

# Adding a file to a repository

## In this article

Adding a file to a repository on GitHub Enterprise Server

Adding a file to a repository using the command line

You can upload and commit an existing file to a repository on GitHub Enterprise Server or by using the command line.

[Mac](#) [Windows](#) [Linux](#)

## Adding a file to a repository on GitHub Enterprise Server

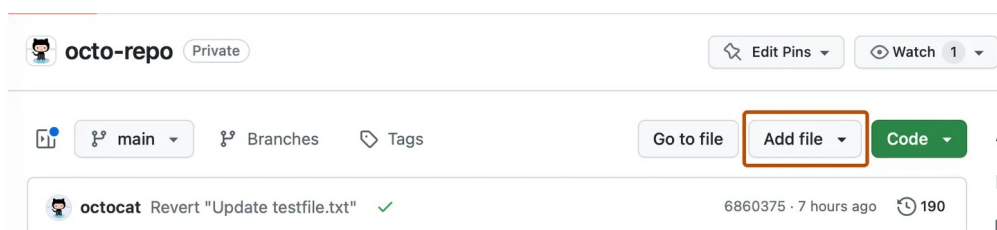
Files that you add to a repository via a browser are limited to 25 MiB per file. You can add larger files, up to 100 MiB each, via the command line. For more information, see "[Adding a file to a repository using the command line](#)." To add files larger than 100 MiB, you must use Git Large File Storage. For more information, see "[About large files on GitHub](#)."

### Tips:

- You can upload multiple files to GitHub Enterprise Server at the same time.
- If a repository has any protected branches, you can't edit or upload files in the protected branch using GitHub. For more information, see "[About protected branches](#)."

You can use GitHub Desktop to move your changes to a new branch and commit them. For more information, see "[Committing and reviewing changes to your project in GitHub Desktop](#)."

- 1 On your GitHub Enterprise Server instance, navigate to the main page of the repository.
- 2 Above the list of files, select the **Add file** dropdown menu and click **Upload files**. Alternatively, you can drag and drop files into your browser.



- 3 To select the files you want to upload, drag and drop the file or folder, or click **choose your files**.
- 4 In the "Commit message" field, type a short, meaningful commit message that describes the change you made to the file. You can attribute the commit to more than one author in the commit message. For more information, see "[Creating a](#)

[commit with multiple authors.](#)"

- 5 Below the commit message fields, decide whether to add your commit to the current branch or to a new branch. If your current branch is the default branch, you should choose to create a new branch for your commit and then create a pull request. For more information, see "[Creating a pull request.](#)"

☐ Commit directly to the `main` branch.

☒ Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

`octocat-patch-1`

**Propose changes** **Cancel**

- 6 Click **Propose changes**.

## Adding a file to a repository using the command line

You can upload an existing file to a repository on your GitHub Enterprise Server instance using the command line.

**Tip:** You can also [add an existing file to a repository from the GitHub Enterprise Server website.](#)

This procedure assumes you've already:

- [Created a repository on GitHub Enterprise Server](#), or have an existing repository owned by someone else you'd like to contribute to
- [Cloned the repository locally on your computer](#)

**Warning:** Never `git add`, `commit`, or `push` sensitive information to a remote repository. Sensitive information can include, but is not limited to:

- Passwords
- SSH keys
- [AWS access keys](#)
- API keys
- Credit card numbers
- PIN numbers

For more information, see "[Removing sensitive data from a repository.](#)"

- 1 On your computer, move the file you'd like to upload to GitHub Enterprise Server into the local directory that was created when you cloned the repository.
- 2 Open TerminalTerminalGit Bash.
- 3 Change the current working directory to your local repository.
- 4 Stage the file for commit to your local repository.

```
$ git add .  
# Adds the file to your local repository and stages it for commit. To  
unstage a file, use 'git reset HEAD YOUR-FILE'.
```

- 5 Commit the file that you've staged in your local repository.

```
$ git commit -m "Add existing file"
# Commits the tracked changes and prepares them to be pushed to a remote
repository. To remove this commit and modify the file, use 'git reset --soft
HEAD~1' and commit and add the file again.
```

- 6 [Push the changes](#) in your local repository to your GitHub Enterprise Server instance.

```
$ git push origin YOUR_BRANCH
# Pushes the changes in your local repository up to the remote repository
you specified as the origin
```

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)