

GitHub Tutorials - Step by Step Guide

This tutorial is designed to help beginners learn GitHub step by step. GitHub is a platform for version control and collaboration, allowing developers to work together on projects efficiently. Follow these instructions to set up and use GitHub effectively.

1. Create a GitHub Account

- 1 Go to <https://github.com/>
- 2 Click on 'Sign Up'.
- 3 Fill in your details: username, email, and password.
- 4 Verify your email address to activate the account.

2. Install Git on Your Computer

- 1 Download Git from <https://git-scm.com/>
- 2 Follow the installation instructions based on your operating system (Windows, macOS, Linux).
- 3 Verify installation by running ``git --version`` in your terminal or command prompt.

3. Configure Git

- 1 Set your username: ``git config --global user.name "Your Name"``
- 2 Set your email: ``git config --global user.email "your_email@example.com"``
- 3 Check configuration using ``git config --list``.

4. Create a New Repository

- 1 Log in to GitHub.
- 2 Click on the '+' icon in the top-right corner and select 'New Repository'.
- 3 Enter repository name and description.
- 4 Choose visibility (Public or Private).
- 5 Click 'Create Repository'.

5. Clone a Repository

- 1 Copy the repository URL from GitHub.
- 2 Open terminal or command prompt.
- 3 Run: ``git clone ``
- 4 Navigate into the cloned folder using ``cd ``.

6. Make Changes and Commit

- 1 Open the project in your code editor.
- 2 Make necessary changes or add new files.
- 3 Stage changes: ``git add .``
- 4 Commit changes: ``git commit -m "Your commit message"``

7. Push Changes to GitHub

- 1 Ensure you are in the repository folder.
- 2 Run: ``git push origin main`` (replace 'main' with your branch name if different).
- 3 Your changes will now be available on GitHub.

8. Pull Changes from GitHub

- 1 To sync your local repository with GitHub, run: ``git pull origin main``.
- 2 This fetches and merges changes from the remote repository.

9. Create a Branch

- 1 Run: ``git branch `` to create a branch.
- 2 Switch to the branch: ``git checkout ``
- 3 You can also use: ``git checkout -b `` to create and switch at once.

10. Merge Branches

- 1 Switch to the main branch: ``git checkout main``.
- 2 Merge your feature branch: ``git merge ``.
- 3 Resolve conflicts if they appear.

Congratulations! ■ You have successfully learned the basics of GitHub. With these skills, you can now manage projects, collaborate with others, and contribute to open-source communities.