

GitHub for Beginners: A Step-by-Step Guide

Welcome, students! This tutorial will explore GitHub, a powerful version control and collaboration platform. By the end of this session, you will have a basic understanding of how to set up a repository, make changes, and collaborate with others.

1. What is GitHub?

GitHub is a platform to store projects (typically code) and track their changes over time. It uses a system called Git for version control, allowing multiple people to work on a project without interfering with each other.

2. Setting Up Your GitHub Account

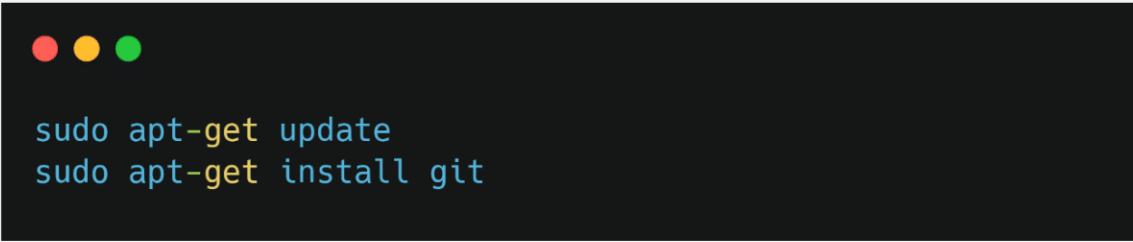
Before you can start using GitHub, you need an account:

1. Go to GitHub (<https://github.com/>).
2. Click on "Sign Up" and follow the prompts.

3. Installing Git

To interact with GitHub from your computer, you need to install Git:

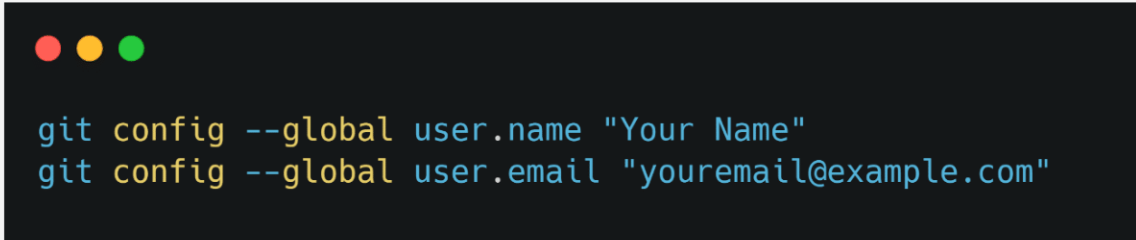
- For *Windows*: Download from [Git for Windows](<https://gitforwindows.org/>).
- For *Mac*: Use `brew install git` if you have Homebrew, or download from Git's website (<https://git-scm.com/>).
- For *Linux*: Use your package manager. For example, on Ubuntu:



```
sudo apt-get update  
sudo apt-get install git
```

4. Setting Up Git

Once installed, open your terminal or command prompt and configure Git with your name and email:



```
git config --global user.name "Your Name"  
git config --global user.email "youremail@example.com"
```

5. Creating a New Repository

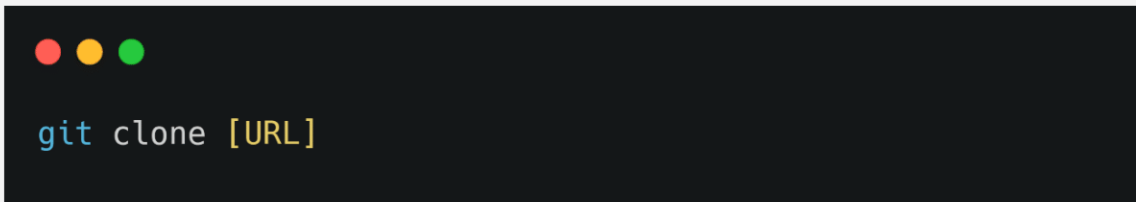
A repository (or "repo") is where your project lives.

1. On GitHub, click the green 'New' button.
2. Name your repository, add a brief description, and choose to initialize it with a README (a file explaining your project).
3. Click 'Create repository'.

6. Cloning Your Repository

To work on your project locally (on your computer), you need to 'clone' it:

1. click the green 'Code' button on your repo page and copy the URL.
2. Open your terminal and type:



```
git clone [URL]
```

Replace [URL] with the URL you copied.

7. Making Changes

Navigate to your cloned repository using:



```
cd [repo-name]
```

Make changes to your project. Once done, you can track these changes:

1. View changes with: `git status`
2. Add changes for commit: `git add .`
3. Commit the changes: `git commit -m "Brief description of changes"`

8. Pushing Changes

To send your changes to GitHub:



```
git push origin main
```

(Note: Some repositories use `master` instead of `main`. Adjust as necessary.)

9. Pulling Updates

If others are working on the project or if you made changes on GitHub's website, you can pull those changes to your local machine:



```
git pull origin main
```

10. Collaboration

1. *Forking*: If you want to work on someone else's project, you can 'fork' it. This creates a copy under your GitHub account. After making changes, you can return them to the original project using a 'pull request'.
2. *Pull Requests (PR)*: After making changes in a fork or branch, you can ask the main repository to accept your changes through a PR.
3. *Issues*: If you find problems or have suggestions for a project, you can open an 'issue' to discuss with the project maintainers.

Final Tip: Remember, practice makes perfect. Start by creating a personal project, making regular commits, and collaborating with classmates. The more you use Git and GitHub, the more comfortable you'll become. Happy coding!

Prof. Tiago Cunha Reis, PhD

<https://www.linkedin.com/in/tiagocunhareis/>