Introduction to Git and Github

-Pulkit Mahajan

Version Control Systems?

Installing Git on your system:

- Windows
    - Download git installer from git-scm.com/downloads
- Linux
    - sudo apt-get update
    - sudo apt-get install git
- Mac OS X

Initialize your Repo

- git init

.git directory stores all your internal metadata useful for tracking your project.

Identify yourself...

- git config --global user.name “LongClaw”
- git config --global user.email “yourEmail@xyz.com”

Checking Status

- git status

Staging files

- git add file1
- git add file1 file2 file3 …
- git add .

Commiting changes

- git commit -m “Commit Message”

Going Back!

git log

- Scroll down to your desired version and copy its *hash*.

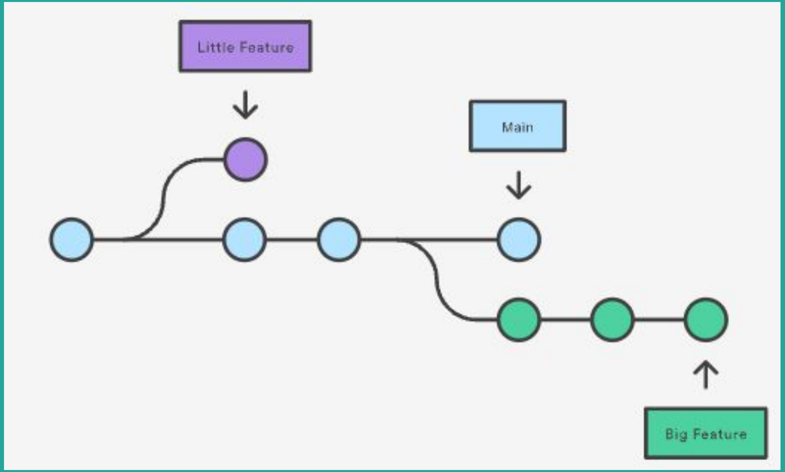
git checkout *commit-hash*

- use git log again
- notice that you are now in a detached head state
- any commit made to this detached head goes to a separate branch.
- If you need that branch: name it!
  - git switch -c *branch name*

To go back to latest version:

- **git checkout master**

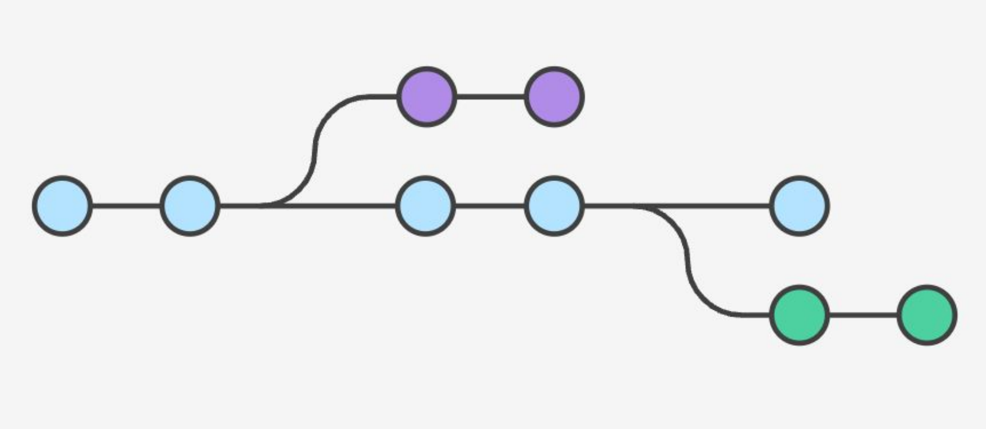
Creating Branches



- git branch
  - Shows the list of all branches
- git branch *new_branch_name*
  - Creates a new branch; branching from your present location on git log
- git checkout *desired_brach*
  - Switches your workspace to the desired branch

Deleting branches:
git branch -d *branch_to_be_deleted*

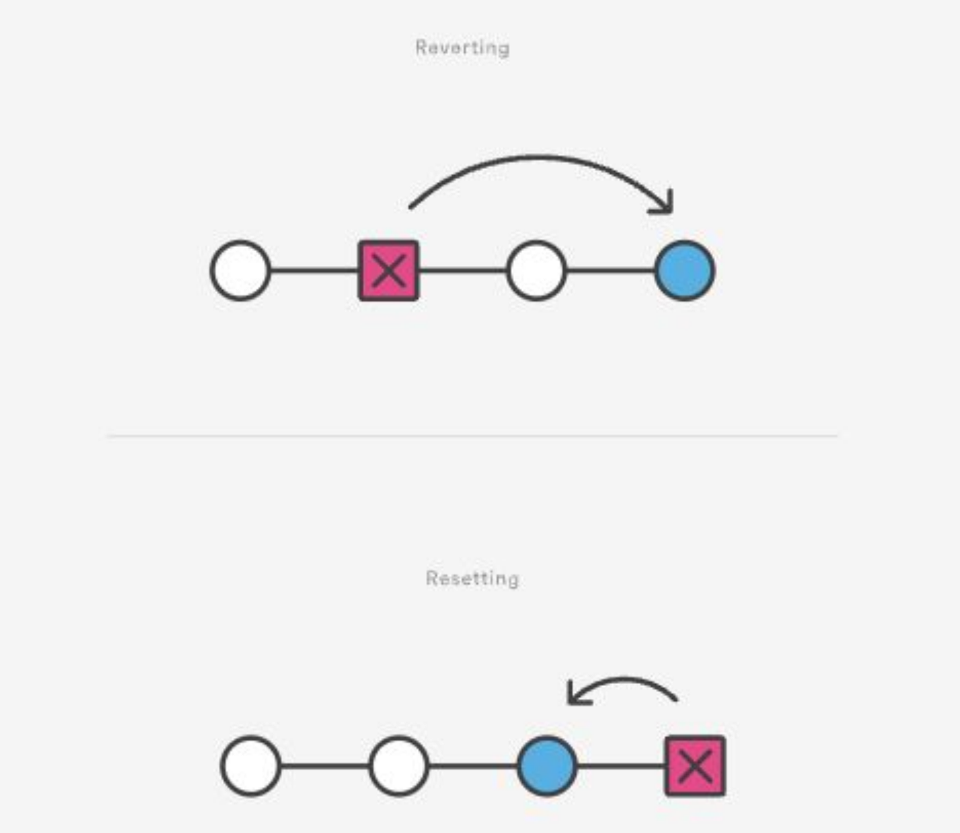
Merging Branches



- git merge *branch_name*

git revert

git reset



GitHub

Social Networking for developers

- Follow
- Rate
- Collaborate
- Communicate

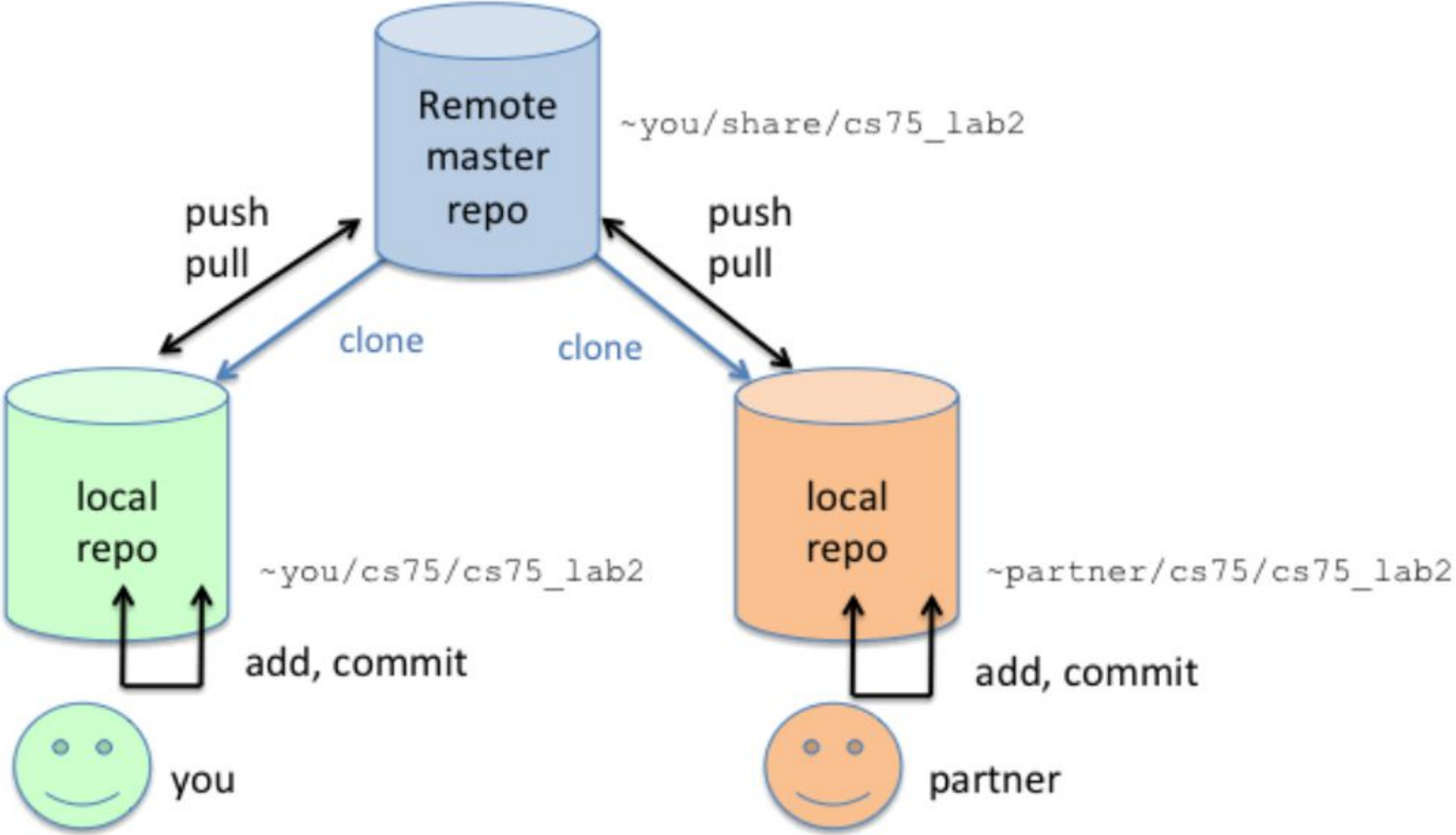
github.com/Pulkit-m

Git vs GitHub

GitHub is a hosting platform for Git repositories. You can use Git on its own without Github (and other similar platforms), but it's difficult to collaborate or share your code with. others.

- **Git is the version control system, the tool that tracks changes to our files over time**
- **Github is a hosting service for projects that use Git.**

Make sure to first familiarize yourself with Git before proceeding. Git is used to store projects inside *repositories* and track the complete history of all changes to the project code. Using GitHub, we can upload a local project repository to a remote cloud-based GitHub repository. We can also interact with public repositories published by other developers.


Remote master repo
~you/share/cs75_lab2
push
pull
push
pull
clone
clone
local repo
~you/cs75/cs75_lab2
local repo
~partner/cs75/cs75_lab2
add, commit
add, commit
you
partner

Register

https://github.com/join

- Set up your account
- Choose your subscription: Free

Using GitHub

Common Workflow

- Add/commit your code locally
- Go to Github and make a new repository
- Connect your local repo to the github repo (add a remote)
- Push your code up to github using the new remote

Step 1: A local Git Repository -*done*

You will either have a local repository that you need to put on GitHub
Or
You may have to clone some repository owned by you or some other developer

Step 2:
Make a Repository

Creating a Personal Access Token

To push another branch to GitHub:

```
git push origin branch_name
```

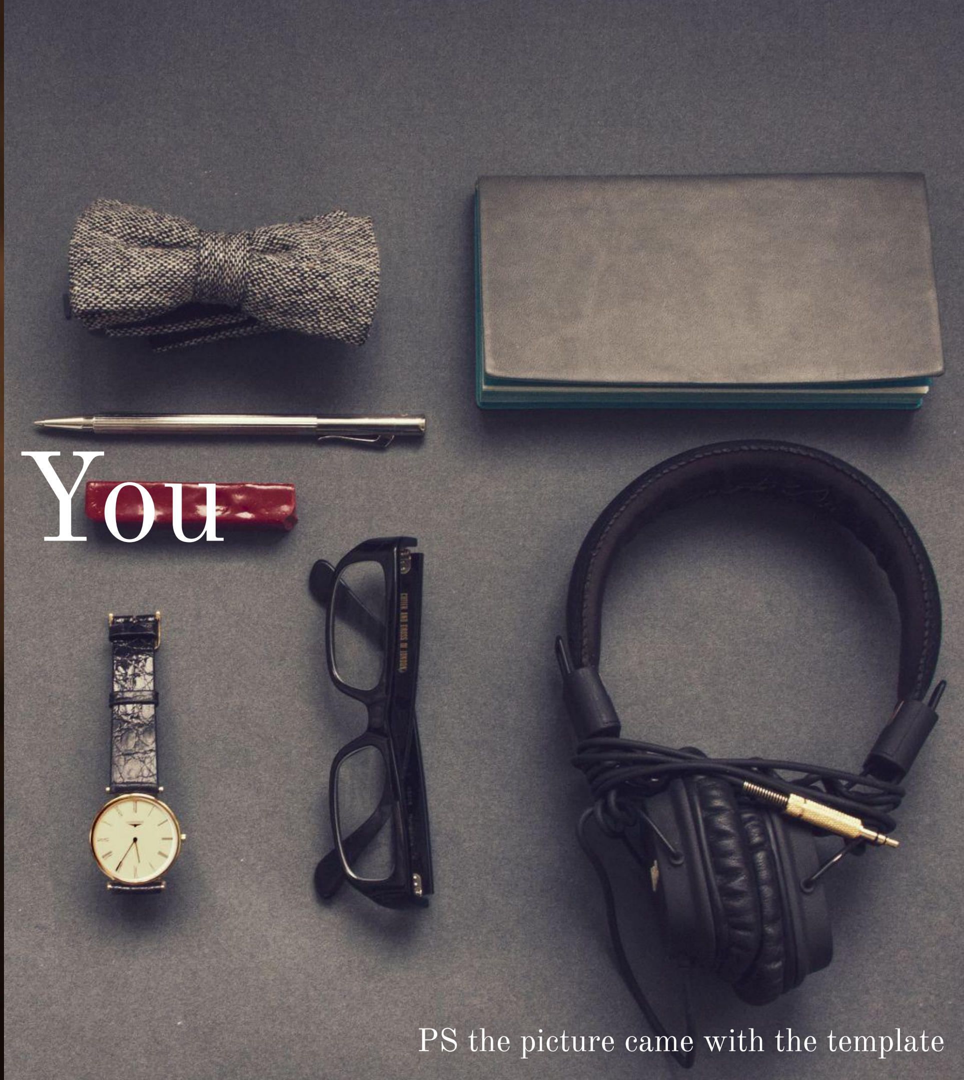
To pull a particular branch from GitHub:

```
git pull origin branch_name
```

Cloning an Existing GitHub Repository

Until now you've been tinkering with your own repository…
When working with someone else's repository… you need to follow certain protocols.

Creating a Pull Request.



Thank You

PS the picture came with the template