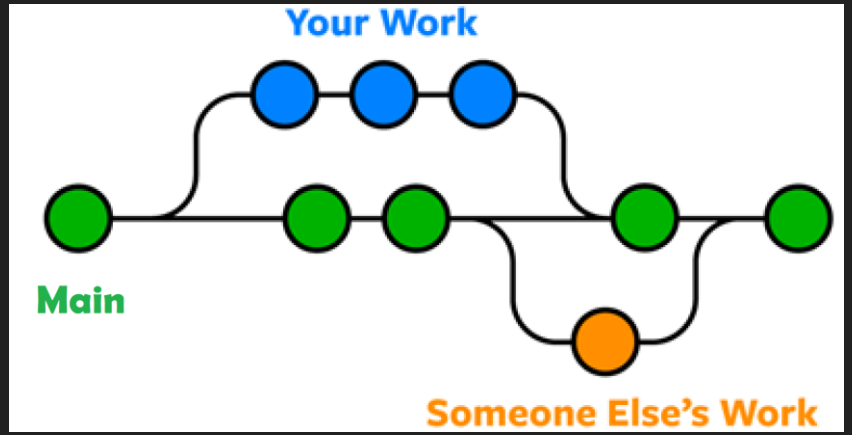
Git Branching

What is Branching?

Branching means you diverge from the main line of development and continue to do work without messing with that main line.

It is very useful when working collaboratively with others - each person can work on their own separate branch, and when ready, merge their work in with the main branch.



Creating a Branch

$ git branch <new branch name>

Note: this does not switch you to the new branch.

To see this new branch, run

$ git branch

You will see a list of all branches with a star next to the branch you are currently on.

Changing Branches

To switch to another branch, use

$ git checkout <branch name>

When you switch branches, git changes the files to reflect the last commit of that branch. (Try it!)

Also, you can’t switch branches if you have uncommitted changes (unless you stash those changes)

The original branch in any repo is called Main

Create and switch to new branch

You can combine creating a new branch and switching into it:

$ git checkout -b <branch name>

Pushing to Branches

Git by default pushes changes to Main. If you want to push to another branch you need to do so explicitly. One option is to specify the branch each time you push:

```
$ git push origin <branch_name>
$ git push origin main
```

You can also permanently set the branch you want to push to with:

```
$ git push --set-upstream origin
```

Recommended Workflow

Clone down the group’s repository.

$ git clone <repo link>

Create an individual branch for yourself in which to do work.

$ git checkout -b <branch_name>

When done, push changes to your branch on github.

$ git push origin <branch_name>