Git Basic

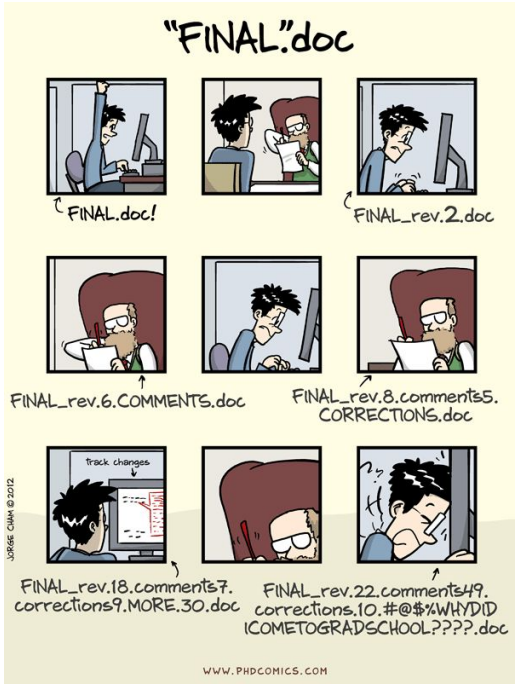
Outline

Novice course - You know NOTHING about Git!

- What is Git & GitHub
- Hands-on Excercise: Setting up Local & Remote Repo
- What are Branches
- Git Collaboration/Workflow
- Hands-on Excercise: Setting up Branches

What is Git

History - Maintain a set of history of our work that can rollback anytime



What is Git

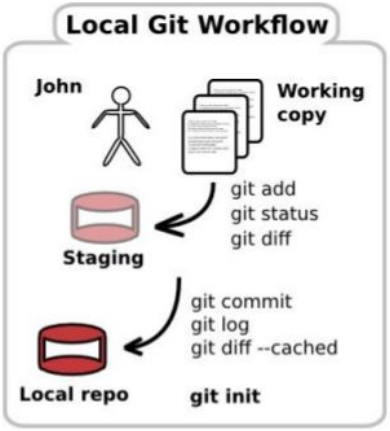
Popular version control system for tracking changes and coordinating work on those files among multiple people

- Unlimited “Undos”
- Start with your base version of your code
- Record changes each step along the way
- You can rewind to start at the base code and play back each change you’ve made up to your current version

Everything locates locally on your desktop, do not need to set up a server. You just need to install a Git software to your desktop.

Work locally and offline

What is Git



Local Git Workflow

Project = Repository (repo) - where git save/keeps all the data/changes/history

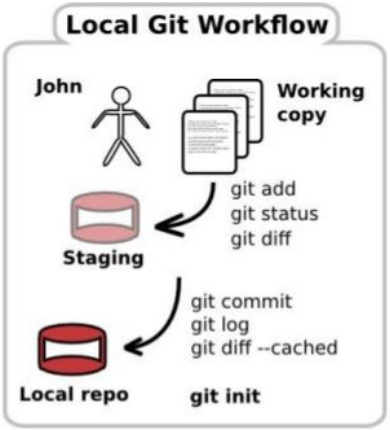
Working Directory - Folder location on your computer where your project (repo) resides

Staging - allows you to continuously work on your file until you are ready to commit. For example, you may have multiple working copies file1 to file 11 that you are working on and at the end, you only want to save file11, you can just add file11 to be committed

Commit - git's way of saving files and tracking its history

Now that you understand what Git is.
Let's set up our first Git.

Git in Action (Setup Local Repo)

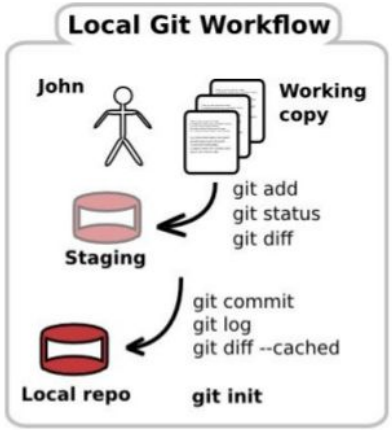


Create a Local Repository

- Download and Install Git from git-scm.com
- Open a terminal to verify that installation was successful
    - $ ***git --version***
      git version #.#.#
- Make sure to setup some default configurations in git like username and email
    - ***git config --global user.name "Your name"***
    - ***git config --global user.email "Your email address"***
- Create a folder on your desktop (e.g. /myfirstgit)
- Go to your folder (**cd /myfirstgit**)
- Type "***git init***" to initialize your local repo so that git knows this is where your project (repo) files resides.

Let's create our first file!

Git in Action (Setup Local Repo)

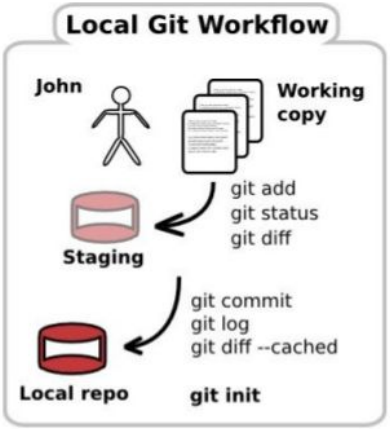


Go you your folder “/myfirstgit”

- Create a file
  - ***touch firstprojectfile.txt***
- Lets see what is changed in this folder (file name is in red)
  - ***git status***
- Lets stage it!
  - ***git add --all***
- Lets look at what’s happening (file name should be green meaning its ready to go)
  - ***git status***
- If everything looks good, lets save it and make a note what was changed on this file so that it will show up when we go back to history.
  - ***git commit -m “created my first file”***
- Lets see our history
  - ***git log***

    commit e21478af1c3babb2b40091bf6662e7791eeeecf2
    Author: Wendy <wlam@ualberta.ca>
    Date: Wed May 2 16:36:22 2018 -0600
      Created this new project file

Git in Action (Setup Local Repo)



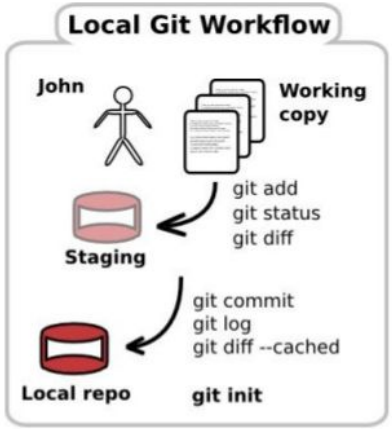
Lets **add content** to the file

- Edit the file “firstprojectfile.txt”, add “This is my first git file” then save
- To see what the difference is between this edited file and the original
  - ***git diff***

```
diff --git a/firstprojectfile.txt b/firstprojectfile.txt
index e69de29..edc1c04 100644
--- a/firstprojectfile.txt
+++ b/firstprojectfile.txt
@@ -0,0 +1 @@
+This is my first git file
```

- **git status** to see the file in staging
- **git add --all** to add the file to staging
- **git commit -m “Added comment to firstprojectfile.txt”**
- **git log** to see the history of our file

Git in Action (Setup Local Repo)



How to revert to previous versions?

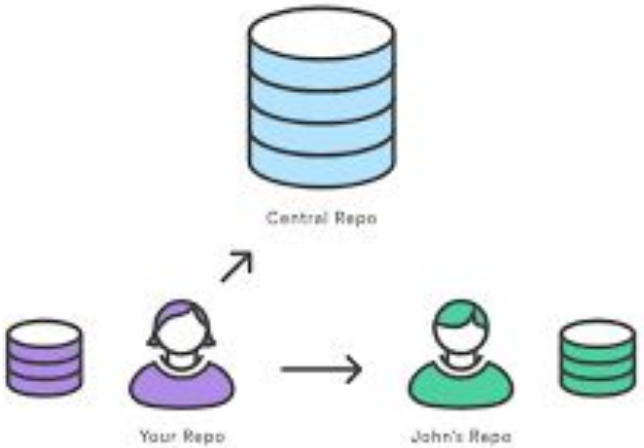
- Lets see the history
  git log
  commit **e21478af1c3babb2b40091bf6662e7791eeeecf2**
  Author: Wendy <wlam@ualberta.ca>
  Date: Wed May 2 16:36:22 2018 -0600
  Created this new project file

- If you want to revert back to your original version, type
  git reset --hard <commit hash number>

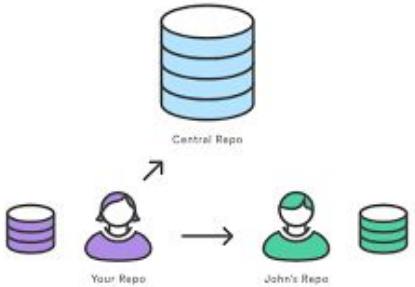
- Commit
  git commit -m “Revert to previous commit version”

What is Git: Collaboration

Collaboration - We can: collaborate with the rest of your team at anywhere and anytime.



What is GitHub



- GitHub is a company that hosts git repository in the web and provides web interface to interact with repo they host
- You can access and collaborate with your team anywhere and anytime
- Anyone can work on your file if you give them permissions
- You will always have up-to-date repository, never behind

Git in Action (Remote Repo: GitHub)

- Create a GitHub account (github.com)
- Choose Free Personal Account (Public Repository = anyone can see your repo but cannot make changes to it unless you give them permission. Do not store sensitive information e.g. password)
- Login to your account in github
- Create a new repo (+ sign top right)
- Add Repository name, name it “myfirstgit”
- Click “Create repository”

Let’s push our repo to GitHub!

Git in Action (Remote Repo:GitHub)

If you had already committed your local and your local repo is up-to-date then skip to the next slide. If not, follow the following commands:

- Go to **/myfirstgit** folder
  * In our example, we've already initialize the folder. However, if you have not initialize your local directory as git repository, type "**git init**"
- Next Staging your files
  "**git add .**" (add all files in your folder to be committed)
- Commits the files that you've staged in your local repository
  "**git commit -m "First commit**"

Git in Action (Remote Repo:GitHub)

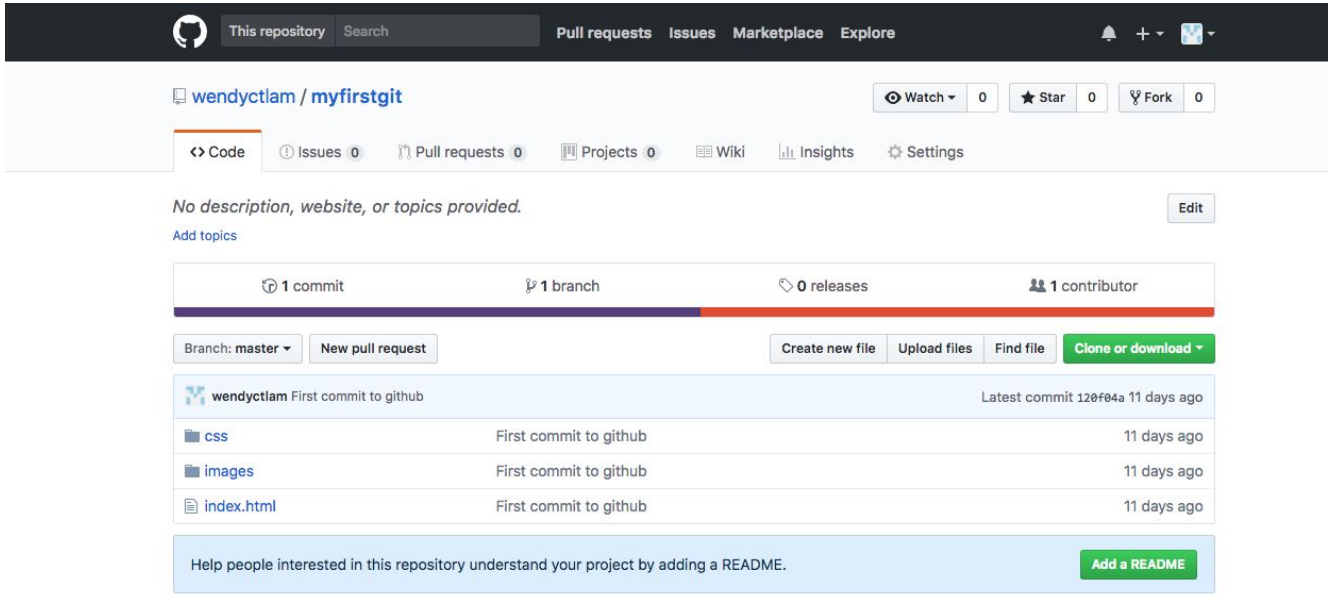
Let's push our local repo to GitHub

- At the top of your github repository's quick setup page, copy the link
  https://github.com/<yourusername>/myfirstgit.git
- In Terminal, add the URL for the remote repository where your local repo will be push to
  git remote add origin <your remote repository URL>
- Verify the new remote URL
  git remote -v
- Now Push your changes in your local repo to GitHub
  git push -u origin master
- **Check in github to see your myfirstgit directory is there.**

Git in Action (Remote Repo:GitHub)

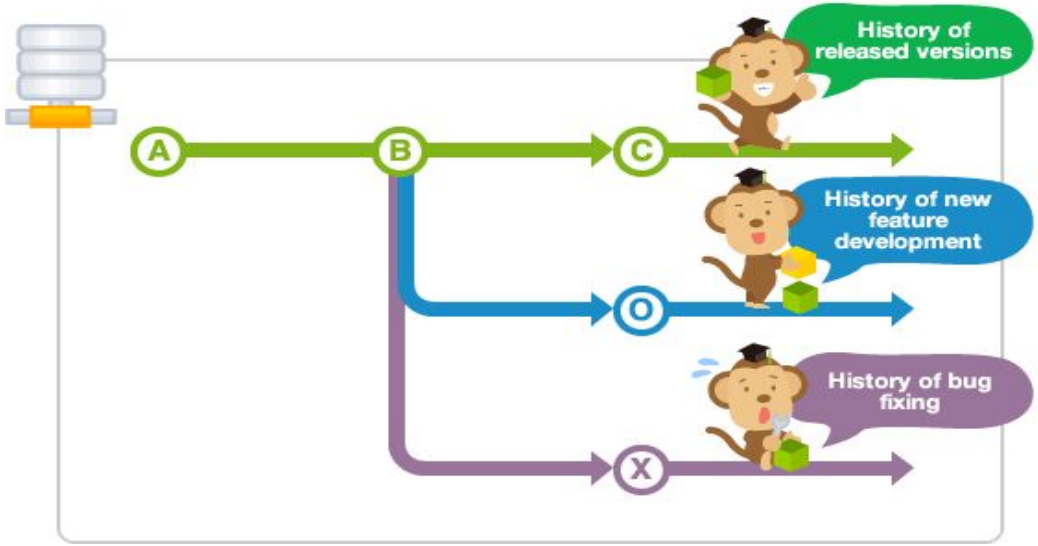
Let's load our files directly through GitHub Web Interface. It is recommended to push it to GitHub using the command lines but if you really need to do it quick, this is an alternatives as well.

Profile > Repositories > Choose the Repository you want to drop your files to



What are Branches

Branches - We can be working on different tasks independently and simultaneously.



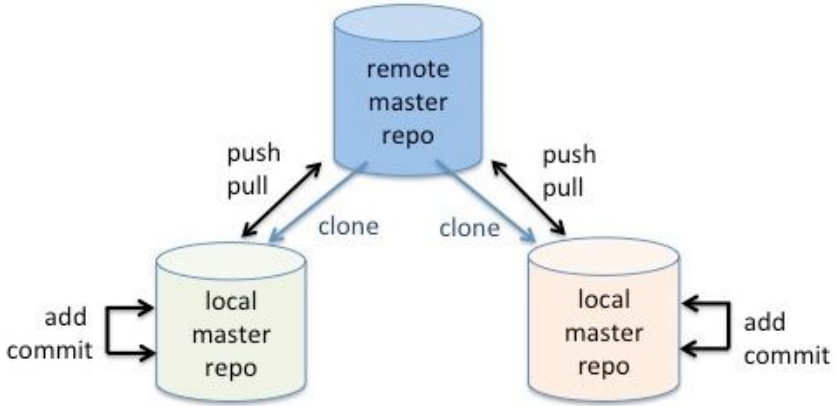
Overview

So far...

- We have a basic knowledge on git
- Benefits of using git
- Basic git commands
- We've created a local and remote repository in GitHub
- Briefly talked about what Branches are

Git Collaboration/ Workflow

Clone GitHub repo



In your local repo, go to your repo directory, then type the following command to clone Denny's full repo.

git clone https://github.com/chaojenyang/denny.git

After clone , you will see a directory called Denny and you will see README.md file

Add contributor to your repo

You can view all the change in GitHub as well by visiting the link
https://github.com/chaojenyang/denny

Manipulate clone repo (local repo)

git status

Touch yourName.txt
Inside the file , write anything you want

git status

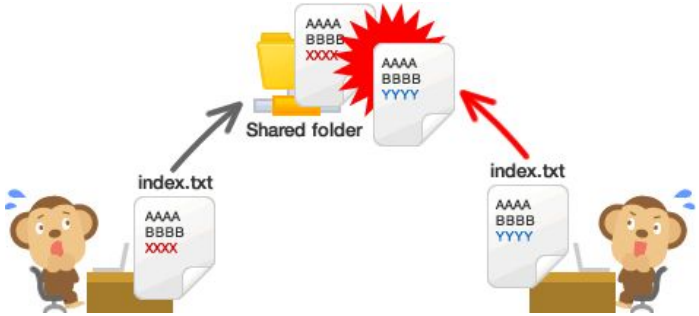
git add yourName.txt

git commit -m “my first collaboration file”

Show your remote repo (GitHub)
git remote show origin

git push

Error push to GitHub repo



You might get error like this

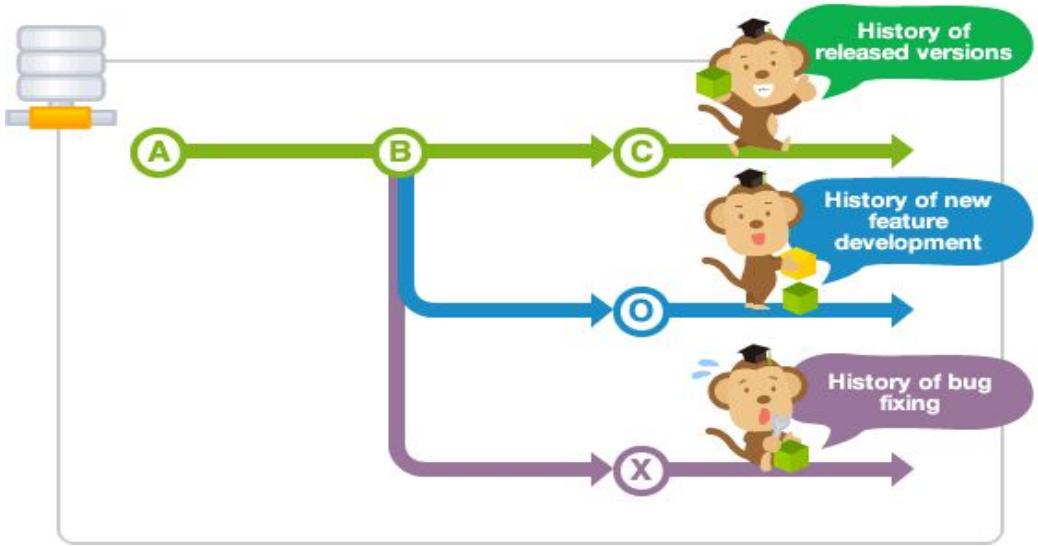
Master branch in Github is ahead of your local repo

To resolve this , merge the remote master to local master
git pull

git push

Branch in Git

Branch is essentially an independent line of development. You can take advantage of branch when working on new features or bug fixes as it helps to isolate your work from that of other team members.



https://backlog.com/git-tutorial/stepup/stepup1_1.html

Create new branch locally and remotely

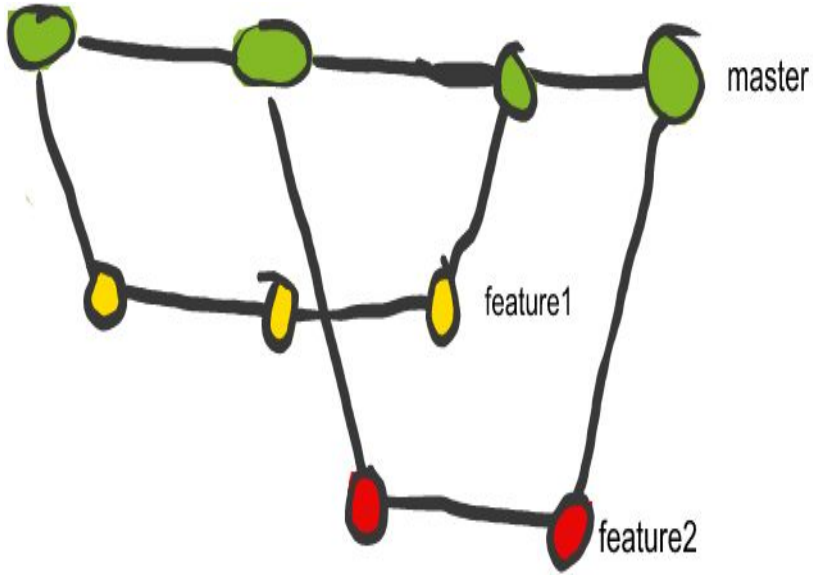
Create new local branch
git checkout -b yourBranchName

Check which branch you are in
git branch

If you are not in the branch that you created
git checkout yourBranchName

Compare local master and your branch
git diff master yourBranchName

Push your local branch to GitHub
git push origin yourBranchName



Make modification to new branch -1

Modify your own file

git add --all

git commit -m “your first change your file in branch”

Git diff master yourBranchName

git push origin yourBranchName

Make modification to new branch-2

Add new file
touch project_yourname.txt

git branch

git add --all

git commit -m “your second change your file in branch”

Check your local branch is different than local master
git diff master yourBranchName

Push new change to GitHub
git push origin yourBranchName

Merge branch

Check which branch you are in
git branch

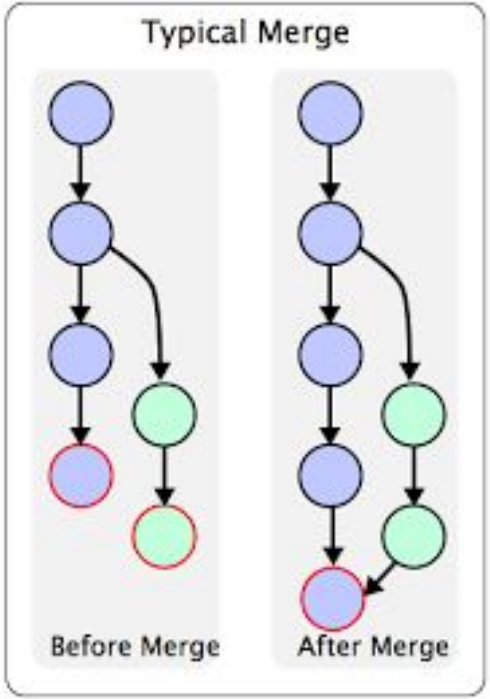
Switch to local master branch
git checkout master

To make sure local master branch is up to date in remote master
git pull

Merge your branch to local master branch
git merge yourBranchName

git diff master yourBranchName

git push



Delete branch remote

Check git branch locally
git branch

Making sure you are master branch

To delete remotely
git push origin --delete yourBranchName

To delete locally
git branch -d yourBranchName

Overall Summary

- What is Git
- Git Repository (Local and Remote using GitHub)
- Collaboration
- Branching