**Remove untracked file from the working directory**

As per the Git Documentation [git clean](https://git-scm.com/docs/git-clean)

Remove untracked files from the working tree

Step 1 is to show what will be deleted by using the -n option:

git clean -n

Clean Step - **beware: this will delete files**:

git clean -f

* To remove directories, run git clean -f -d or git clean -fd
* To remove ignored files, run git clean -f -X or git clean -fX
* To remove ignored and non-ignored files, run git clean -f -x or git clean -fx

**Note** the case difference on the X for the two latter commands.

If clean.requireForce is set to "true" (the default) in your configuration, one needs to specify -fotherwise nothing will actually happen.

Again see the [git-clean](http://git-scm.com/docs/git-clean) docs for more information.

## Options

**-f**

**--force**

If the Git configuration variable clean.requireForce is not set to false, git clean will refuse to run unless given -f, -n or -i.

**-x**

Don’t use the standard ignore rules read from .gitignore (per directory) and $GIT\_DIR/info/exclude, but do still use the ignore rules given with -e options. This allows removing all untracked files, including build products. This can be used (possibly in conjunction with git reset) to create a pristine working directory to test a clean build.

**-X**

Remove only files ignored by Git. This may be useful to rebuild everything from scratch, but keep manually created files.

**-n**

**--dry-run**

Don’t actually remove anything, just show what would be done.

**-d**

Remove untracked directories in addition to untracked files. If an untracked directory is managed by a different Git repository, it is not removed by default. Use -f option twice if you really want to remove such a directory.

References:

1. <https://stackoverflow.com/questions/61212/how-to-remove-local-untracked-files-from-the-current-git-working-tree>