# Getting started with GitHub Desktop

**In this article**

Learn how to set up, authenticate, and configure GitHub Desktop to allow you to contribute to projects directly from your machine.

Mac    Windows

## Introduction 🔗

GitHub Desktop is a free, open source application that helps you to work with code hosted on GitHub or other Git hosting services. With GitHub Desktop, you can perform Git commands, such as committing and pushing changes, in a graphical user interface, rather than using the command line. For more information, see "About GitHub Desktop."

This guide will help you get started with GitHub Desktop by setting up the application, authenticating your account, configuring basic settings, and introducing the fundamentals of managing projects with GitHub Desktop. You will be able to use GitHub Desktop to collaborate on projects and connect to remote repositories after working through this guide.

You might find it helpful to have a basic understanding of Git and GitHub before getting started with GitHub Desktop. For more information, see the following articles.

- "Using Git"
- "Learning about GitHub"
- "Get started with GitHub documentation"

GitHub Desktop is an open source project. You can see the roadmap, contribute to the project, or open an issue to provide feedback or feature requests. For more information, see the `desktop/desktop` repository.

## Part 1: Installing and authenticating 🔗

You can install GitHub Desktop on any supported operating system. For more information, see "Supported operating systems for GitHub Desktop."

To install GitHub Desktop, visit the download page for GitHub Desktop. For more information, see "Installing GitHub Desktop."

After you have installed GitHub Desktop, you can authenticate the application with your account on GitHub or GitHub Enterprise. Authenticating allows you to connect to remote repositories on GitHub or GitHub Enterprise.

1. Before you can authenticate to GitHub or GitHub Enterprise, you will need an account. For more information, see "[Signing up for a new GitHub account](#)."

2. In the menu bar, select **GitHub Desktop**, then click **Preferences**.
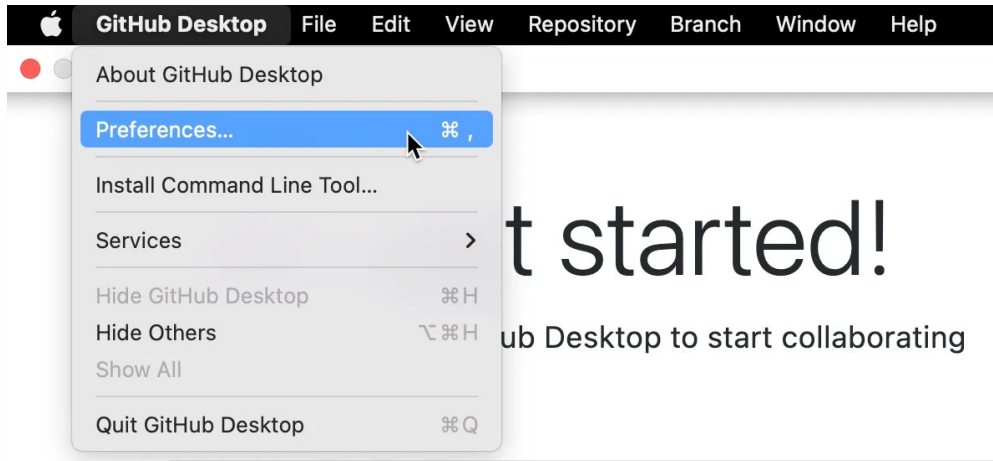


3. In the "Preferences" window, on the **Accounts** pane, click one of the **Sign In** buttons to sign in to your account on GitHub.com or GitHub Enterprise.
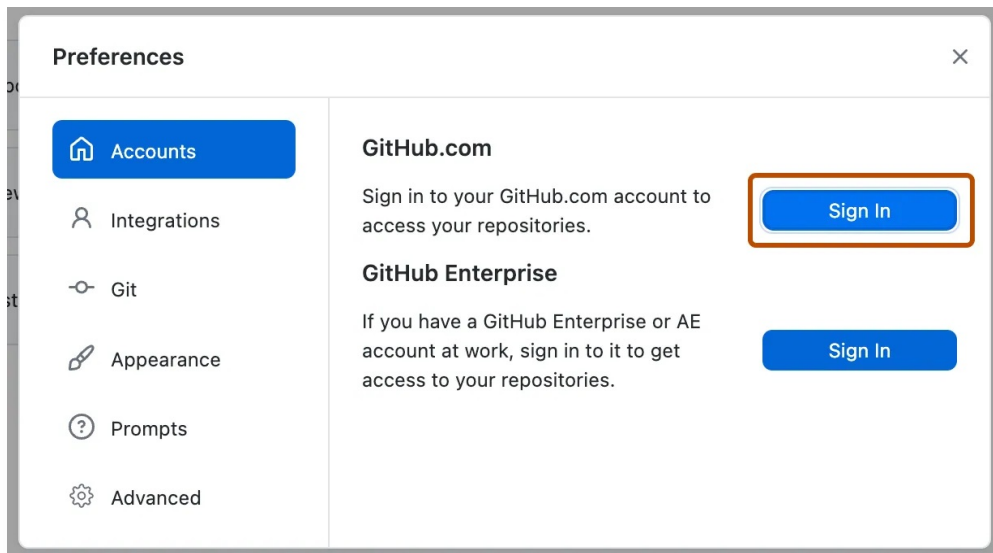


4. Follow the steps to sign in. For more information about authentication, see "[Authenticating to GitHub in GitHub Desktop](#)."

1. Before you can authenticate to GitHub or GitHub Enterprise, you will need an account. For more information, see "[Signing up for a new GitHub account](#)."

2. In the File drop-down menu, click **Options**. In the options window, click **Accounts** and follow the steps to sign in. For more information on authenticating, see "[Authenticating to GitHub in GitHub Desktop](#)."
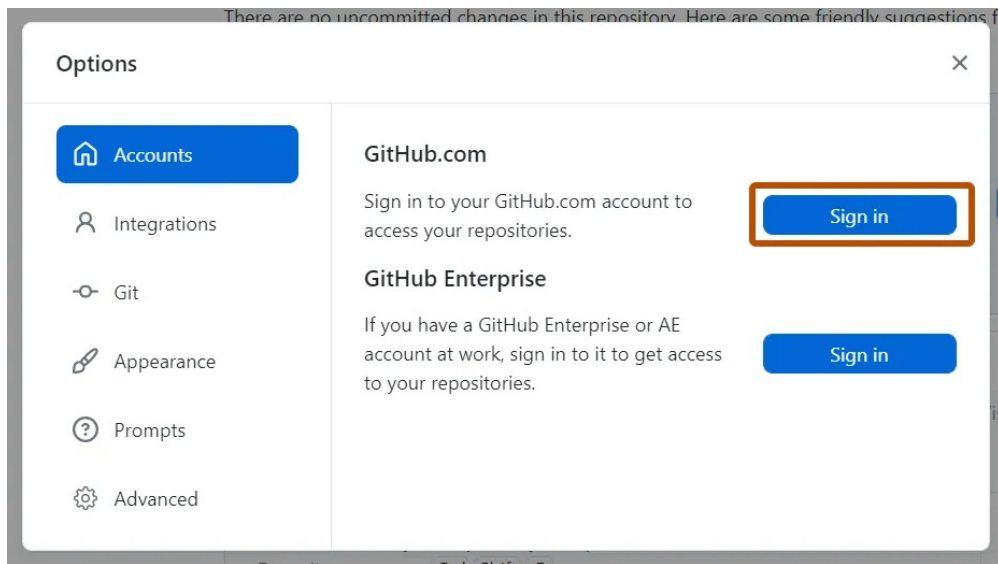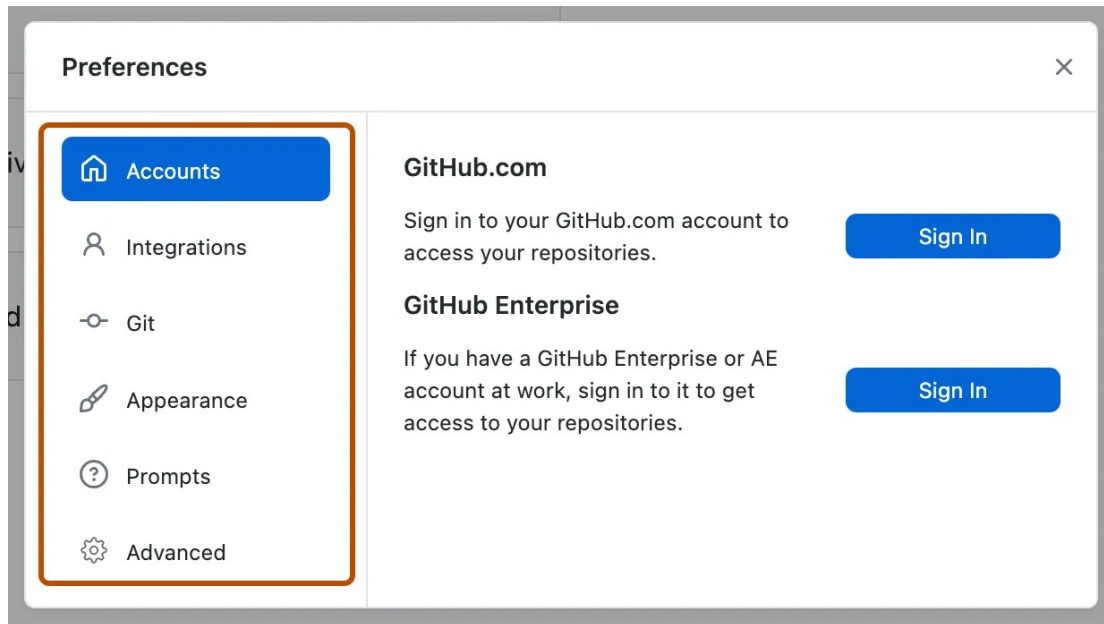
# Part 2: Configuring and customizing GitHub Desktop

🔗

After you install GitHub Desktop, you can configure and customize the app to best suit your needs.

You can connect or remove accounts on GitHub or GitHub Enterprise, choose a default text editor or shell, edit your Git configuration, change the appearance of GitHub Desktop, customize system dialog boxes, and set privacy preferences in the GitHub Desktop Preferences window. For more information, see "Configuring basic settings in GitHub Desktop."
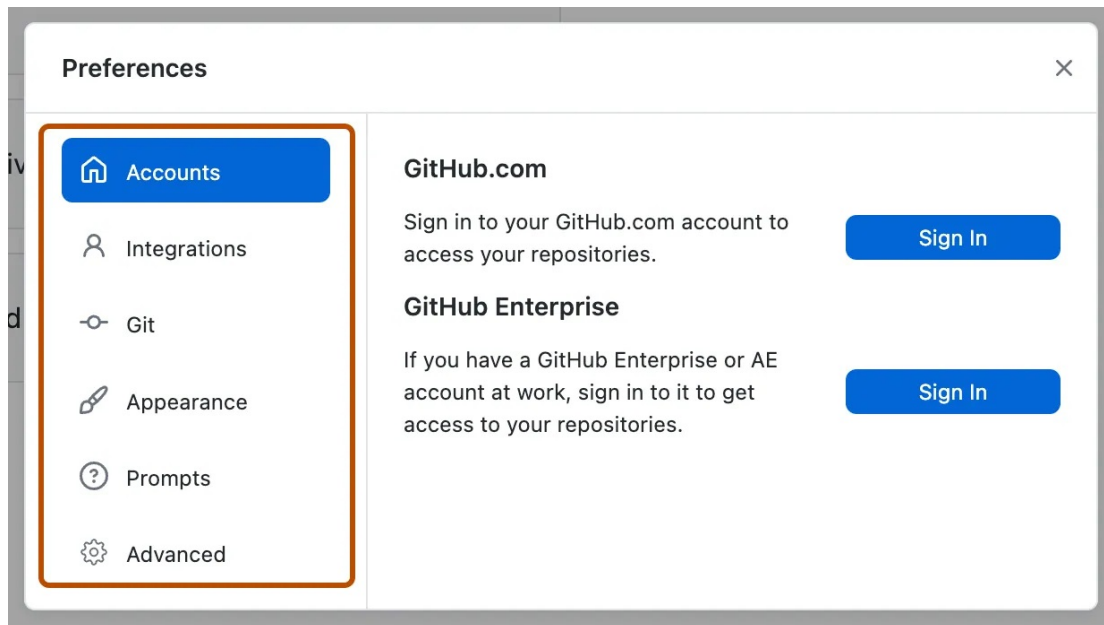


You can connect or remove accounts on GitHub or GitHub Enterprise, choose a default text editor or shell, edit your Git configuration, change the appearance of GitHub Desktop, customize system dialog boxes, and set privacy preferences in the GitHub Desktop Options window. For more information, see "Configuring basic settings in GitHub Desktop."

# Part 3: Contributing to projects with GitHub Desktop 🔗

---

After installing, authenticating, and configuring the app, you are ready to start using GitHub Desktop. You can create, add, or clone repositories and use GitHub Desktop to manage contributions to your repositories.

## Creating, adding, and cloning repositories 🔗

You can create a new repository by selecting **File** in the "GitHub Desktop" menu bar and clicking **New repository...**. For more information, see "[Creating your first repository using GitHub Desktop](#)."

You can add a repository from your local computer by selecting **File** and clicking **Add Local Repository...**. For more information, see "[Adding a repository from your local computer to GitHub Desktop](#)."

You can clone a repository from GitHub by selecting **File** and clicking **Clone Repository...**. For more information, see "[Cloning and forking repositories from GitHub Desktop](#)."
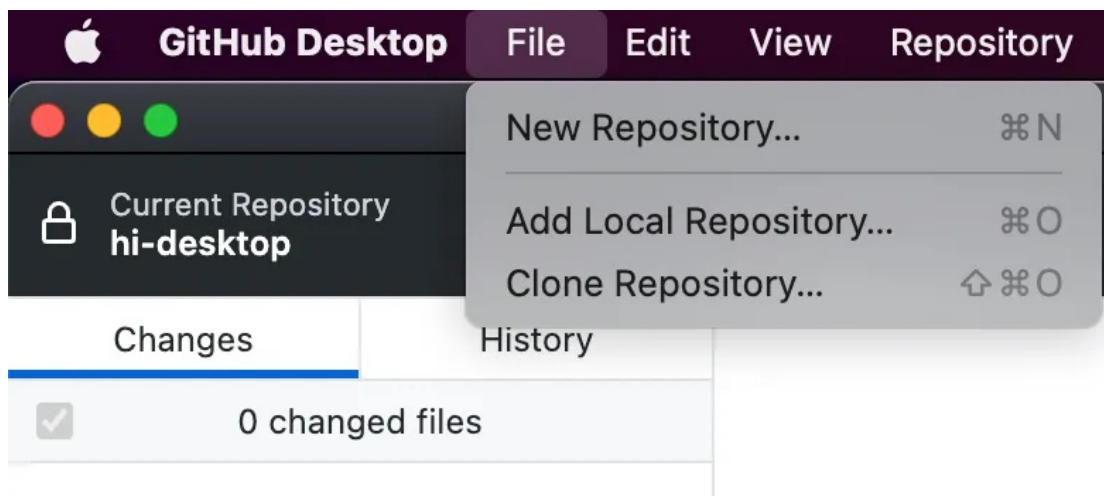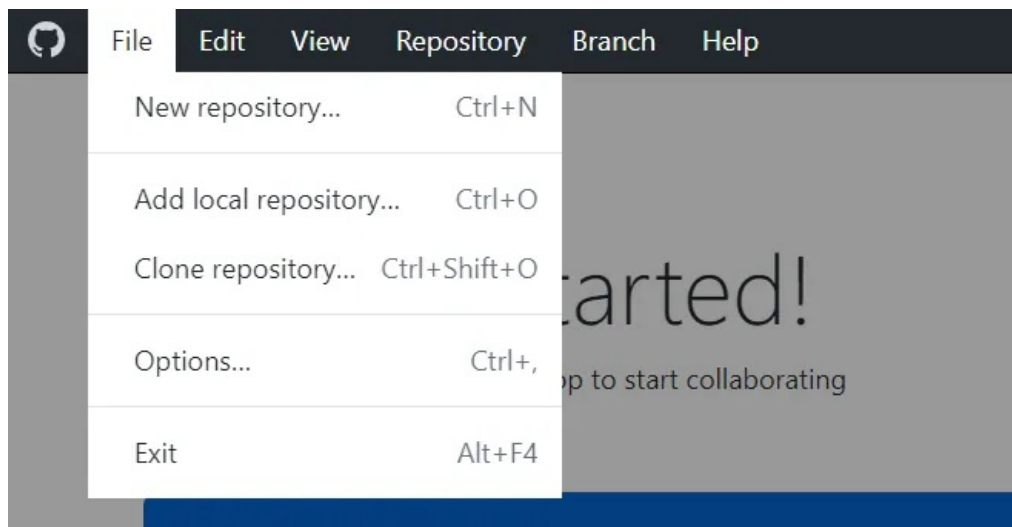
## Making changes in a branch 🔗

You can use GitHub Desktop to create a branch of a project. Branches isolate your development work from other branches in the repository, so that you can safely experiment with changes. For more information, see "[Managing branches in GitHub Desktop](#)."

After you make changes to a branch, you can review them in GitHub Desktop and make a commit to keep track of your changes. For more information, see "[Committing and reviewing changes to your project in GitHub Desktop](#)."

If you want to access your changes remotely or share them with other people, you can push your commits to GitHub. For more information, see "[Pushing changes to GitHub from GitHub Desktop](#)."

## Collaborating with GitHub Desktop 🔗

You can use GitHub Desktop to create issues or pull requests to collaborate on projects with other people. Issues help you keep track of ideas and discuss possible changes to projects. Pull requests let you share your proposed changes with others, receive feedback, and merge changes into a project. For more information, see "[Creating an issue or pull request from GitHub Desktop](#)."'

You can view your own or your collaborator's pull requests in GitHub Desktop. Viewing a pull request in GitHub Desktop lets you see any proposed changes and make additional changes by opening the project's files and repositories in your default text editor. For more information, see "[Viewing a pull request in GitHub Desktop](#)."

## Keeping your local repository in sync 🔗

When you make changes to your local repositories or when other people make changes to the remote repositories, you will need to sync your local copy of the project with the remote repository. GitHub Desktop can keep your local copy of a project in sync with the remote version by pushing and pulling commits. For more information, see "[Syncing your branch in GitHub Desktop](#)."