**Git command cheat sheet + step-by-step tutorials**

Download Git: <http://git-scm.com>This is a Git Command cheat sheet, designed to help pushing/pulling files to/from GitHub.Note: Do not include <>’s when writing commands, they are a marker to display areas you replace in your use case.

|  |  |
| --- | --- |
| Git Basics | |
| git init | Initialise your current directory as a local git repository.  Note: this creates a .git file that is hidden by default in your directory. |
| git clone <repo URL> | Retrieve/clone a GitHub repository to your local machine via URL. |
| git add <file>  git add . | Use <file> to stage specific file(s), add multiple files like so: git add <file> <file2> <file3>. Remember to include file extensions.  Use git add . to stage all files in current directory. |
| git status | Use git status to show which files have been created, changed, or deleted. Also shows if they need to be added (staged) with above command or committed with below command. |
| git commit -m “<descriptive message>” | Commit your staged file(s) as a new commit snapshot. The descriptive message can be anything useful regarding your changes. |
| git branch -M main | Creates the branch for the current commit. Here “main” is an semi-optional name. I have not included it inside <> to avoid confusion as “main” is very commonly used, alongside “master”, however, you could name it anything. |
| git remote add origin <repo URL> | Creates the connection between your local repository, and your remote repository (on GitHub). |
| git push -u origin main | Pushes your committed files to the defined origin. The “main” argument here defines the branch you want to push to, I.e., if you wanted to push to a different branch, you would swap “main” for the name of the branch e.g., “wip” (work in progress)  After this command, if all goes well, your files will be pushed to the GitHub repo. (You may need to authenticate yourself on GitHub in browser). |
| git log | Displays the commit history and unique commit IDs for use in the below command. |
| git checkout <unique commit ID> | Used to rollback/forward to the selected commit provided from the above command. |
| git branch | Displays a list of your current branches. A “\*” will be displayed beside your current branch. |
| git pull <repo URL> | Retrieves the specified remote repository copy of current branch and merges it into local copy. |

**Step-by-step for creating a new local repository on the command line and pushing local files to a remote repository for the first time:**

* Having set up your GitHub Repository
* Go Inside the directory you want to push files from (open directory using cd <directory>)

Then enter the following commands:

1. git init
   * git add README.md (entirely optional)
2. git commit -m “descriptive message”
3. git branch -M main
4. git remote add origin https://github.com/<username>/<github\_repo.git>
5. git push -u origin main

refresh your GitHub repo to see changes.

**Once you have completed this action in full in your command prompt, while still in that same command prompt some of these steps become redundant.**

* For example, if you want to push a new file/file change to your remote repo you only need to run the following steps:

1. git add .
2. git commit -m “descriptive message”
3. git push -u origin main

refresh your GitHub repo to see changes.

**Troubleshooting:**

A common error is finding that your updates were “rejected because the remote contains work you do not have locally…”

* git tells you the command you need to run “git pull …”
* replace the “…” with the url of the repo you want to pull
  + after pulling the remote repo, your local and remote repo should be identical
* re-try your push or command that caused the error

NOTE: Git has many more commands that provide a wide functionality, visit the Git website posted at the top of this cheat sheet for more info, or visit a third-party explanatory site.