

1. Create a new repository on GitHub. To avoid errors, do not initialize the new repository with README, license, or gitignore files. You can add these files after your project has been pushed to GitHub.
2. Open Terminal.
3. Change the current working directory to your local project.
4. Initialize the local directory as a Git repository.

```
git init
```

5. Add the files in your new local repository. This stages them for the first commit.

```
git add .
```

Adds the files in the local repository and stages them for commit. To unstage a file, use `git reset HEAD YOUR-FILE`.

6. Commit the files that you've staged in your local repository.

```
git commit -m "First commit"
```

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.

7. Copy remote repository URL field. At the top of your GitHub repository's Quick Setup page, click, to copy the remote repository URL.
8. In Terminal, add the URL for the remote repository where your local repository will be pushed.

```
git remote add origin remote repository URL
# Sets the new remote
git remote -v
# Verifies the new remote URL
```

9. Push the changes in your local repository to GitHub.

```
git push origin master
```

Pushes the changes in your local repository up to the remote repository you specified as the origin

Ignore files:

première méthode

include: files directory *.extension in a **.gitignore** file to list all excluded files in sub directories.

The **.gitignore** file have to be committed to the git repository

Second method

To make a list that will not be included in remote repository use the **.git/info/exclude** file. Use the same syntax that for the **.gitignore** file.

List ignored file

to check for top level directory

```
`git status --ignored`
```

to check for all files that would be ignored: `git status --ignored --untracked-files=all`