Git Commands Cheat Sheet

git config --global user.name <name>
- add new Git user

git config --global user.email <email>
- link email address to Git user

git config --list
- view new user settings

cd C:\Local_Git_Repository
- change directory to local repo

mkdir <name of new directory>
- create a new directory for new local repo

git init
- initialize the local repo

git status
- check any change in the repository.

git add <filename>
- add the file to staging.

git add *
- '*' requests git to stage all files within current repo folder

git commit -m <comment/remark>
- commit the files in the staging to the local repository.

git diff
- to show the differences between old and new files
- green lines indicate that there are additions or lines that exist in the second file but not in the first original file.
- Redlines indicate the deletions or lines that exist in the first file but not in the second file

==========

git branch
- list the branches in a repository.

git branch <new name>

- create a new branch.

git checkout <branch to switch to>
- switch to a branch.

git merge <from which branch>
- merge from a branch to the current branch.

git remote add origin <url of GitHub repository>
- specify URL of remote GitHub repository for pushing

git remote set-url origin {URL}
- Update the remote origin to the URL of your new Github repository

git remote -v
- check if local Git repository has the remote repository server correctly configured

git push --set-upstream origin master
- setup the remote upstream branch and push the master branch
- only use it for the first push. Git remembers the relationship between the local branch and the corresponding branch on the remote repository.

git push -u origin
-push command for subsequent changes in master branch

git tag –a <tag> –m <comment>
- create a new Git tag

git push origin <tag>
- push newly created tag from local repo to remote repo

git submodule add <Git submodule repository URL> <name of submodule>
- add Git submodule to existing local repo

git clone <URL of repository to clone>
-clone another GitHub repository locally onto laptop/desktop

git log
-view and verify commit history