



[Home](#) » [Tutorials](#) » [Git](#) » Git – Resolve Merge Conflicts

Git – Resolve Merge Conflicts

Many time, when we do git push/pull or git merge, we end up with conflicts.

In most cases, solution to merge-conflict is as simple as discarding local changes or remote/other branch changes.

Following is useful in those cases...

Resolving merge conflicts

Find files with merge conflict

Change working directory to project folder.

Let's stay in touch! 

Email Address

Subscribe

```
cd project-folder
```

Search for all conflicting files.

```
grep -lr '<<<<<<' .
```

Above will list all files which has marker special marker <<<<<< in them.

Resolve easy/obvious conflicts

At this point you may review each files. If solution is to accept local/our version, run:

```
git checkout --ours PATH/FILE
```

If solution is to accept remote/other-branch version, run:

```
git checkout --theirs PATH/FILE
```

If you have multiple files and you want to accept local/our version, run:

```
grep -lr '<<<<<<' . | xargs git checkout --ours
```

If you have multiple files and you want to accept remote/other-branch version, run:

```
grep -lr '<<<<<<' . | xargs git checkout --theirs
```

For complex conflicts

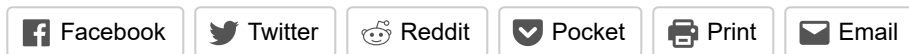
For files that needs manual review/edit, use vim or any text editor to resolve differences.

Make sure you run `git add FILENAME` for files edited using vim.

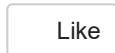
Finally, review if all files are ready for commit using `git status`

And run `git commit -am "MSG"` followed by optional `git push`

Share this:



Like this:



Be the first to like this.