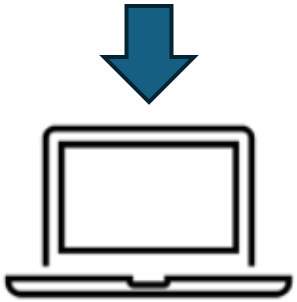
CLONE



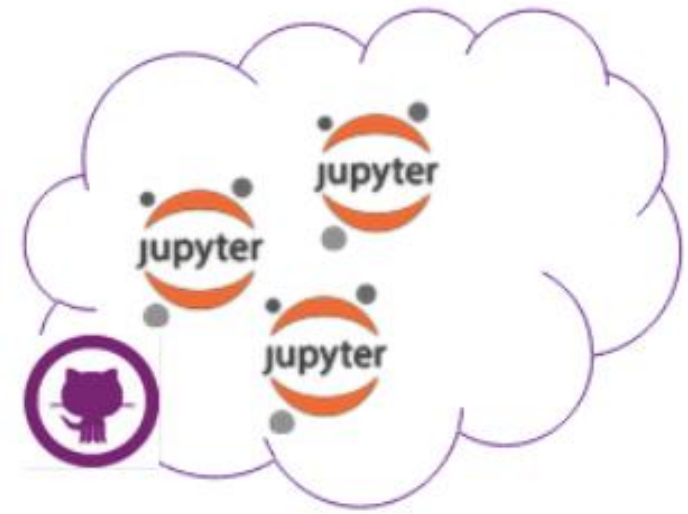


github.com/student2/bios-584-sample-repo

CLONE

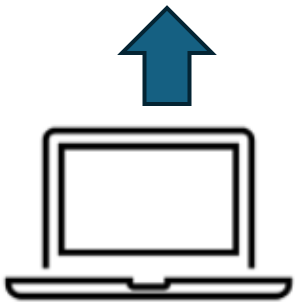


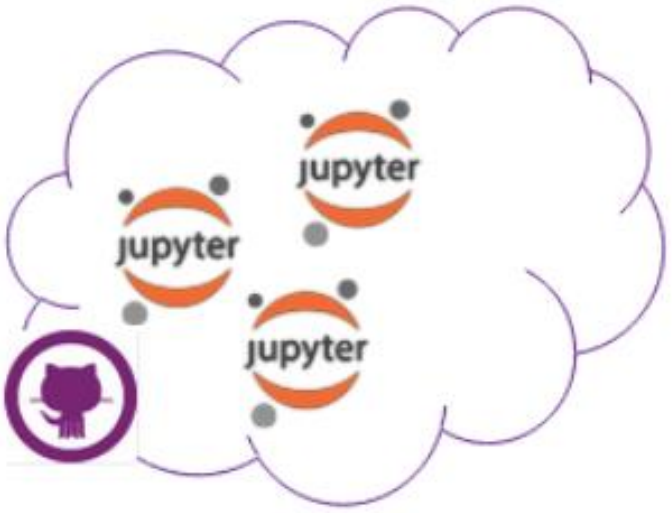
Cloning Example



github.com/student1/bios-584-sample-repo

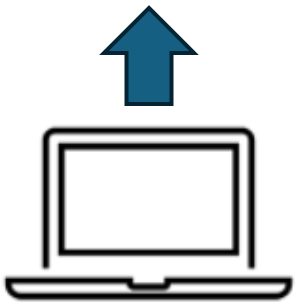
PUSH





github.com/student2/bios-584-sample-repo

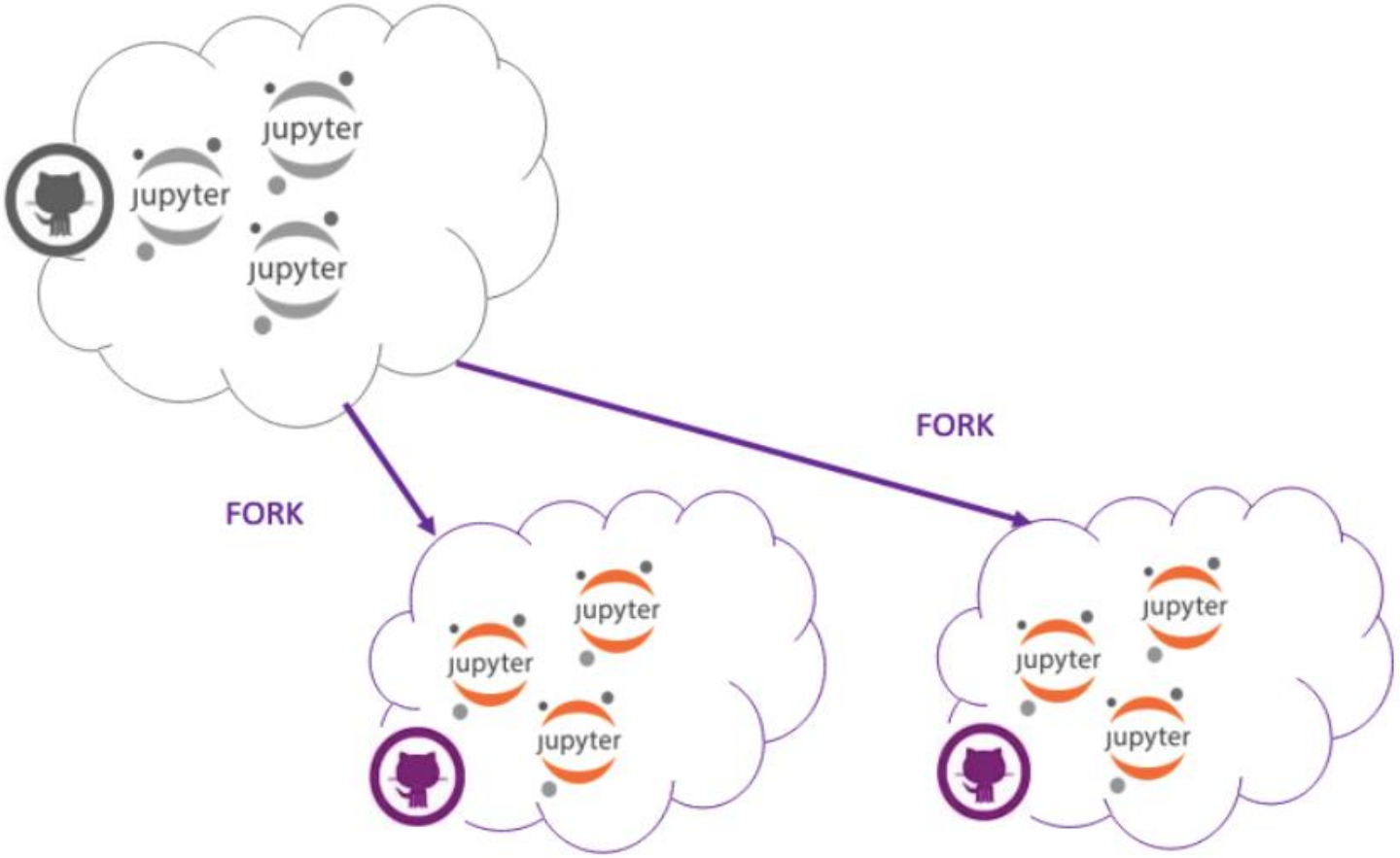
PUSH



Forking

- Create your own copy of a repository in your remote GitHub page
- You can contribute changes to your copy without affecting the original repository.
- Often used to create a personal version of a project for custom modifications or development
  - You can then submit a pull request to the original repository to suggest changes

Forking Illustration

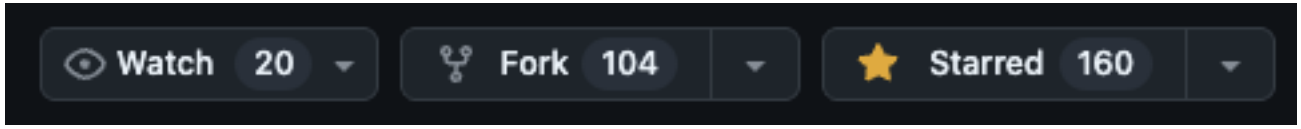


github.com/student1/bios-584-sample-repo

github.com/student2/bios-584-sample-repo

Forking Example

- Fork the repo using the GitHub page
  - Create a remote copy of the repo on your GitHub
  - Then clone the repo to the local computer and work on it
  - Push changes to your remote repo
  - Or submit a pull request to the original repo to suggest changes
- You cannot fork your own repo.
- Click on “fork” button and select the destination if you belong to multiple organizations.



Forking

- The changes made by each student are not shared across different websites.
- Over time, the repos could look different.

Comparing cloning and forking

Feature	Cloning	Forking
Local Copy	Yes	No (Initially only on GitHub)
Direct Collaboration	Yes (if granted with permission)	No (requires pull requests, (PRs))
Synchronization	Yes (with original repo)	Yes (can fetch updates from the original)
Control	Limited (depend on permissions)	Full (over the forked copy)
Use Case	Contributing directly, local development	Independent development, contributing via PRs
Includes	All files, branches, commit history	All files, commit history, but no issues or PRs

Remove cloned and forked repos

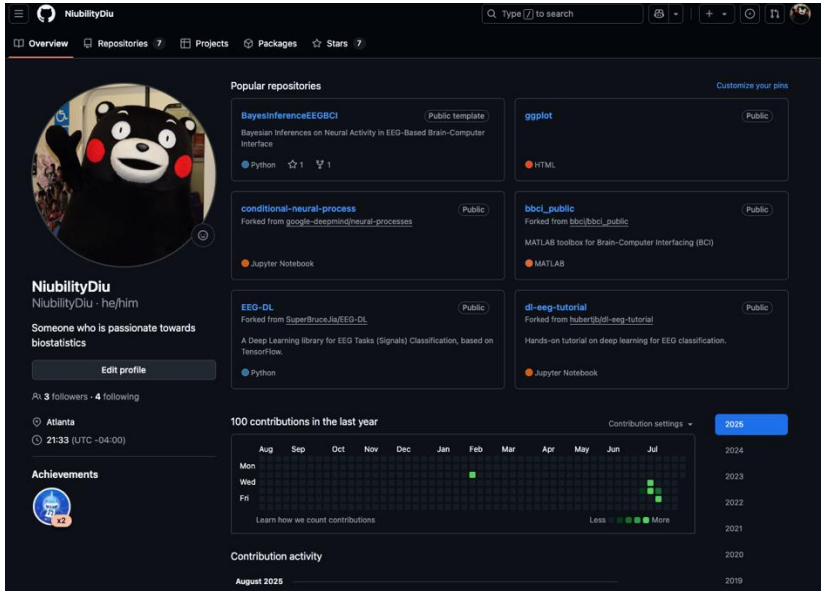
- If you no longer need your cloned or forked repos, you can delete the local copy.
- Deleting the local copy does not affect the remote repo.
- You can also delete the remote repo if you have written access to it.
  - Always be careful with deleting files in GitHub!

Questions?

Step-by-Step Instruction

Step 1: Create a GitHub Account

- https://github.com/
- Free educational account with Emory email address
- Be careful with your username as it is visible to the public

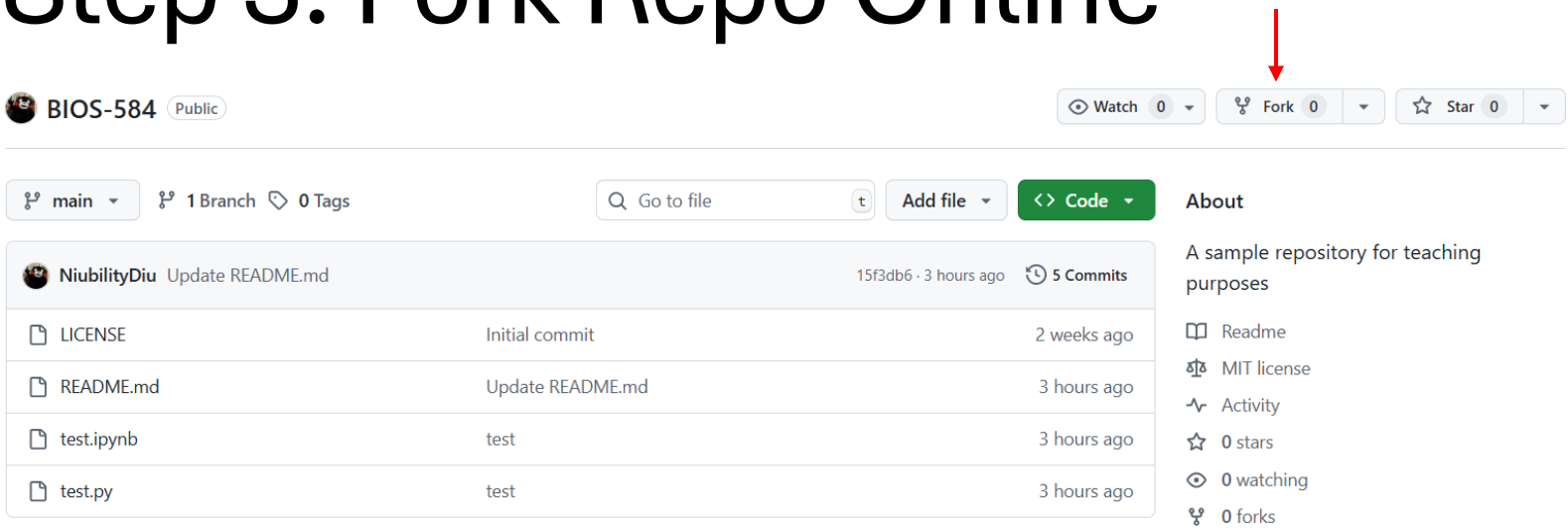


This is my profile page.

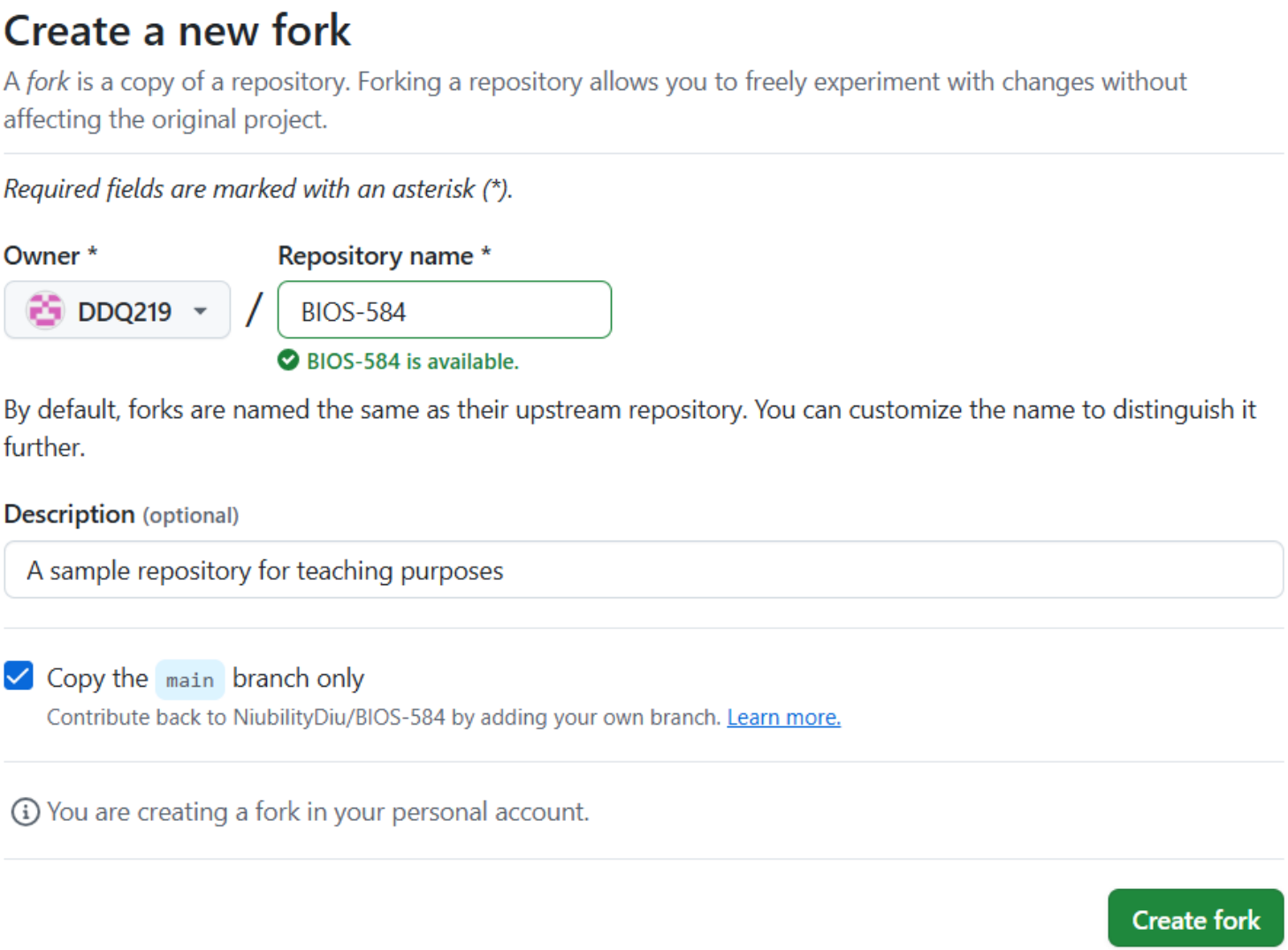
Step 2: Download and Install GitHub Desktop

- https://desktop.github.com/
- Helpful graphic user interface (GUI) for Git
- Available for Windows and macOS.

Step 3: Fork Repo Online

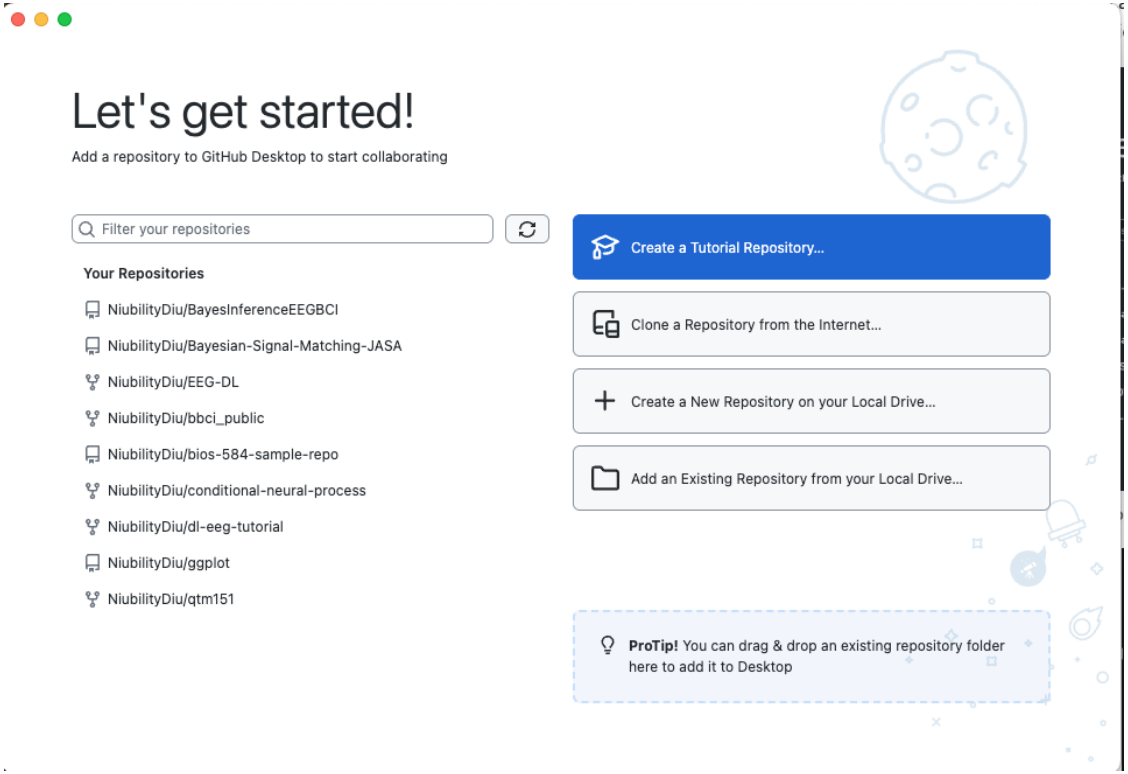


Step 4: Create a new Fork



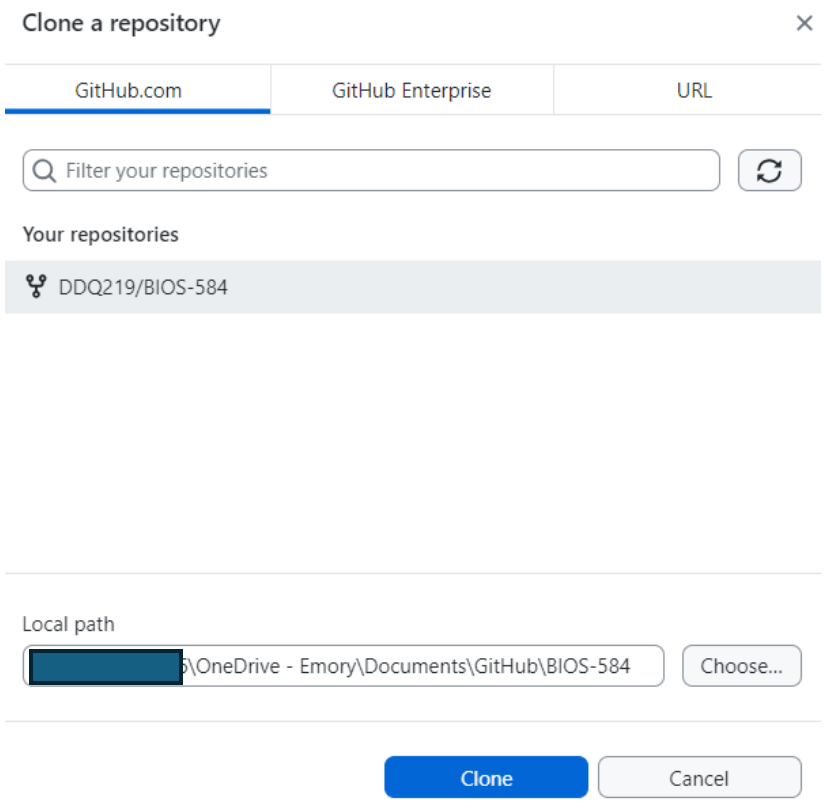
Step 5: Clone Repo to Desktop

- Click “Clone a Repository from the Internet” and select the one you forked.

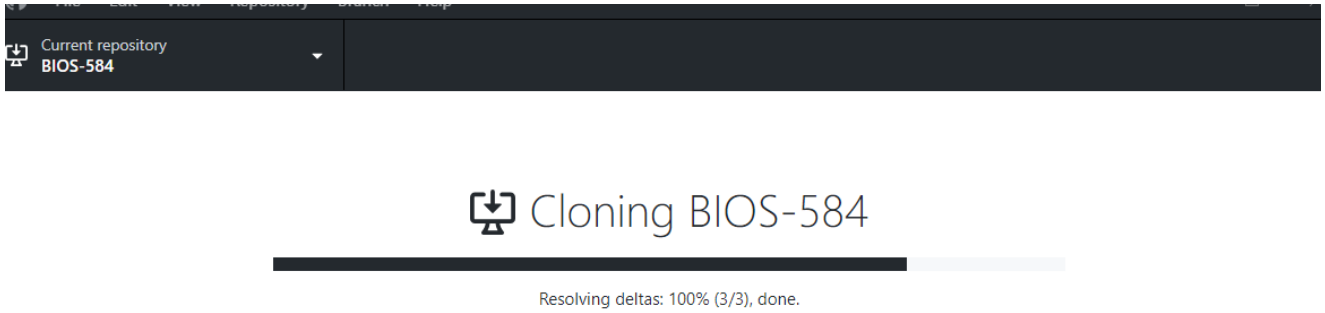


Step 5: Clone Repo to Desktop

- The forked one will appear here.
  - Note that the repo directory is under your name instead of mine!
- Refresh if necessary.

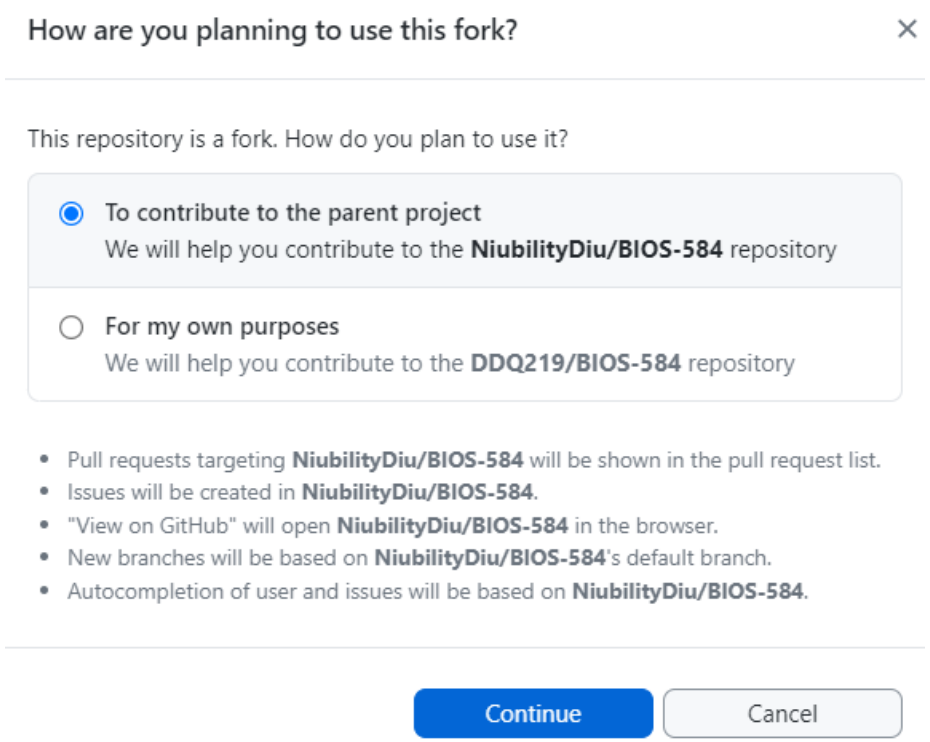


Step 5: Clone Repo to Desktop



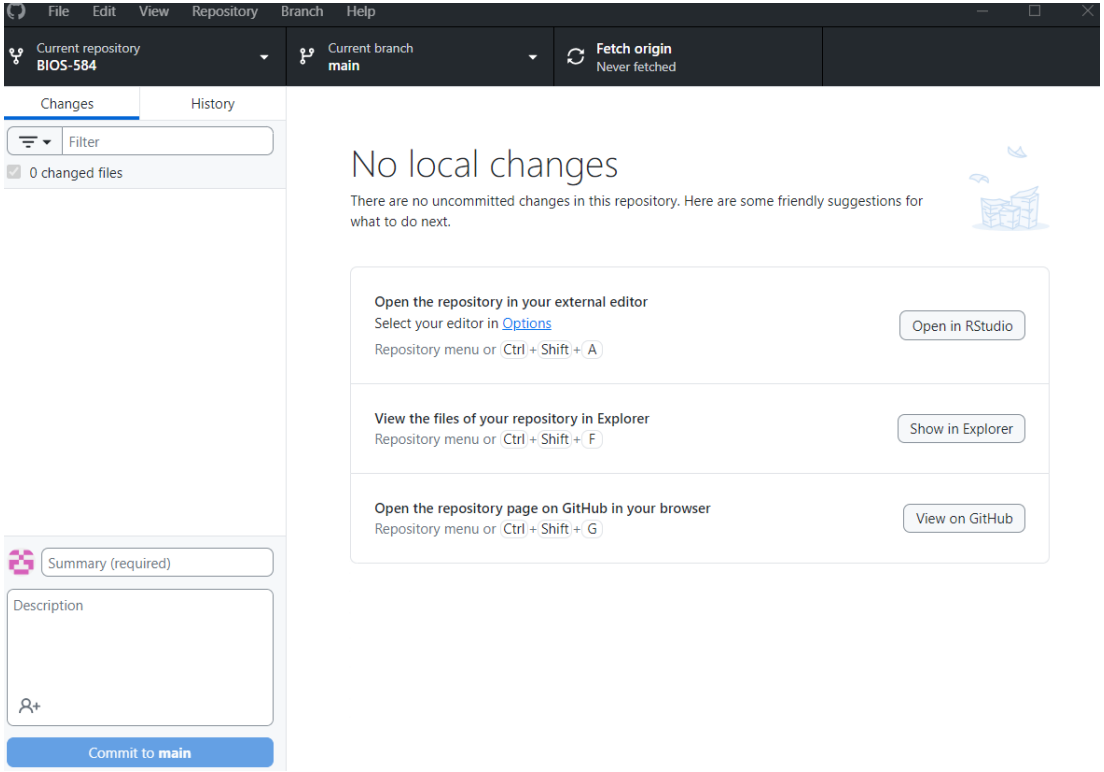
Step 5: Clone Repo to Desktop

- Click “contribute to the parent project”



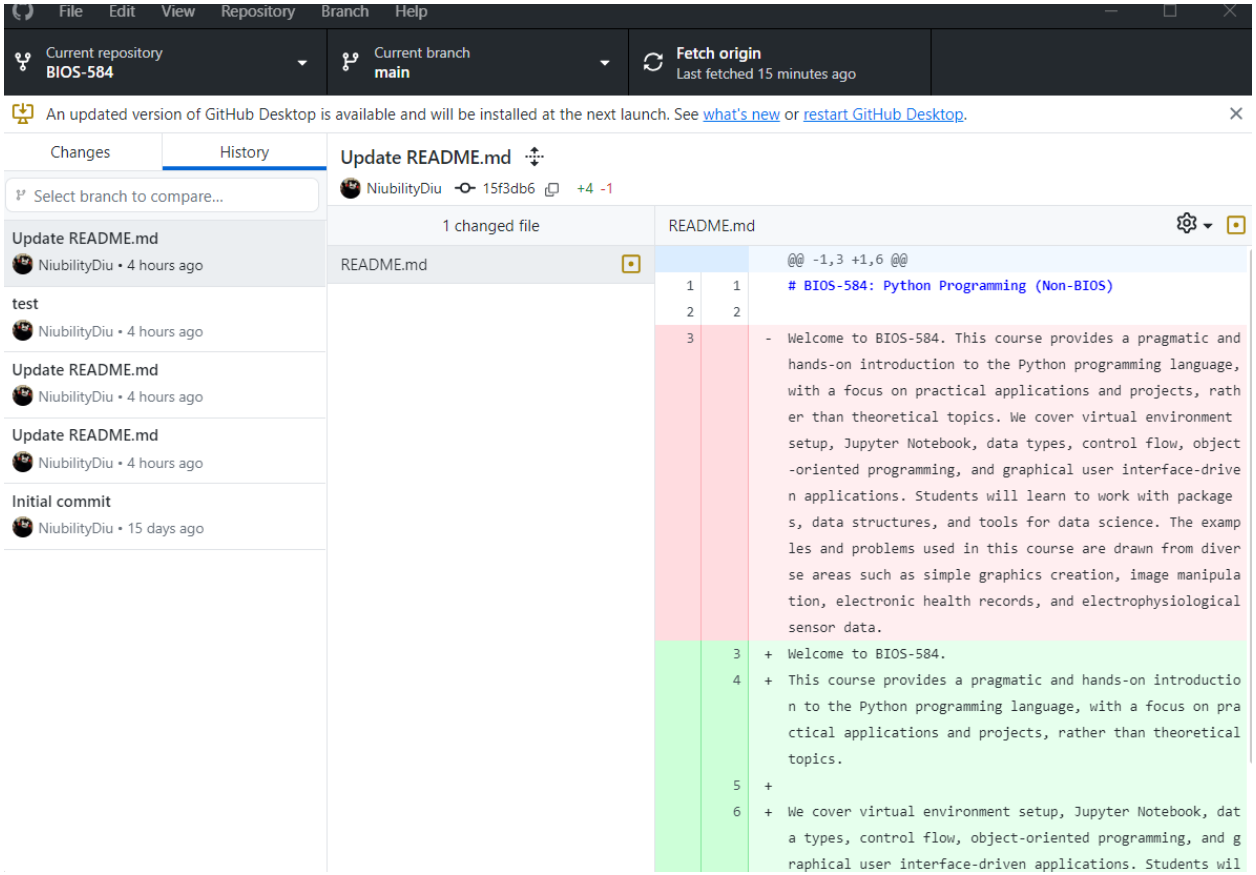
Step 5: Clone Repo to Desktop

- You should achieve the following status.



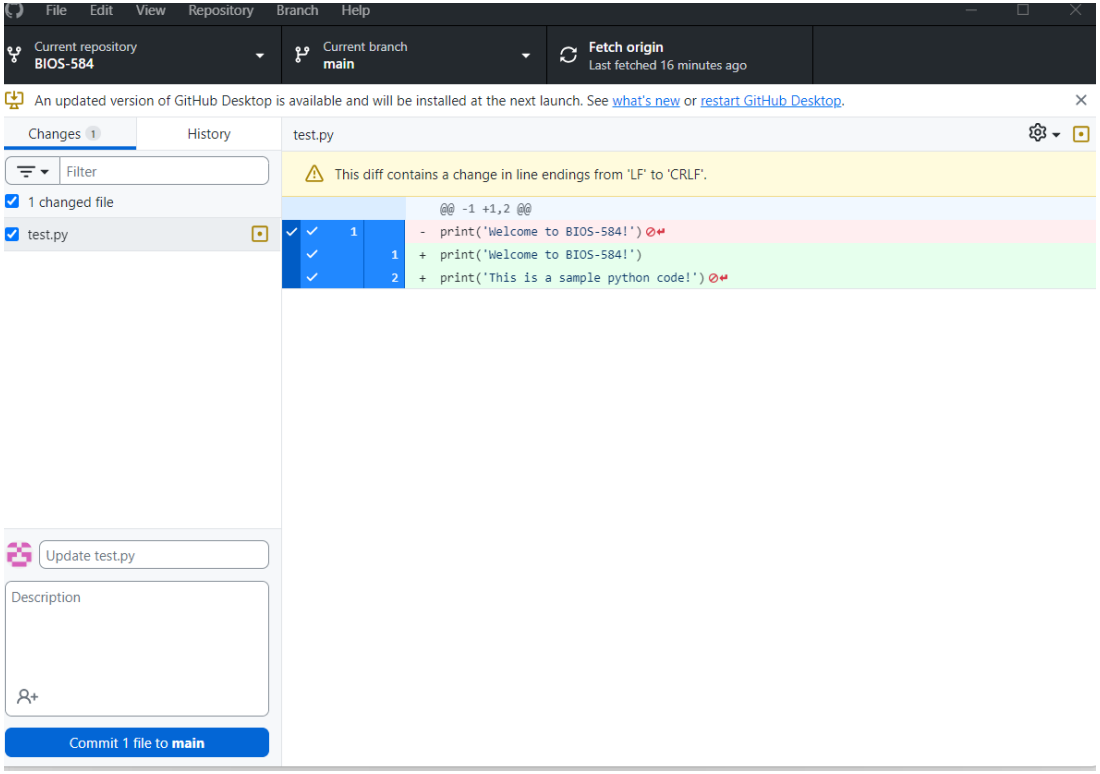
Step 6: Add Changes

- You will see History of my work.



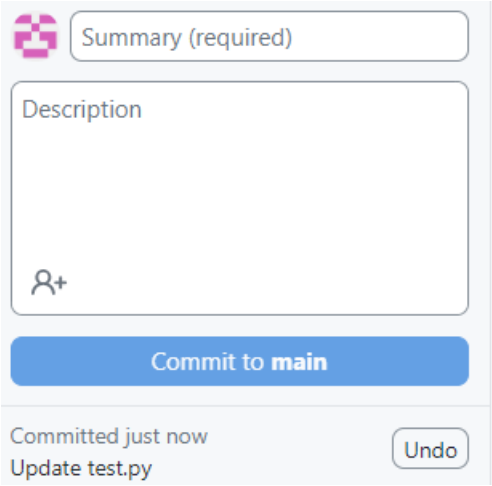
Step 6: Add Changes

- We create a “test.py” under the GitHub folder.
- GitHub desktop detects the change.

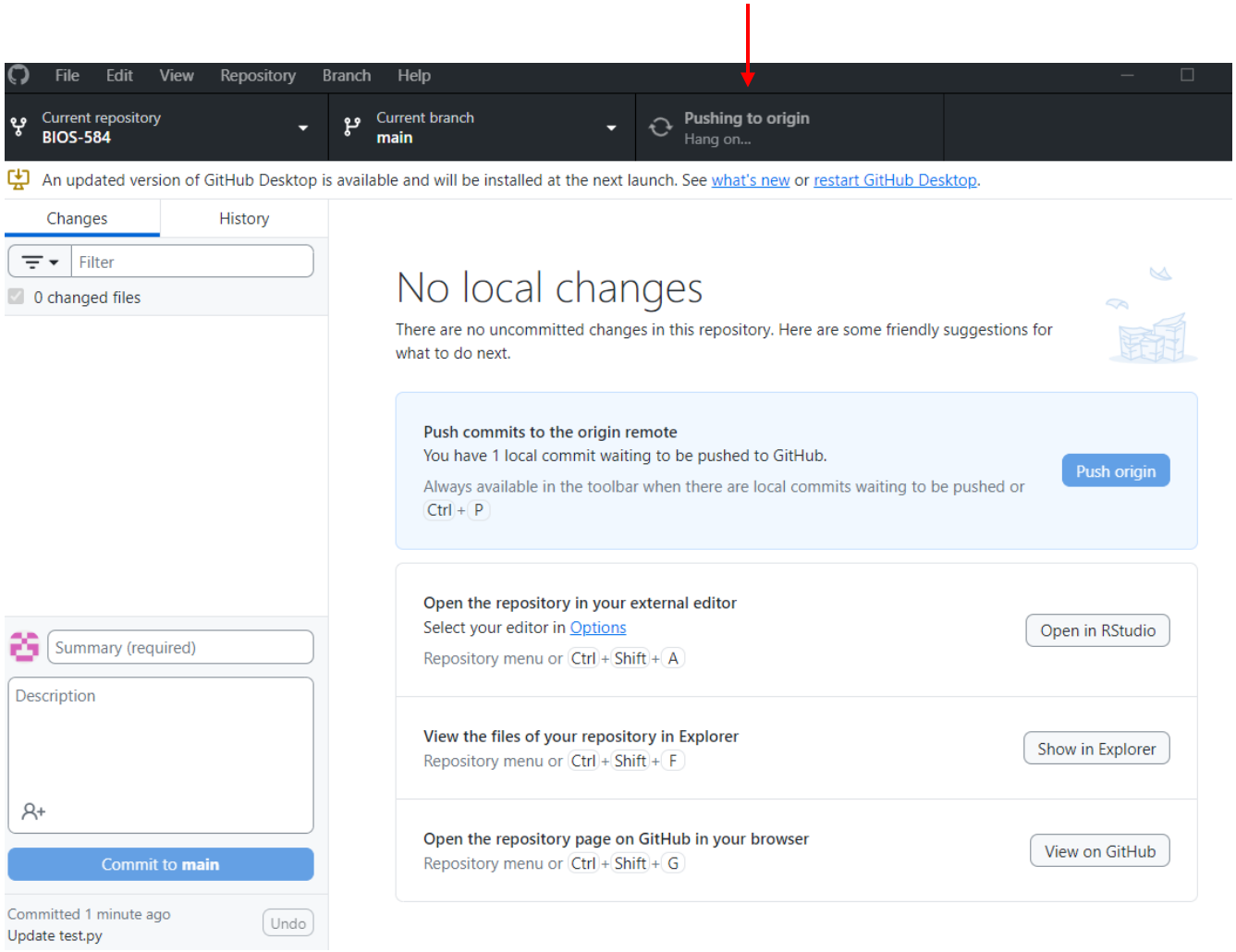


Step 7: Commit changes

- We commit this change.
- It is recommended to write a brief description for future tracking purposes.



Step 8: Push Changes



Step 8: Push Changes

- Your changes will be reflected in History.
- You should be able to see the new files on your GitHub repository online as well.