Configuring Git Large File Storage

Once [Git LFS is installed](), you need to associate it with a large file in your repository.

Mac   Windows   Linux

If there are existing files in your repository that you'd like to use GitHub Enterprise Server with, you need to first remove them from the repository and then add them to Git LFS locally. For more information, see "[Moving a file in your repository to Git Large File Storage]()."

If there are referenced Git LFS files that did not upload successfully, you will receive an error message. For more information, see "[Resolving Git Large File Storage upload failures]()."

> **Note:** Before trying to push a large file to GitHub Enterprise Server, make sure that you've enabled Git LFS on your enterprise. For more information, see "[Configuring Git Large File Storage for your enterprise]()."

① Open TerminalTerminalGit Bash.

② Change your current working directory to an existing repository you'd like to use with Git LFS.

③ To associate a file type in your repository with Git LFS, enter `git lfs track` followed by the name of the file extension you want to automatically upload to Git LFS.

For example, to associate a *.psd* file, enter the following command:

```
$ git lfs track "*.psd"
> Adding path *.psd
```

Every file type you want to associate with Git LFS will need to be added with `git lfs track`. This command amends your repository's *.gitattributes* file and associates large files with Git LFS.

> **Note:** We strongly suggest that you commit your local *.gitattributes* file into your repository.
>
> - Relying on a global *.gitattributes* file associated with Git LFS may cause conflicts when contributing to other Git projects.
> - Including the *.gitattributes* file in the repository allows people creating forks or fresh clones to more easily collaborate using Git LFS.
> - Including the *.gitattributes* file in the repository allows Git LFS objects to optionally be included in ZIP file and tarball archives.

④ Add a file to the repository matching the extension you've associated:

```
git add path/to/file.psd
```

5. Commit the file and push it to GitHub Enterprise Server:

```
git commit -m "add file.psd"
git push
```

You should see some diagnostic information about your file upload:

```
> Sending file.psd
> 44.74 MB / 81.04 MB  55.21 % 14s
> 64.74 MB / 81.04 MB  79.21 % 3s
```

Further reading 🔗

- "[Collaboration with Git Large File Storage](#)"