Set up Git

In this article

At the heart of GitHub is an open-source version control system (VCS) called Git. Git is responsible for everything GitHub-related that happens locally on your computer.

Using Git 🔗

To use Git on the command line, you will need to download, install, and configure Git on your computer. You can also install GitHub CLI to use GitHub from the command line. For more information, see "About GitHub CLI."

If you want to work with Git locally, but do not want to use the command line, you can download and install the GitHub Desktop client. For more information, see "About GitHub Desktop."

If you do not need to work with files locally, GitHub Enterprise Cloud lets you complete many Git-related actions directly in the browser, including:

- Creating a repository
- Forking a repository
- Managing files
- Being social

Setting up Git 🔗

1. Download and install the latest version of Git.

   > **Note**: If you are using a Chrome OS device, additional setup is required:
   >
   > a. Install a terminal emulator such as Termux from the Google Play Store on your Chrome OS device.
   >
   > b. From the terminal emulator that you installed, install Git. For example, in Termux, enter `apt install git` and then type `y` when prompted.

2. Set your username in Git.

3. Set your commit email address in Git.

Authenticating with GitHub from Git 🔗

When you connect to a GitHub repository from Git, you will need to authenticate with GitHub Enterprise Cloud using either HTTPS or SSH.

> **Note:** You can authenticate to GitHub Enterprise Cloud using GitHub CLI, for either HTTP or SSH. For more information, see `gh auth login` .

Connecting over HTTPS (recommended) 🔗

If you clone with HTTPS, you can cache your GitHub credentials in Git using a credential helper. For more information, see "[About remote repositories](#)" and "[Caching your GitHub credentials in Git](#)."

Connecting over SSH 🔗

If you clone with SSH, you must generate SSH keys on each computer you use to push or pull from GitHub Enterprise Cloud. For more information, see "[About remote repositories](#)" and "[Generating a new SSH key and adding it to the ssh-agent](#)."

Next steps 🔗

You now have Git and GitHub all set up. You may now choose to create a repository where you can put your projects. Saving your code in a repository allows you to back up your code and share it around the world.

- Creating a repository for your project allows you to store code in GitHub. This provides a backup of your work that you can choose to share with other developers. For more information, see "[Create a repository](#).".

- Forking a repository will allow you to make changes to another repository without affecting the original. For more information, see "[Fork a repo](#)."

- Each repository on GitHub is owned by a person or an organization. You can interact with the people, repositories, and organizations by connecting and following them on GitHub Enterprise Cloud. For more information, see "[Be social](#)."

- GitHub has a great support community where you can ask for help and talk to people from around the world. Join the conversation on [GitHub Community](#).