

Git Tutorial

backlog + **git**
by nulab

Backlog lets you manage projects with Git integration

Learn more →

Syncing repositories

Remote repositories allow us to share our changes with other members of the team. They can be on a private server, on a different computer than yours, or hosted using a service like Backlog. Wherever yours is hosted, you'll need to be able to sync your local repository with the remote repository frequently. You'll do this using three commands: `git push`, `git pull`, and `git merge`.

Git push

In order to start sharing changes with others, you have to push them to a remote repository using the "push" command. This will cause the remote repository to update and synchronize with your local repository.



Push your local changes to a remote repository.

Git pull

Git Tutorial

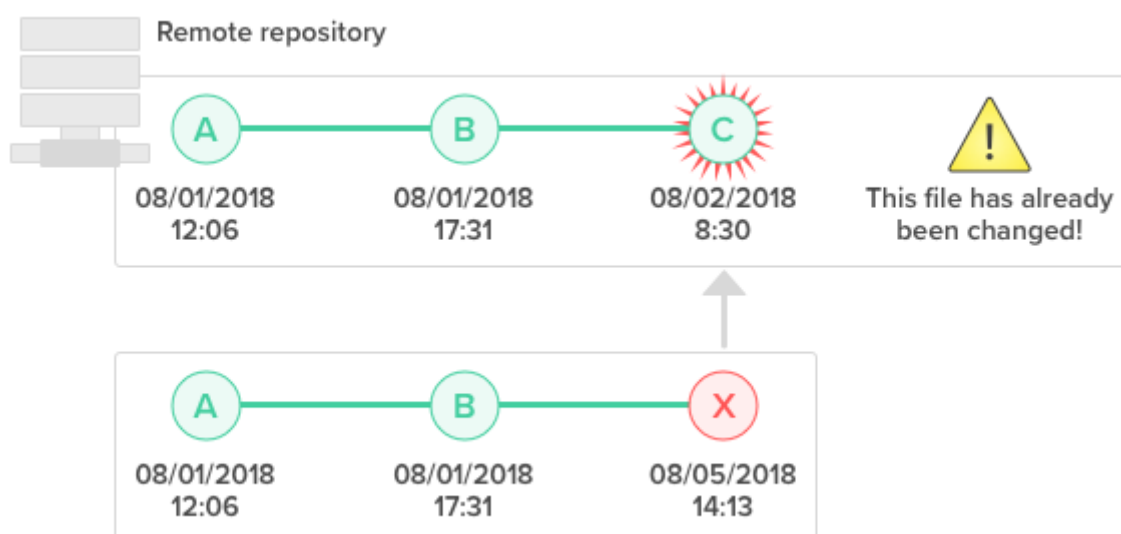
When the pull is executed, the latest revision history will download from the remote repository and import to your local repository.



Pull changes from a remote repository to your local repository.

Git merge

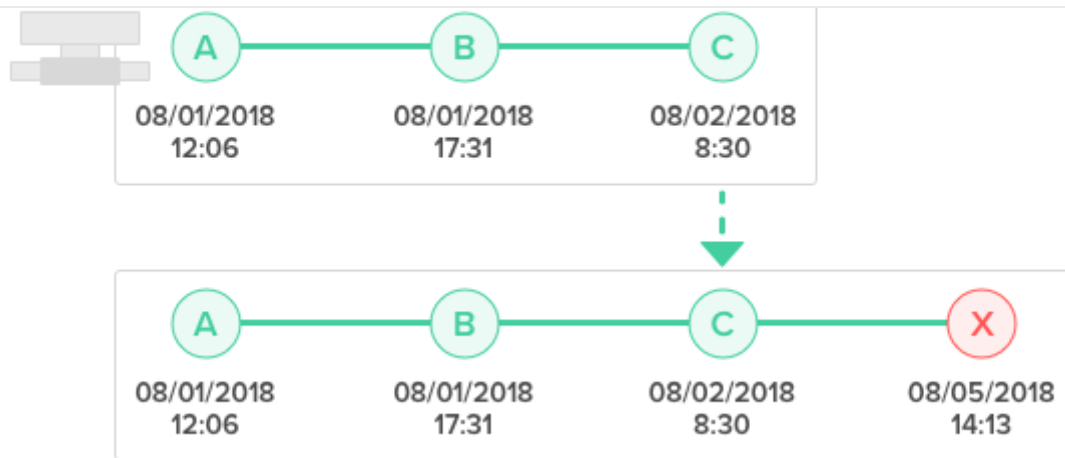
Your push to the remote repository will be rejected if your local repository is out of date, possibly because there are some updates on the remote repository that you do not have locally yet.



You are unable to push to the remote repository if your local repo is out of date.

If that is the case, you'll have to use the git merge command to grab the latest change from the remote repository before you are allowed to push. Git

Git Tutorial



You must merge the latest changes before pushing.

During a "merge", Git will attempt to automatically apply those history changes and merge them with the current branch. However, if there is a conflict in changes, Git will throw an error prompting you to resolve the conflict manually.

Resolve merge conflicts

When merging two branches, you may come across a conflict that needs resolving before you can properly complete the merge. For example, when two or more members make changes on the same part of a file in the two different branches (remote and local branches in this case), Git will not be able to automatically merge them.

When this happens, Git will add some standard conflict-resolution markers to the conflicting file. The markers help you figure out which sections of the file need to be manually resolved.

Git Tutorial

```
10 >>>>>>>>>> 17c860612953c0f9d88f313c8dfbf7d858e02e91
11 <<<<<<<<<<<
12 </html>
```

Example of a conflict occurrence.

In our example, everything above "=====" is your local content, and everything below it comes from the remote branch.

You must resolve the conflicting parts as shown below before you can proceed with creating a merge commit.

```
1 <html>
2   <head>
3     <title>hello</title>
4   </head>
5   <body>
6     <strong>Hello</strong>
7   </body>
8 </html>
```

Revising the conflict.

Revise the commit to eliminate the conflict.

—

..

Backlog has built-in Git repositories for every project

Try it free

Follow us on



English

Terms

Privacy

© 2021 Nulab, Inc. All rights reserved.