

Github

Aims

- To learn how to use GitHub for version control and collaboration.
- To practice debugging code with an IDE.
- To practice using classes, objects, properties, attributes and methods.

Clone repository

1. If you haven't already, please download GitHub Desktop from the following link: <https://desktop.github.com/>.
2. Sign in to your GitHub account by following the instructions [here](#).
3. Clone the `GitHub_Practice` repository. To do this, first select the option to “Clone a repository from the Internet...” in the GitHub Desktop client (this option is shown circled in Figure 1).

Let's get started!

Add a repository to GitHub Desktop to start collaborating

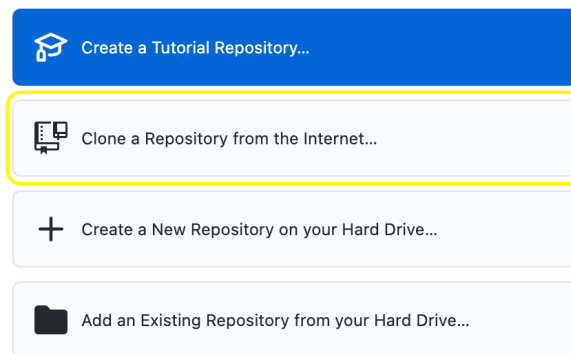


Figure 1: Click “Clone a repository from the Internet...”

4. A popup should appear titled “Clone a Repository”. Enter “AmeliaKh/GitHub.Practice” into the prompt, as shown in Figure 2.

Please note: your Local Path should be different to mine as it specifies where on your computer this project will be saved. You can change this path if you wish. Please **remember what you set this to.**

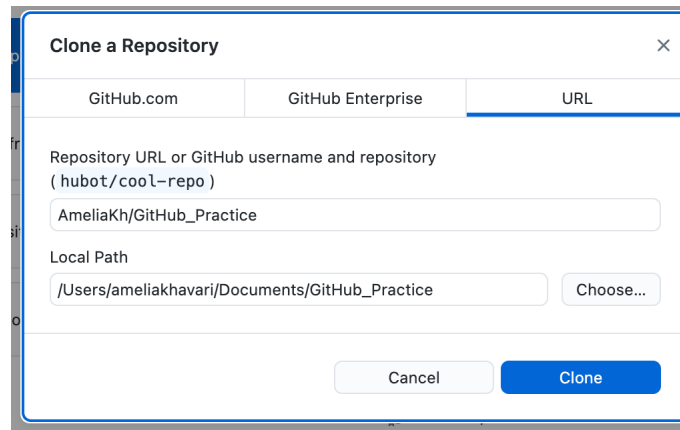


Figure 2: Enter "AmeliaKh/GitHub_Practice"

5. Click the blue **Clone** button in the bottom left hand corner.
6. You should now be able to see the `GitHub_Practice` repository in the place you 'cloned' at the location you chose in step 4.

Code Overview

`GitHub_Practice` contains 2 files: `main.py` and `test_main.py`. The former holds some code which needs to be fixed. You have each been assigned one or two functions to fix here. `test_main.py` contains tests for the methods in `main.py` which will pass if you correctly fix the bugs.

What to Do

1. Create a file called `[your name].md` and enter some text into it (enter whatever you like).
2. Go back to the GitHub Desktop app and you should see the new file in the list of uncommitted changes on the left (as shown in Figure 3).

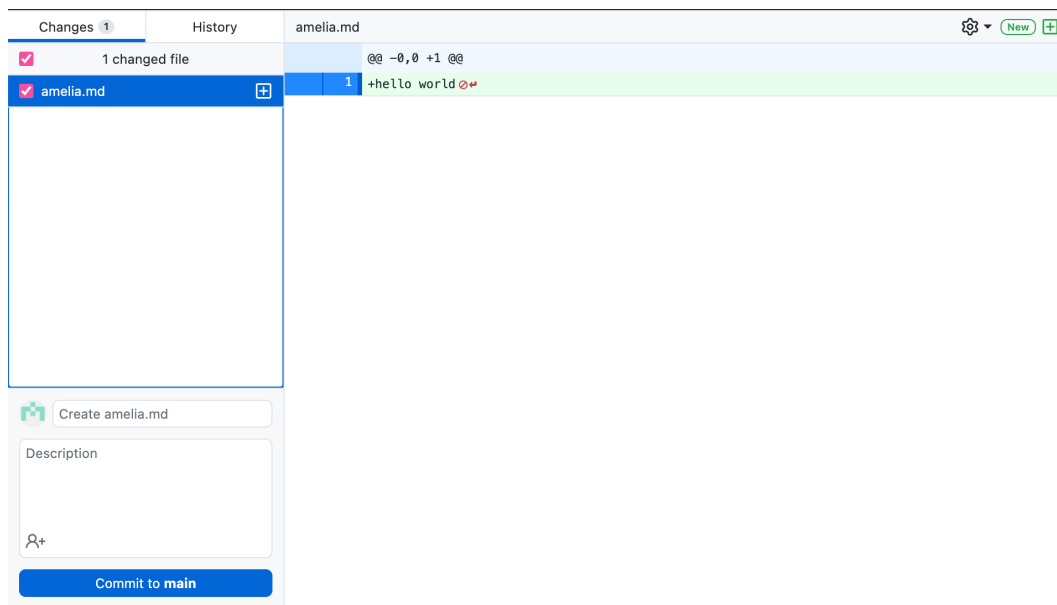


Figure 3: Uncommitted changes

When there's a green + next to a filename, it means the file is new. A yellow/orange circle means the file is an existing file which has been modified.

3. In the lower-left corner of the GitHub Desktop app (where it says “Create amelia.md” and “Description”), type a commit message, and then click **Commit to main**.
4. Click **Push origin** to push the local changes to the remote repository.
5. Complete the tasks you have been assigned in `main.py` so that all corresponding tests in `test_main.py` pass.

Hint: look at the tests if you're not sure what you need to do

6. Go back to the GitHub Desktop app and you should see `main.py` in the list of uncommitted changes on the left with a yellow/orange circle to its right.
7. In the right side of the screen, you should see the changes you made.
8. Once you are happy with the changes, enter a commit message (as in step 3), click **Commit to main** and **Push origin**.
9. Two of you should get a warning message saying “Newer Commits on Remote” (shown in Figure 4). This is because you are all trying to edit the same file at the same time and GitHub needs you to confirm which changes you want to keep. To solve this problem, first click **Fetch**. Then click **Pull origin**. This will ‘pull’ the changes that your classmate made.

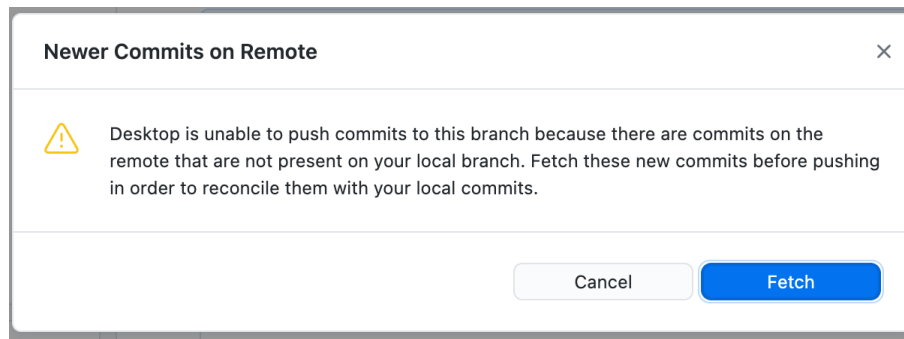


Figure 4: “Newer Commits on Remote” warning

10. Now, a popup will appear saying “Resolve conflicts before Merge”. Close the popup by clicking the x in the upper-right corner.
11. Reopen `main.py` in PyCharm and we will go through how to resolve this conflict together.