{Git & Github Cheat Sheet}

Sepideh K. Gharamaleki

Config, Create and Branch

- Git Config

- git config −−global user.email “youremail”
- Make vscode default instead of VIM:
  git config −−global core.editor "code −−−wait"

- SSH Config

- Click here.

- Create Repo:

- git status
- git init

- Commit:

- git add file
- git commit -m “message”/ git commit

- Log:

- git log −−oneline

- Fixing Commits (just one commit ago)

- git add forgotten_file
- git commit –amend

- Create and Switch to Branch

- git checkout -b branchname
- git switch -c branchname
- If switch before committing changes: doesn’t work if the file is shared between the branches. Refer to stashing changes.

- Manipulating Branches

- Delete: git branch -d or -D if branch is not commited
- Rename: Switch to branch first: git branch -m newbranchname
- Merge branches: We merge to HEAD. git merge branchname
- git branch -v : shows branchnames hash of commit last commit
- If conflicts: resolve then add and commit the changes.

Monitor changes

- Manipulating Commits

- Comparing changes:
  git diff (commits or files or . . .)
- Stashing: When you have uncommited changes on one branch and want to jump to another:
  git stash
- Getting back in time:
  git checkout hashofanoldcommit/HEAD∼1..n
  - Results in detached head To get rid of everything after the last commit: git checkout HEAD
- To get rid of everything after any commit: git reset/revert hashofcommit
- Revert keeps the commits that are deleted cause it makes a new commit.

- Rebase

- To merge commits and delete the history of the merged commits: git rebase master
- Merge conflicts are resolved with instructions git gives you.
- Rebase to rewrite commits:
  git rebase -i HEAD 4

Github

- git remote add origin url

- Push

- git push origin <branch>

- Fetch

- Accessing changes on repo without having to commit on local repo. => Fetch:
  git fetch origin branch
- to see the changes:
  git checkout origin/branchname
- git pull origin branch= git fetch+ git merge
- wherever you run git pull, the branch will merge there. Before pushing, pull to see if anyone made changes
- To resolve conflicts when you pull request:
  - git fetch origin
  - git switch my-new-feature
  - git merge master
  - fix conflicts!
  - git switch master
  - git merge my-new-feature
  - git push origin master

Misc.

- Tagging

- List all tags: git tag
- Switch to a tag: git checkout tagname
- Difference bet. Tags: git diff tagname1 tagname2
- Create tag: git tag tagname/ git tag -a tagname
- Earlier commits: git tag tagname commithash
- You have to push tags: git push −−tags

- Reflogging

- displays a log of all the local reference updates made in your repo.
- git reflog show Head
- Reflogs only local logs and expire after 90 days.
- To retrieve previous commits you can use reflog:git reset –hard master@{1}
- You can undo rebase with reflog.

- Aliasing

- Alter /.gitconfig file.
- Can be done on terminal too.
- git config −−global alias.nameofalias nameof function