GIT & GITHUB
LESSON 7

NANODEGREE → FOUNDATION MODULE

AGENDA

GIT

- Git is the most commonly used version control system.

- Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to.

- Git also makes collaboration easier, allowing changes by multiple people to all be merged into one source.
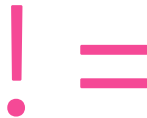
Master
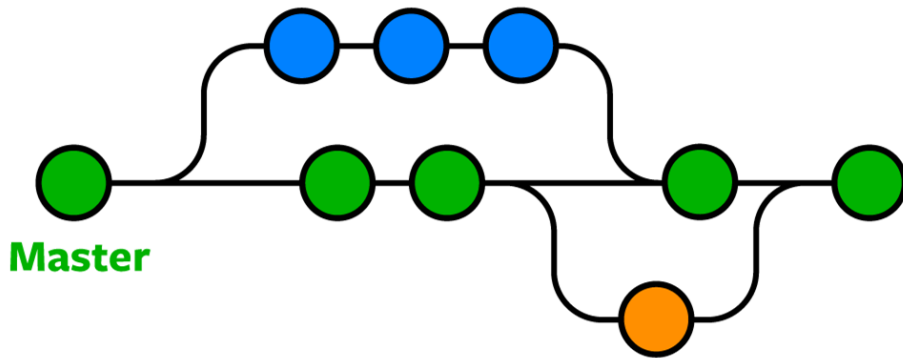
GIT & GitHub

GIT

!=

GITHUB

"version control system that lets you manage and keep track of your source code history"

"cloud-based hosting service that lets you manage Git repositories"

GITHUB

CLOUD BASED REPOSITORY

- GitHub is a for-profit company that offers a cloud-based Git repository hosting service.

- Without GitHub, using Git generally requires a bit more technical savvy and use of the command line.

- Additionally, anyone can sign up and host a public code repository for free

GIT

Core Concepts
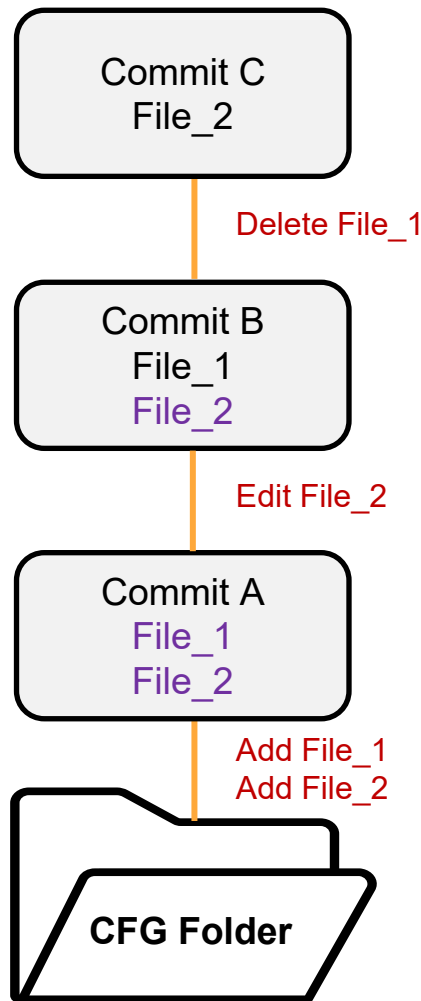
git init – creates a new Git repository

git add – adds your files to a queue such that they will be *committed later*

git commit – creates a "snapshot" of your repository at a specific time.
- commits the *added files.*
- *everything at this point is still local (on your machine).*

git push – sends the committed changes to the remote repository.

git init → **CFG Folder**

Commit C
File_2

Delete File_1

Commit B
File_1
File_2

Edit File_2

Commit A
File_1
File_2

Add File_1
Add File_2

CFG Folder

GIT

Core Concepts

Working Tree

| File_1 |
| File_2 |

folder and the corresponding subfolders on your computer associated with a git repository

Staging Area (Index)

| File_1 |
| File_2 |

a place to store files before commit

History

| Commit C |
| Commit B |
| Commit A |
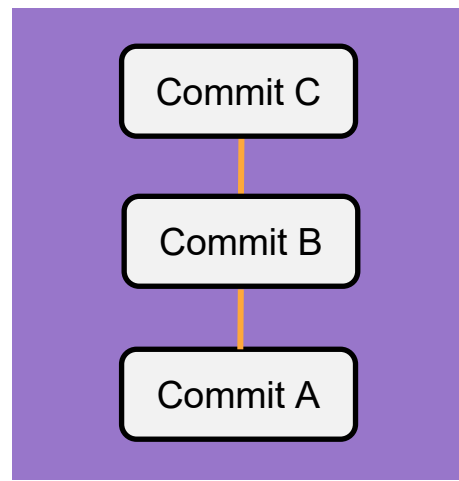
It is a hidden directory that saves all our commits

working directory -> git add -> staging area -> git commit -> repository

GITHUB EXAMPLES
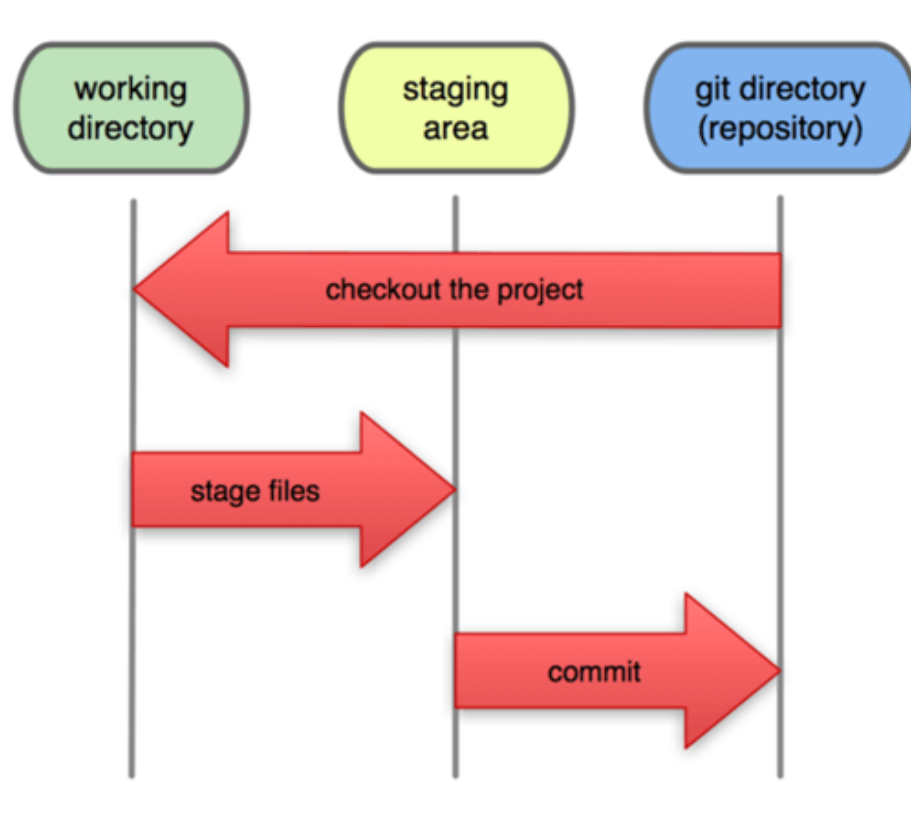
⧖ KEY STATES OF GIT VERSIONED LIFE CYCLE

CHECKOUT KEY RESOURCES

https://git-scm.com/doc

https://git-scm.com/book/en/v2

GIT

Branches

Your work

Master (Main)

Your colleague's work
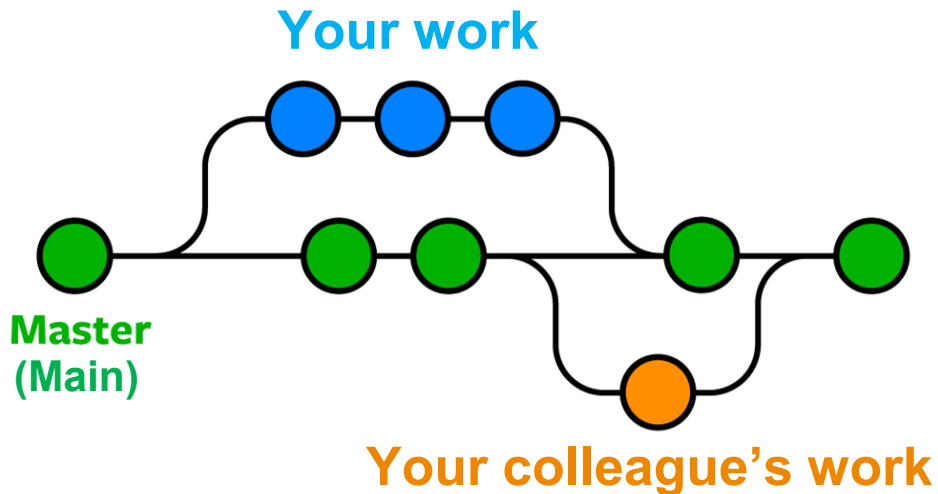
- Use branches from **Main** to experiment, make edits, additions and changes.
- Once a branch gets all the necessary approvals, then it will be merged in the **Main** branch.
- To create **Pull Requests** we need to have a separate branch.

Pull Requests "let you tell others about changes you've pushed to a branch in a repository on GitHub. Once a pull request is opened, you can discuss and review the potential changes with collaborators and add follow-up commits before your changes are merged into the base branch."

GITHUB EXAMPLES

VISUALIZING GIT CONCEPTS

git status – displays the state of the working directory and the staging area
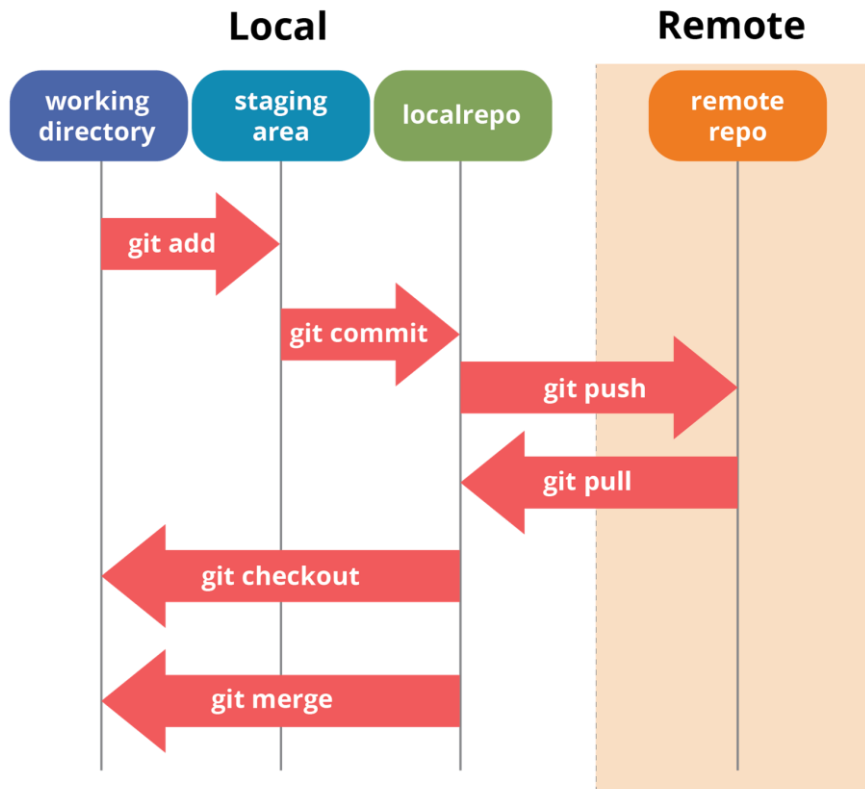
git pull – **git fetch** + **git merge**

git fetch – downloads all your changes from the remote to your local
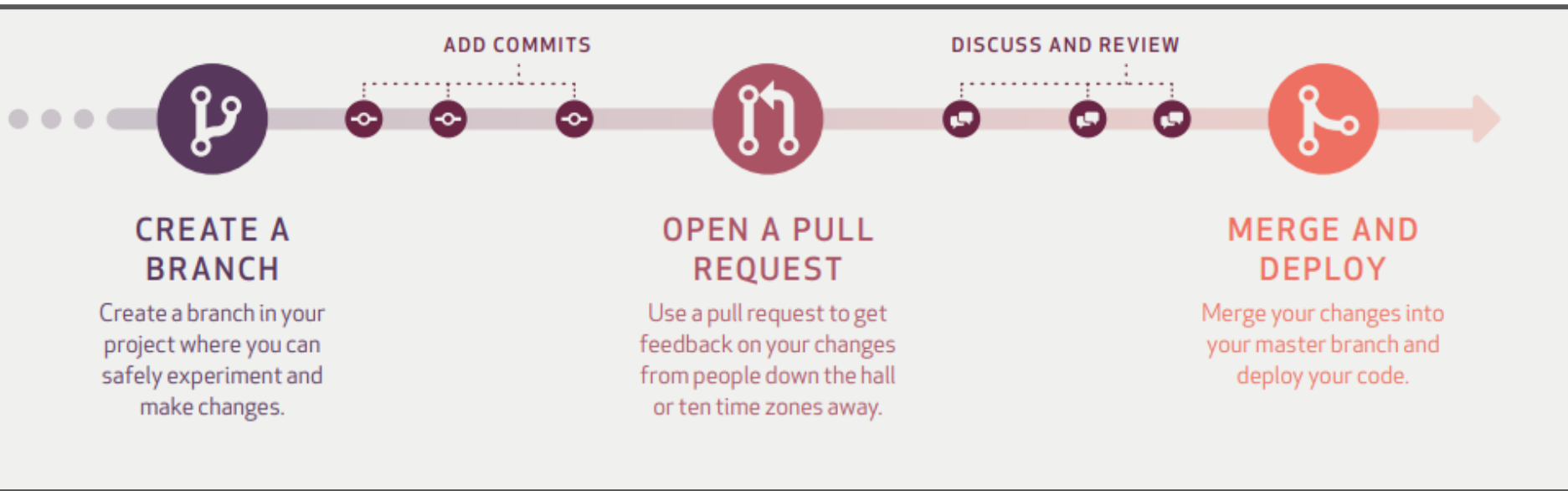**git merge –** applies those changes to the local

git checkout – tells Git which branch or commit to apply your new changes
- if you want to make changes on a branch that you don't currently have checked out, you need to check it our first

Local   **Remote**

working directory | staging area | localrepo | remote repo

git add
git commit
git push
git pull
git checkout
git merge

GITHUB EXAMPLES

BRANCH BASED WORKFLOW

ADD COMMITS

DISCUSS AND REVIEW

CREATE A BRANCH

Create a branch in your project where you can safely experiment and make changes.

OPEN A PULL REQUEST

Use a pull request to get feedback on your changes from people down the hall or ten time zones away.

MERGE AND DEPLOY

Merge your changes into your master branch and deploy your code.

GIT

PRACTICE

You Didn't Learn to ~~Ride a Bike~~ **use Git** by Reading a Manual.

GIT & GITHUB PRACTICE

OBJECTIVES

- PyCharm vs Terminal (Git Bash) VSC

- How to backup your code

- How to work with branches and pull requests

- How to collaborate

- How to resolve conflicts

TASK

LET'S SET UP OUR GIT

Note that from now on we will be actively using Git & GitHub to share materials and manage our code, as well as homework.

TASK

- In your GitHub create a new repo called Homework.
- Add your instructor as a collaborator to this repo, so they can access your homework submissions in the future and comment on your code.
- Clone this repo to your machine
- Create a new file called *my_first_git_homework_<your_name>.txt*
- Commit and push it to the remote repo
- Ensure that the new file can be seen in your GitHub when you access it from a browser like Chrome
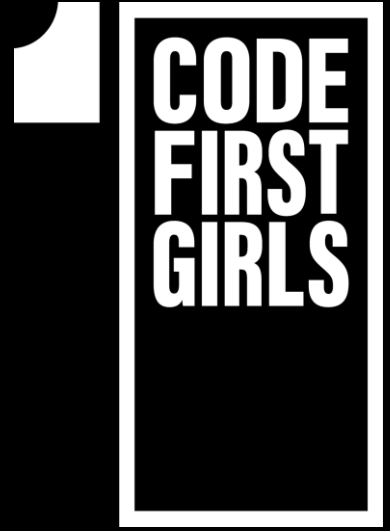
LET'S TALK

VSC

Why do we need to start using
Git & GitHub now**?**

USE CHAT WINDOW TO SHARE YOUR THOUGHTS

THANK YOU!