1. [Create a new repository](https://help.github.com/articles/creating-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 Git Bash.
3. Change the current working directory to your local project.
4. Initialize the local directory as a Git repository.
5. git init
6. Add the files in your new local repository. This stages them for the first commit.
7. git add .
8. # Adds the files in the local repository and stages them for commit. To unstage a file, use 'git reset HEAD *YOUR-FILE*'.
9. Commit the files that you've staged in your local repository.
10. git commit -m "First commit"
11. # 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.
12. At the top of your GitHub repository's Quick Setup page, click  to copy the remote repository URL.
13. In the Command prompt, [add the URL for the remote repository](https://help.github.com/articles/adding-a-remote) where your local repository will be pushed.
14. git remote add origin *remote repository URL*
15. # Sets the new remote
16. git remote -v
17. # Verifies the new remote URL
18. [Push the changes](https://help.github.com/articles/pushing-to-a-remote) in your local repository to GitHub.
19. git push origin master

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

**or create a new repository on the command line**

echo "# secondary" >> README.md

git init

git add README.md

git commit -m "first commit"

git remote add origin https://github.com/devops165/secondary.git

git push -u origin master

**…or push an existing repository from the command line**

git remote add origin https://github.com/devops165/secondary.git

git push -u origin master