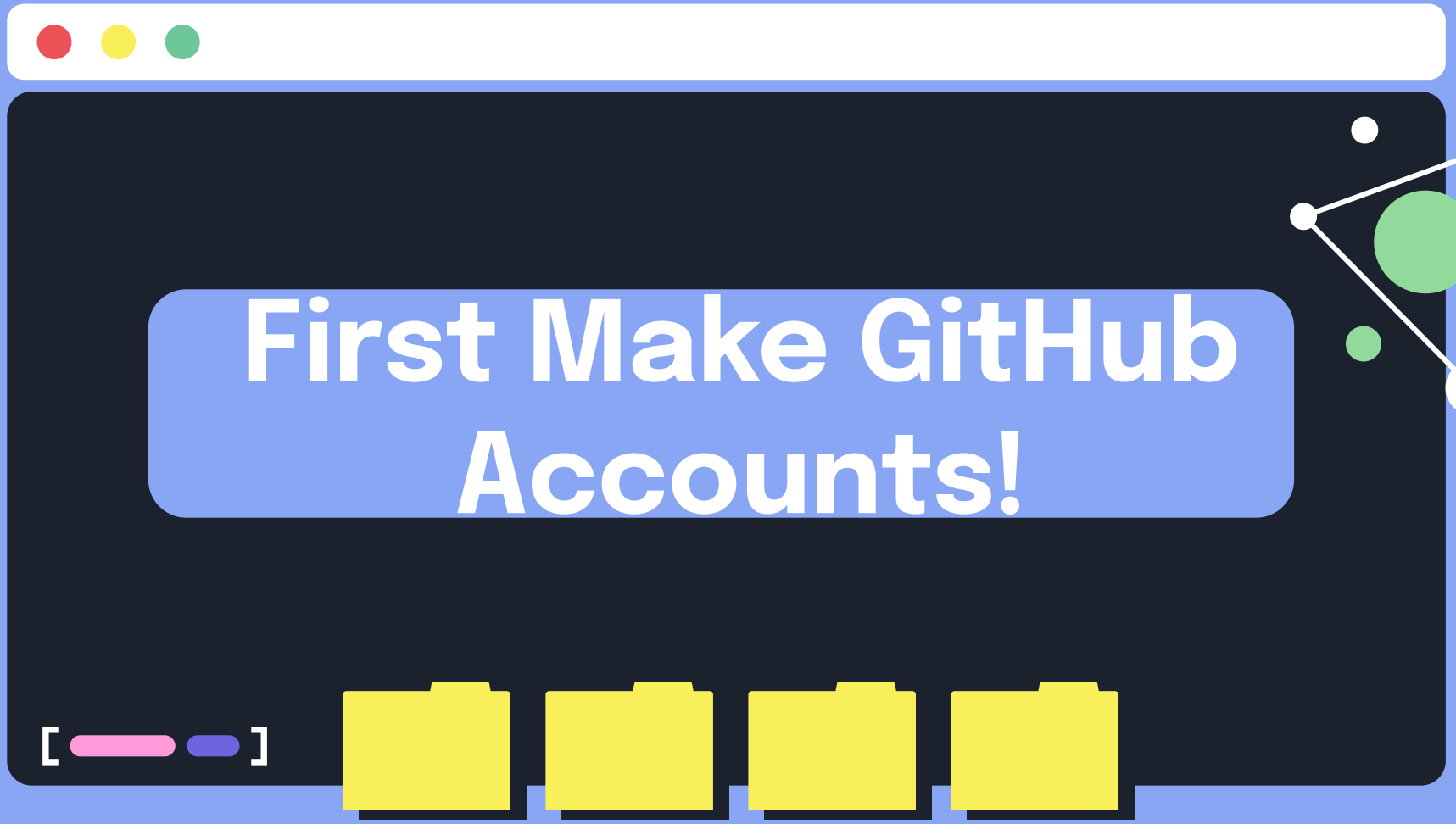
# GitHub Workshop

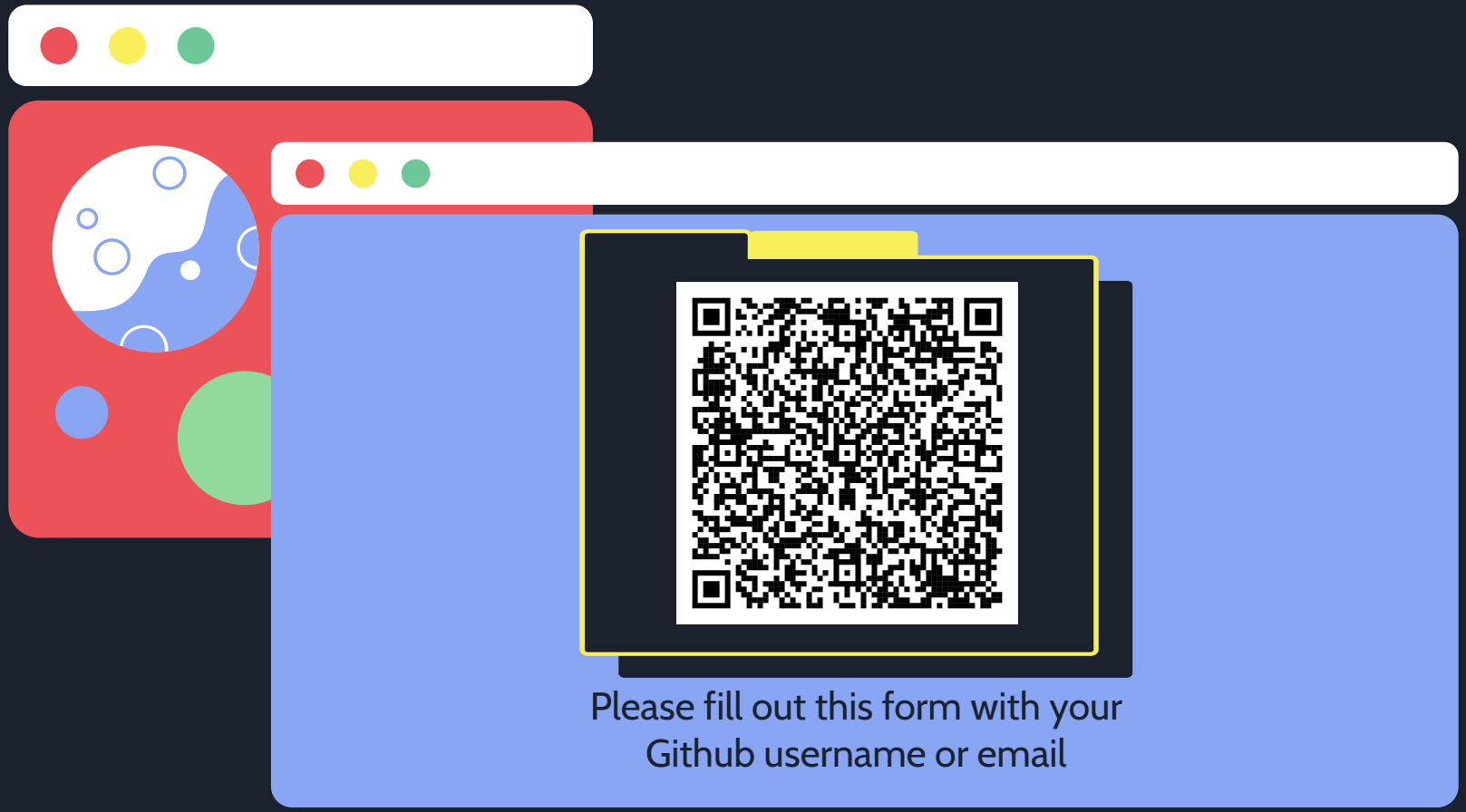
## w/ WiCS

11/13/24

## MyWPI

## Attendance


First Make GitHub Accounts!


Please fill out this form with your
Github username or email

# Upcoming events...

**Thanksgiving!**

11/25 w/ friends or family

**WiCS Study Hours**

12/11 @ 5pm in IS205

**WiCS Exec Elections**

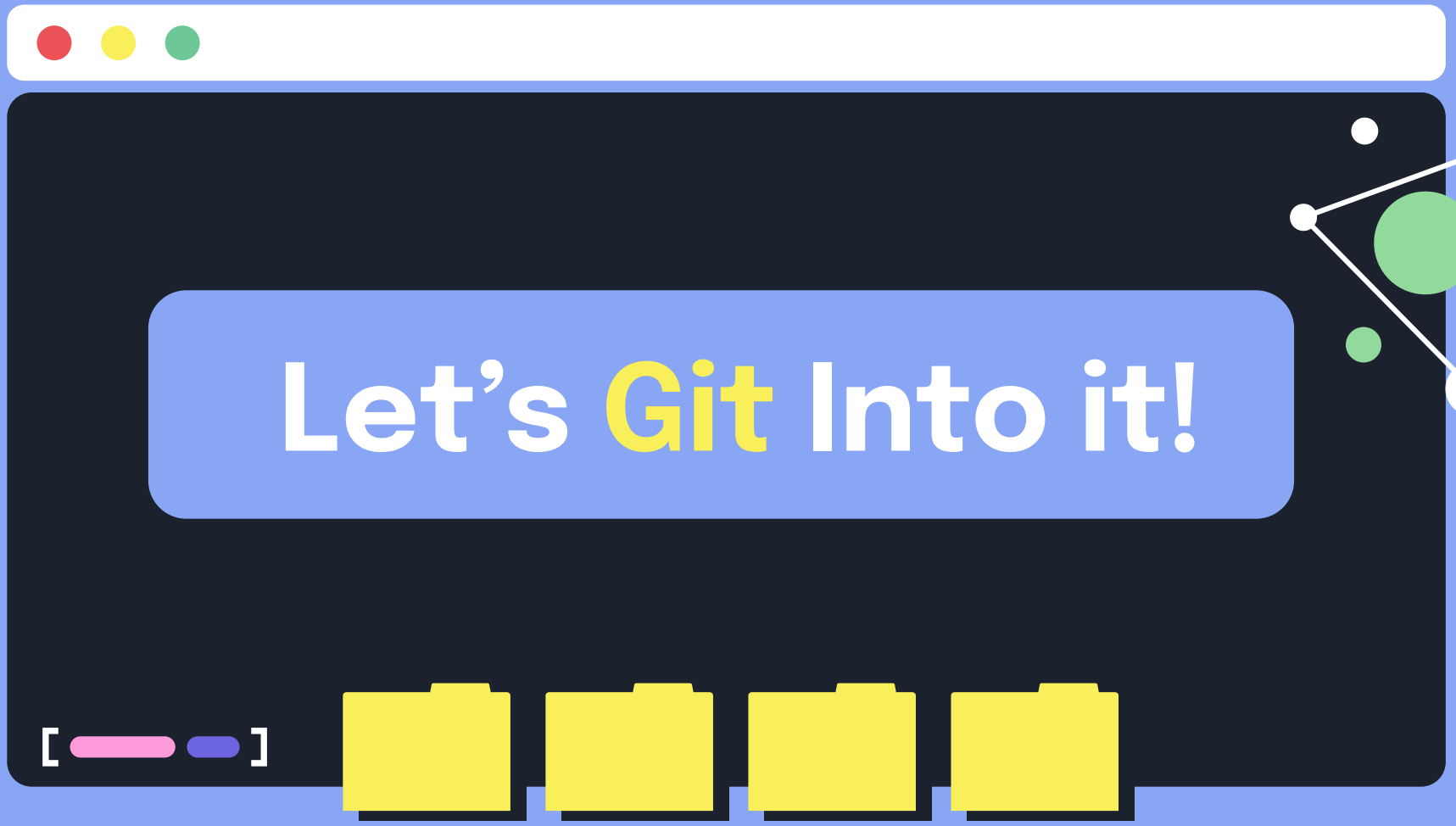
11/20 @ 5pm in IS205

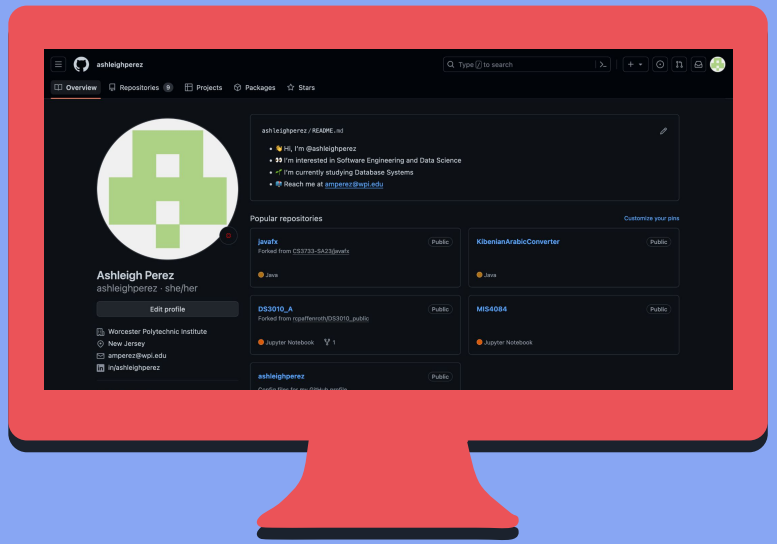
**WiCS Holiday Bash**

12/04 @ 5pm in IS205

**Winter Break!!!**

Go home and rest!

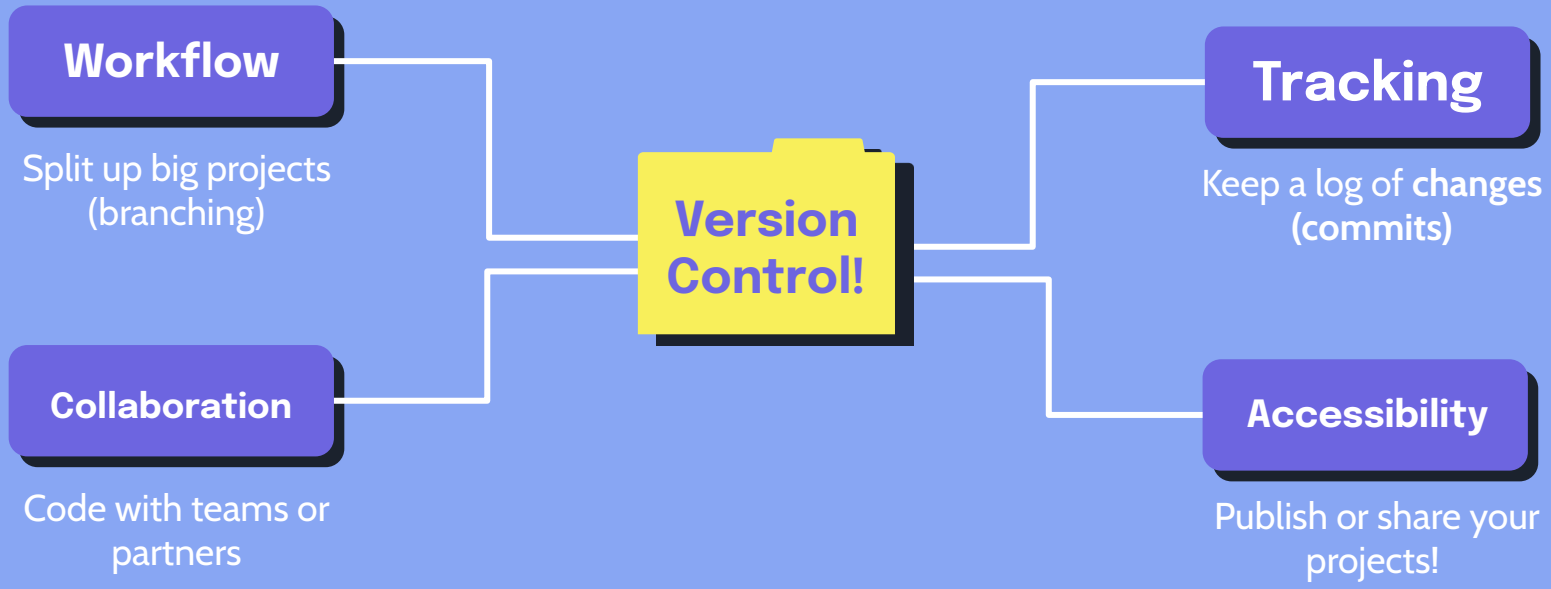

Let’s Git Into it!




# GitHub

a cloud-based Git repository hosting service

# What is Git?

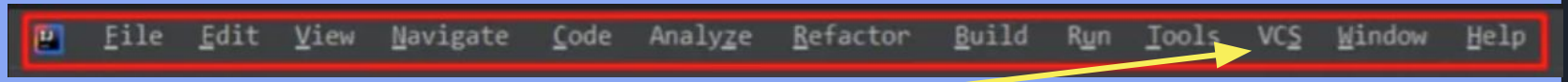

Workflow
Split up big projects (branching)
Collaboration
Code with teams or partners
Version Control!
Tracking
Keep a log of changes (commits)
Accessibility
Publish or share your projects!

# Setup!

Command Prompt


Through your IDE

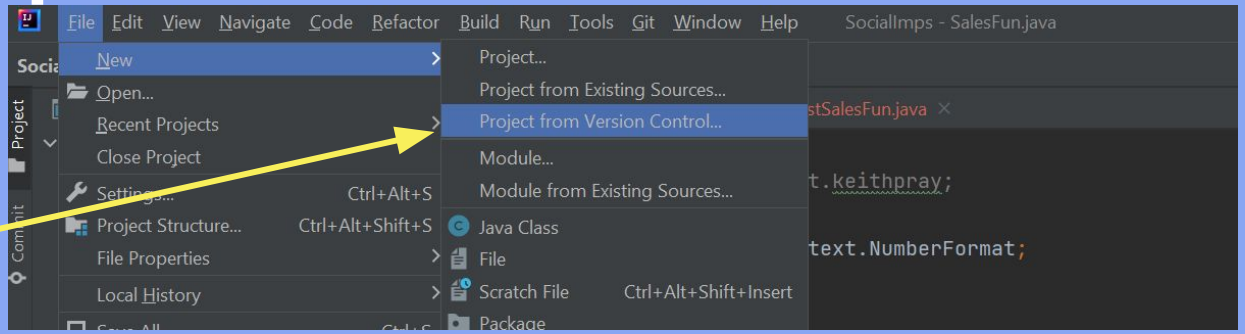
# Setup in IntelliJ



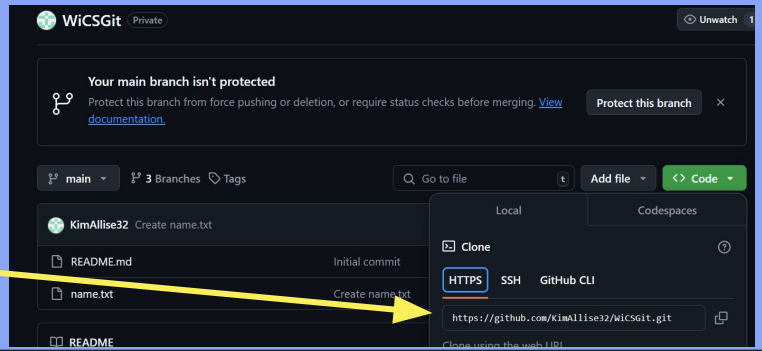

Click this and then
click git

# Setup in IntelliJ

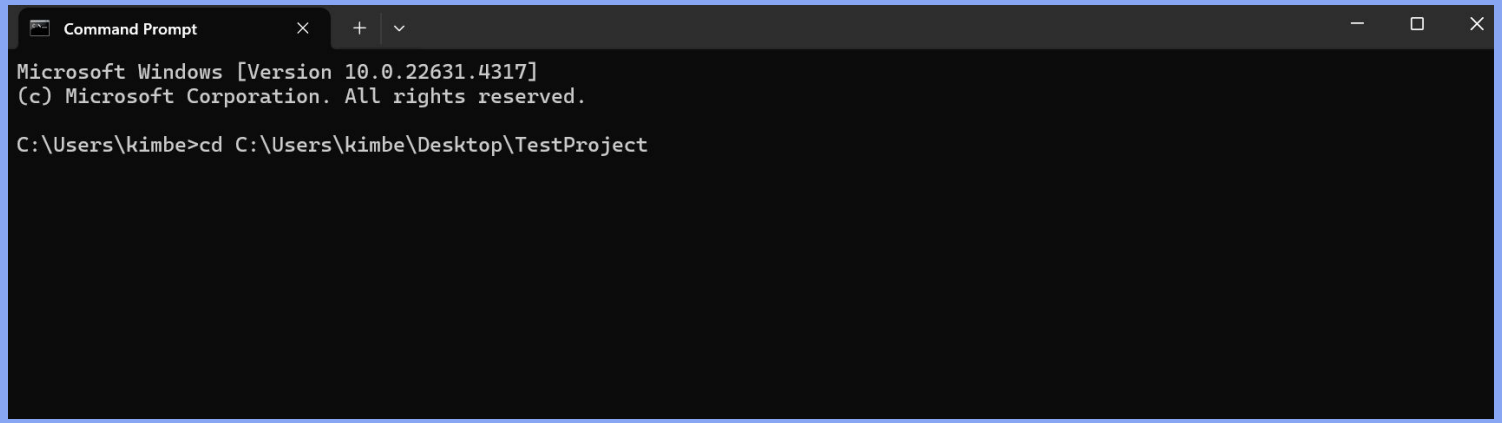
IDE




GitHub



# Setup in Command Line



# Setup in Command Line

First, you'll need to tell git about yourself. Get your username and token together from your settings page.

Then run:

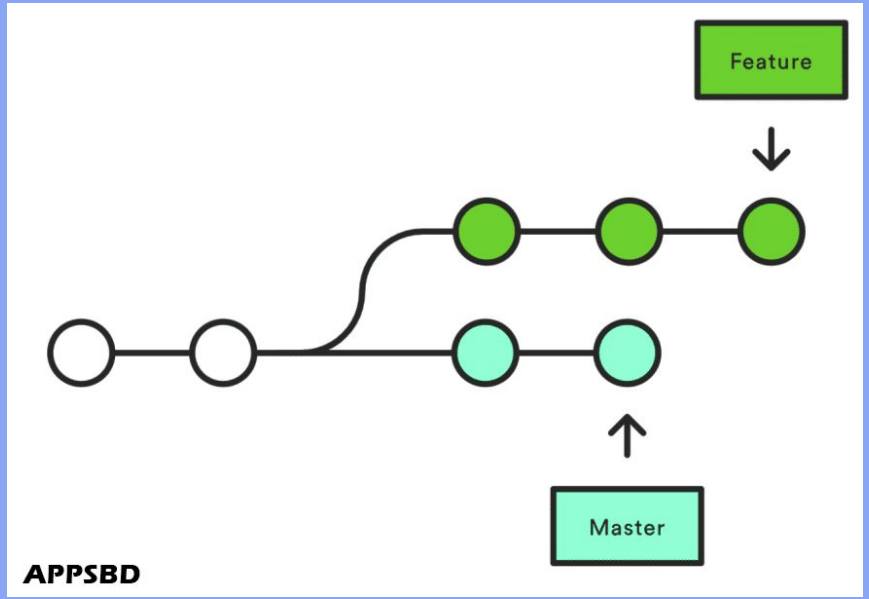
```
git config --global github.user YOUR_USERNAME
git config --global github.token YOURTOKEN
```

You will need to generate a new key if you don't have a back-up of your key.

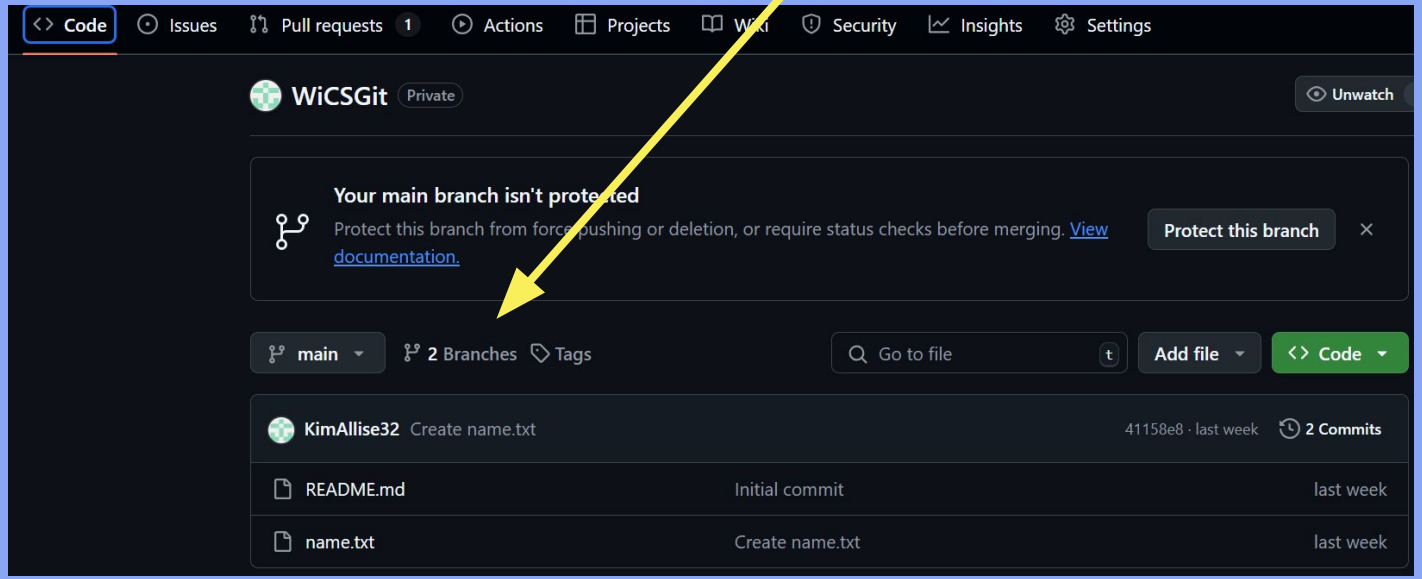
Then you should be able to run:

```
git clone git@github.com:YOUR_USERNAME/YOUR_PROJECT.git
```

# What is Branching

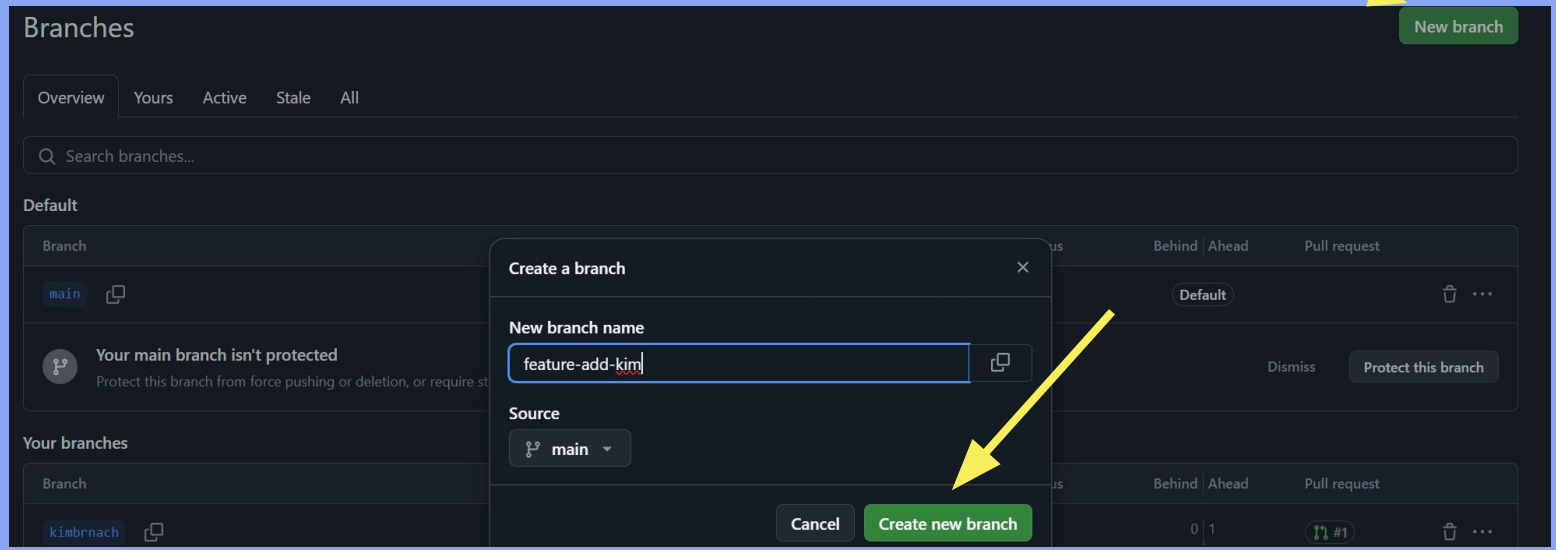


# What is Branching



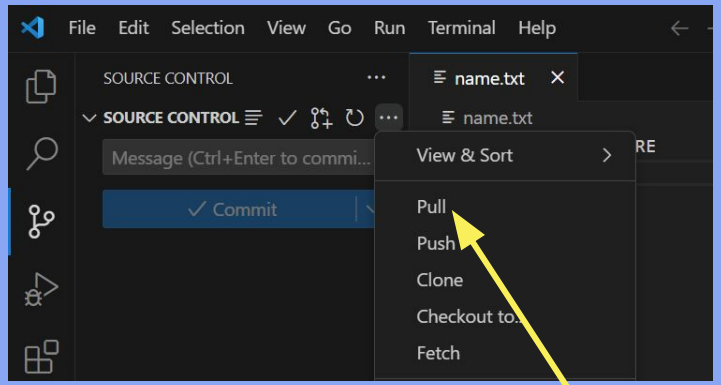
# What is Branching

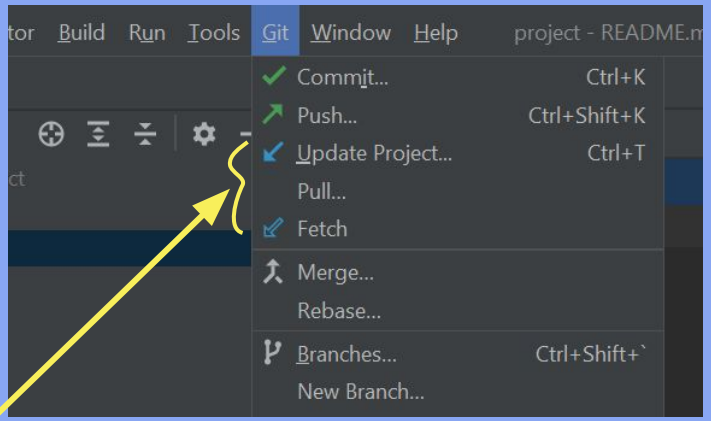
Descriptive names for branches!



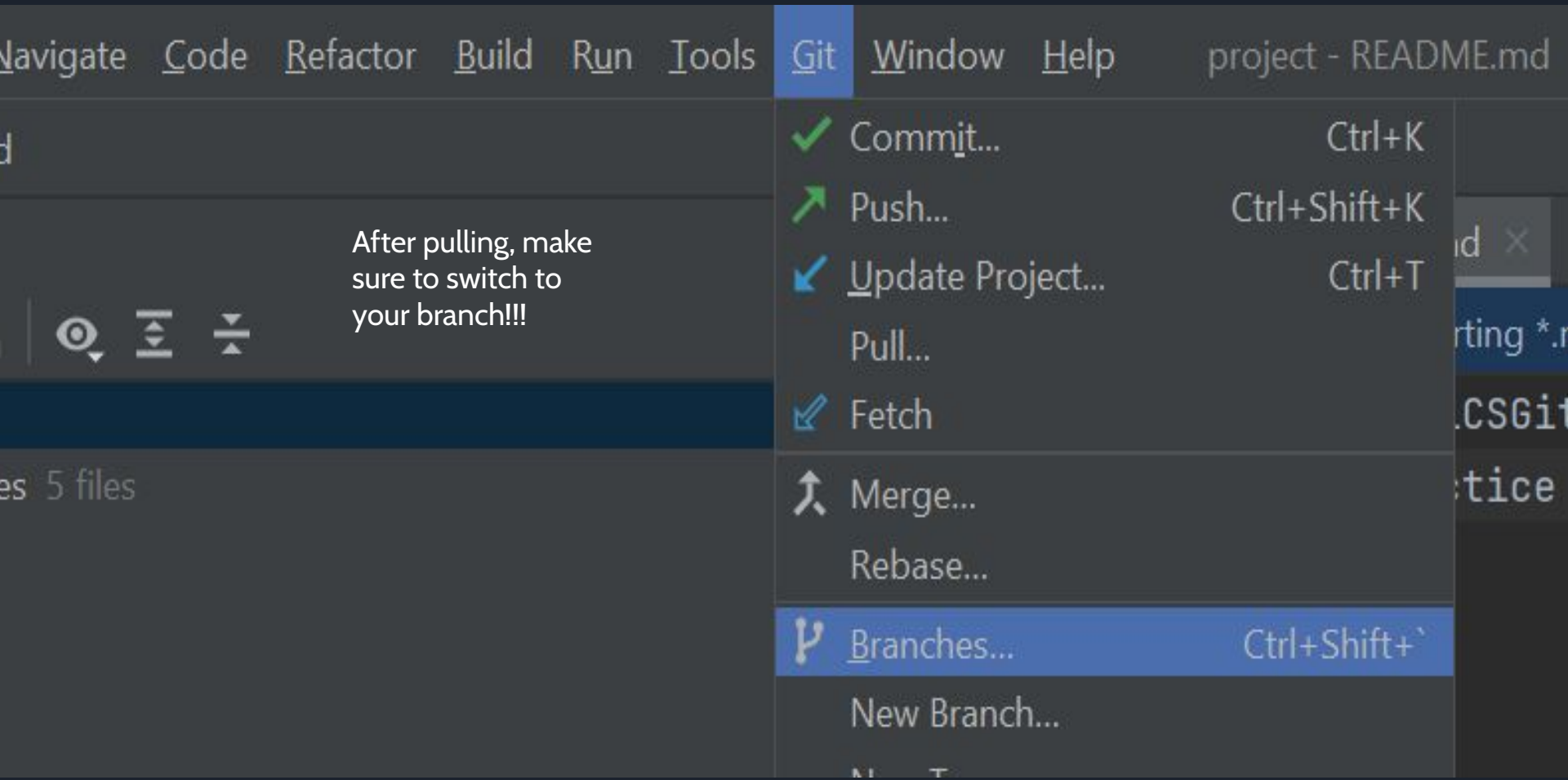
# What is Pulling

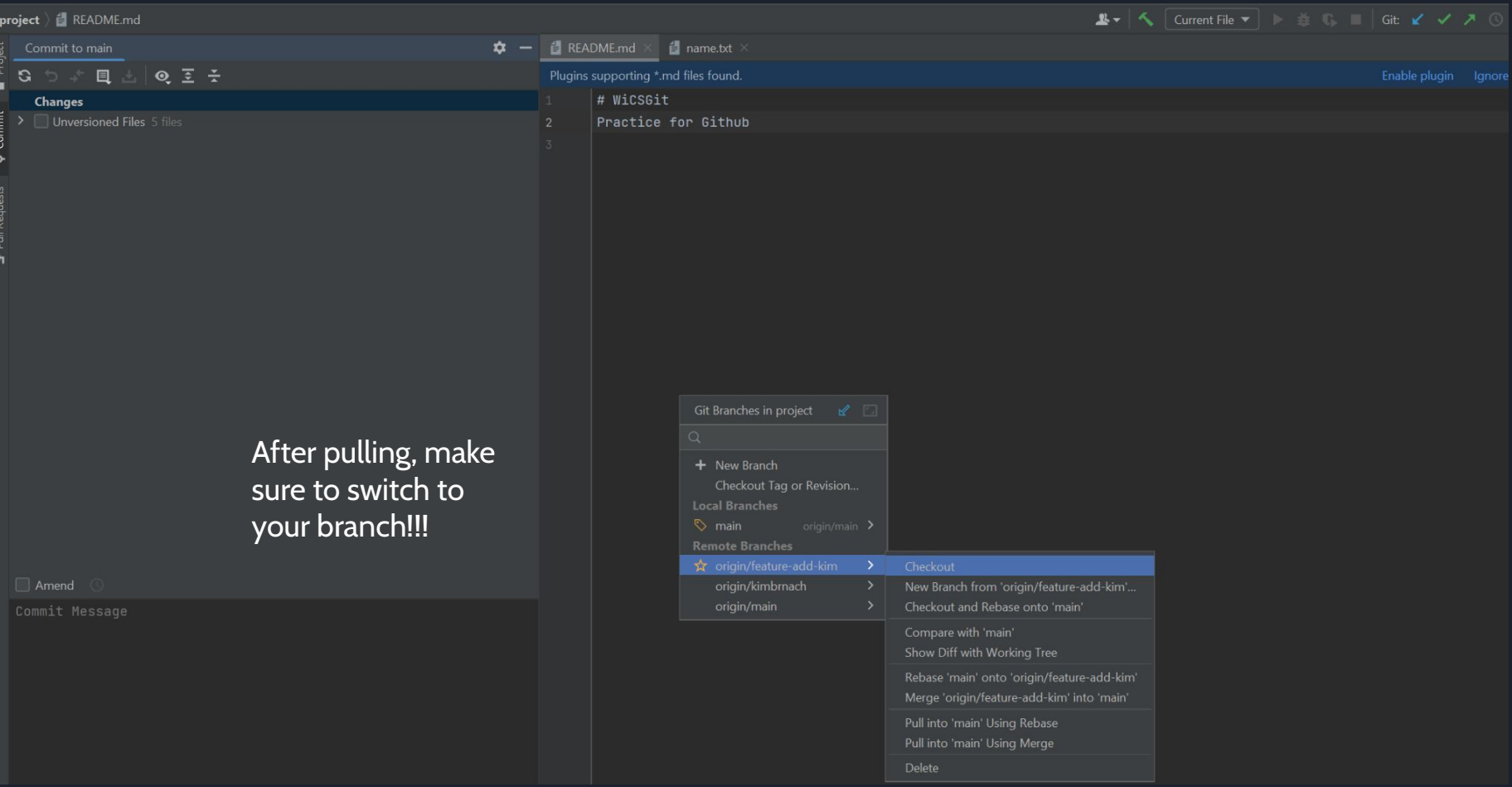
# What is Pulling




Different IDEs will have slightly
different locations for git commands


Navigate Code Refactor Build Run Tools Git Window Help project - README.md
Commit... Ctrl+K
Push... Ctrl+Shift+K
Update Project... Ctrl+T
Pull...
Fetch
Merge...
Rebase...
Branches... Ctrl+Shift+`
New Branch...
After pulling, make sure to switch to your branch!!!
es 5 files

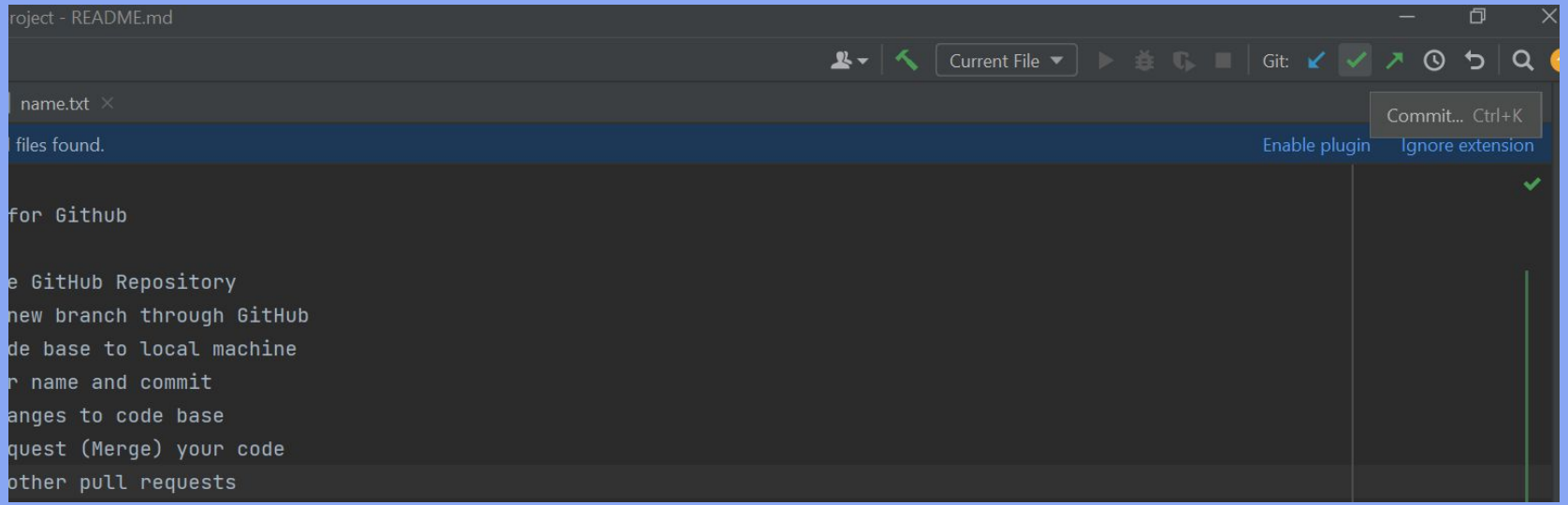



After pulling, make sure to switch to your branch!!!

# What is a Commit

Commit after making
changes to you branch!

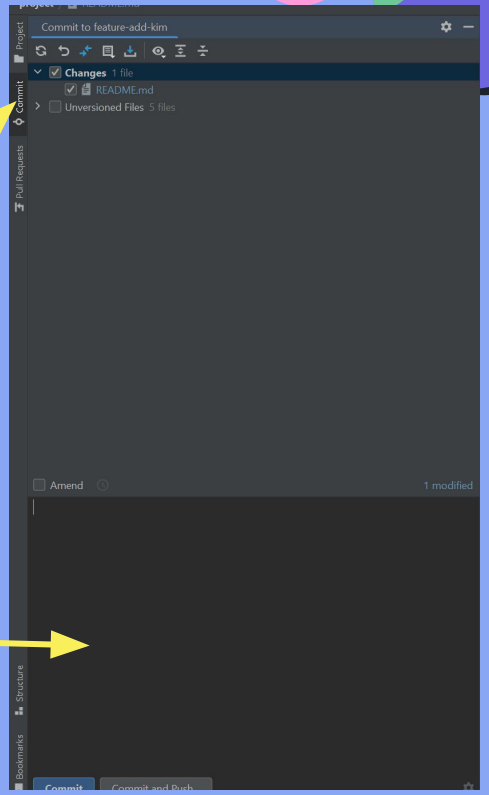
# What is a Commit



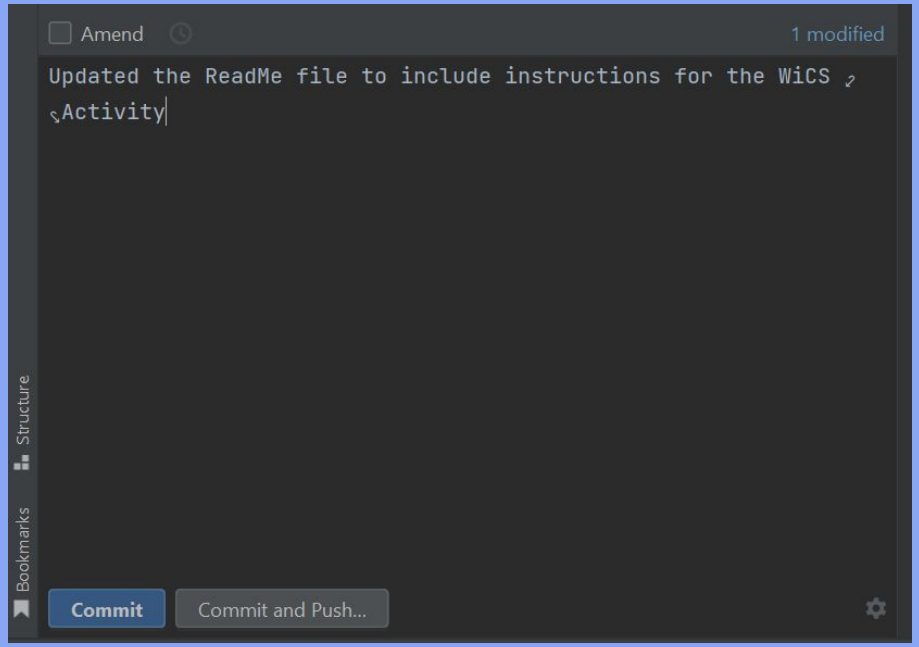
# What is a Commit

The files you changed. Make sure to check that the correct files were changed!!!

The commit message box. Make sure to use descriptive language for every commit.

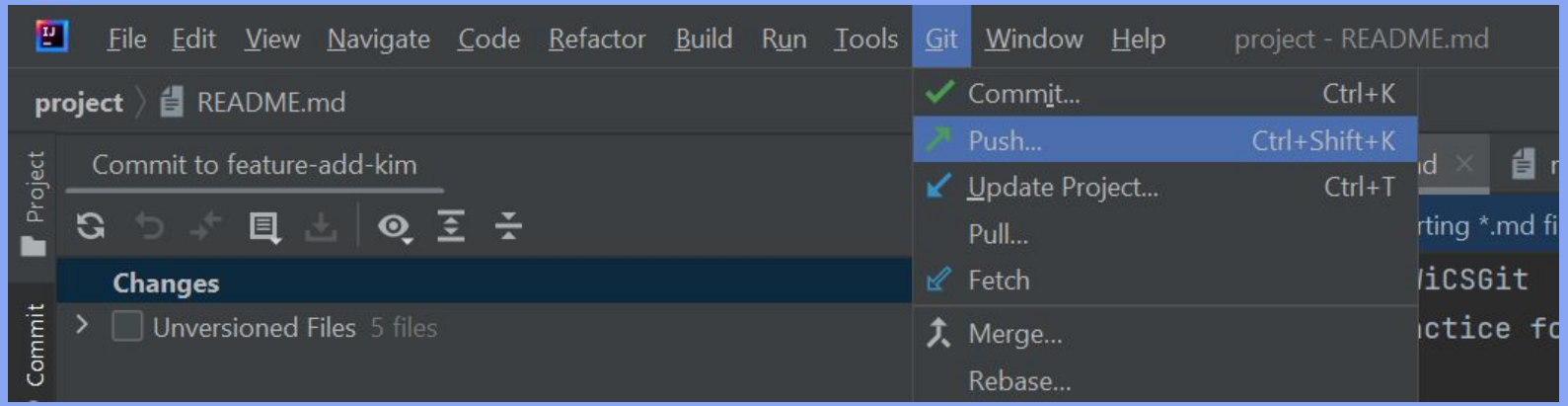


# What is a Commit

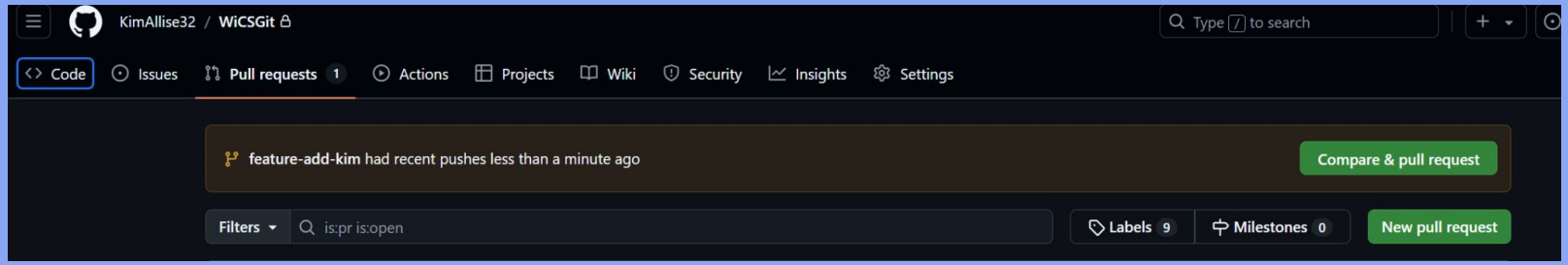


# What is a Push

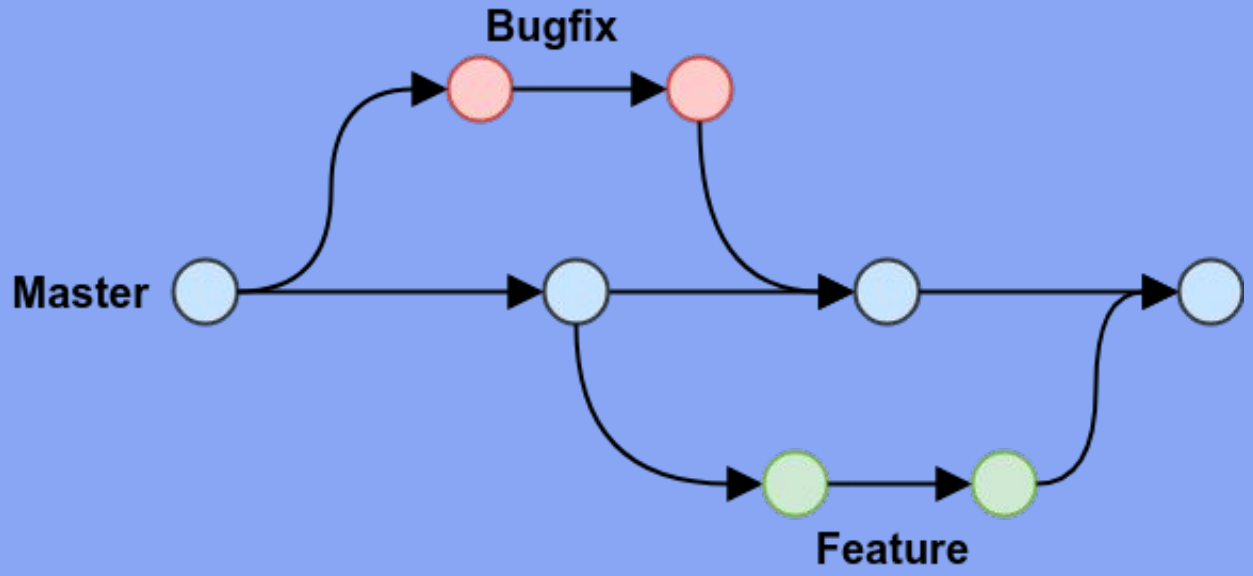
# What is a Push



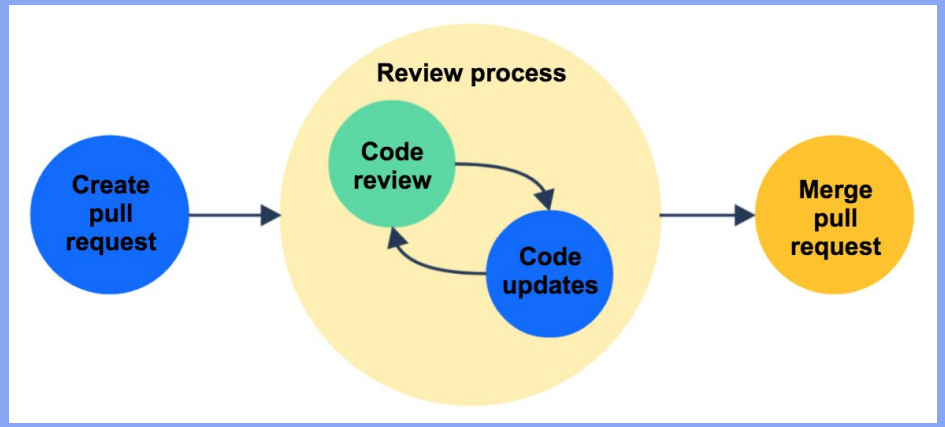
# What is a Push



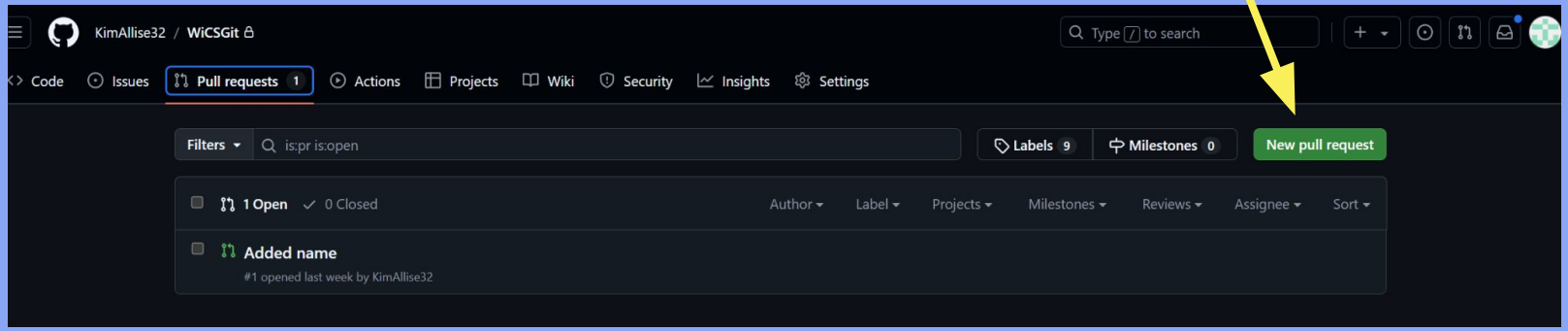
# What is Merging



# What are Pull Requests



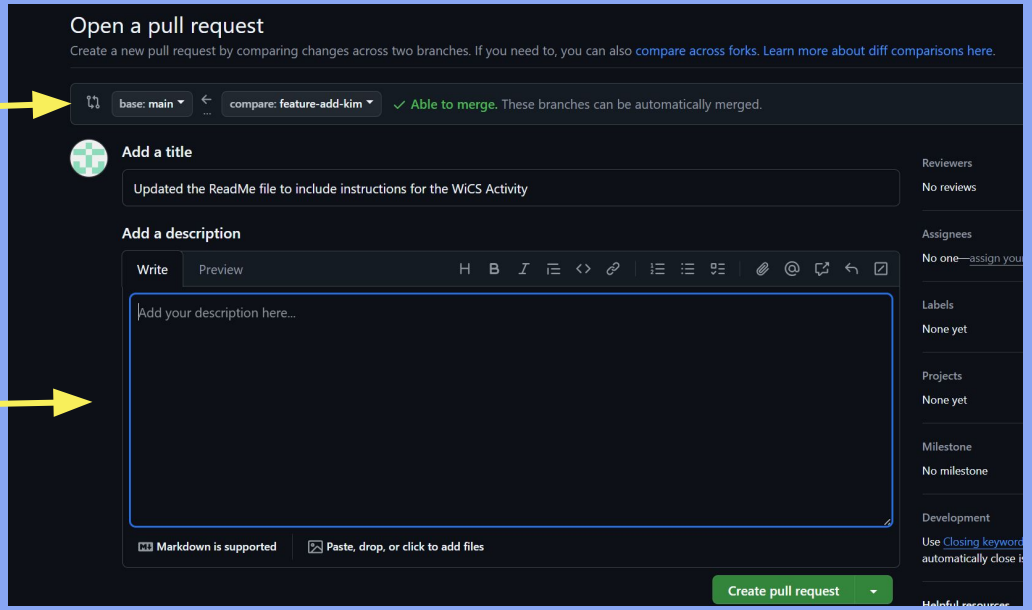
# What are Pull Requests



# What are Pull Requests

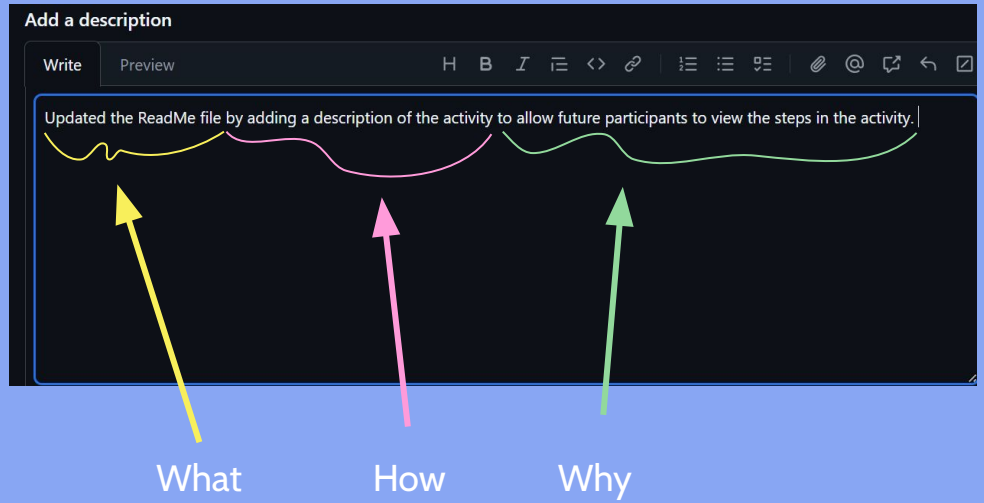
Make sure you are merging changes into the correct branch

Add a good description here too!



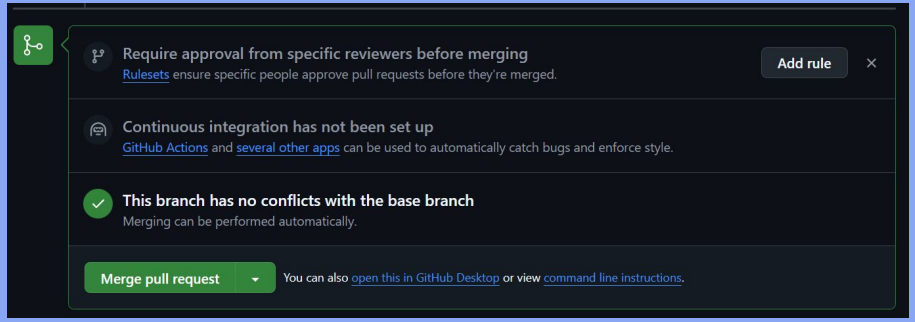
# Good PR Descriptions

→ What?
→ Why?
→ How?
→ Testing?
→ Screenshots (optional)
→ Additional Info (optional)

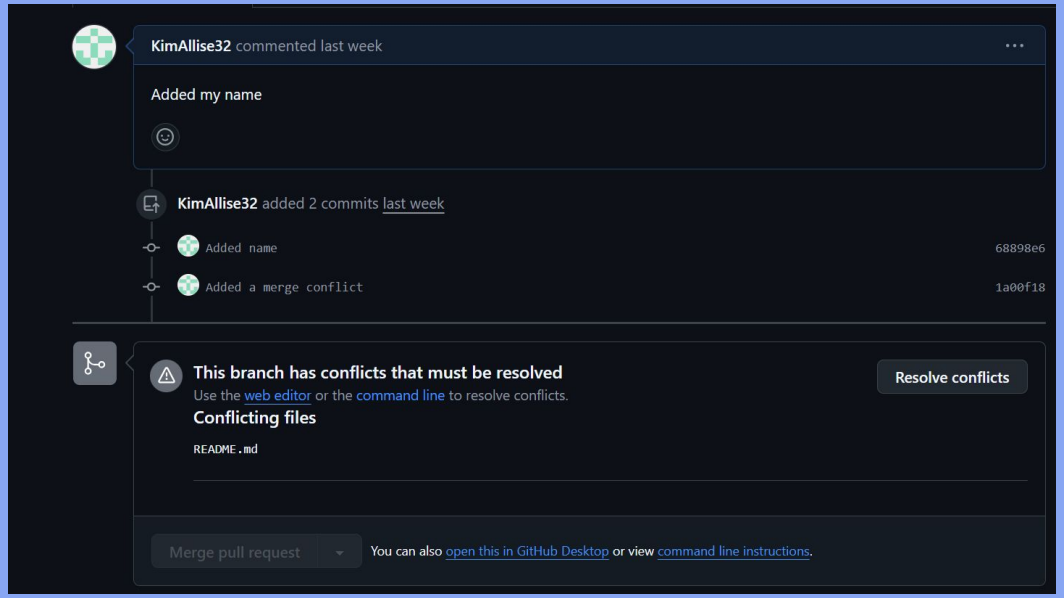


# Reviewing PRs

I would recommend on big projects downloading the branch and making sure it works before allowing the PR through



# Watch out for Merge Conflicts



# Watch out for Merge Conflicts

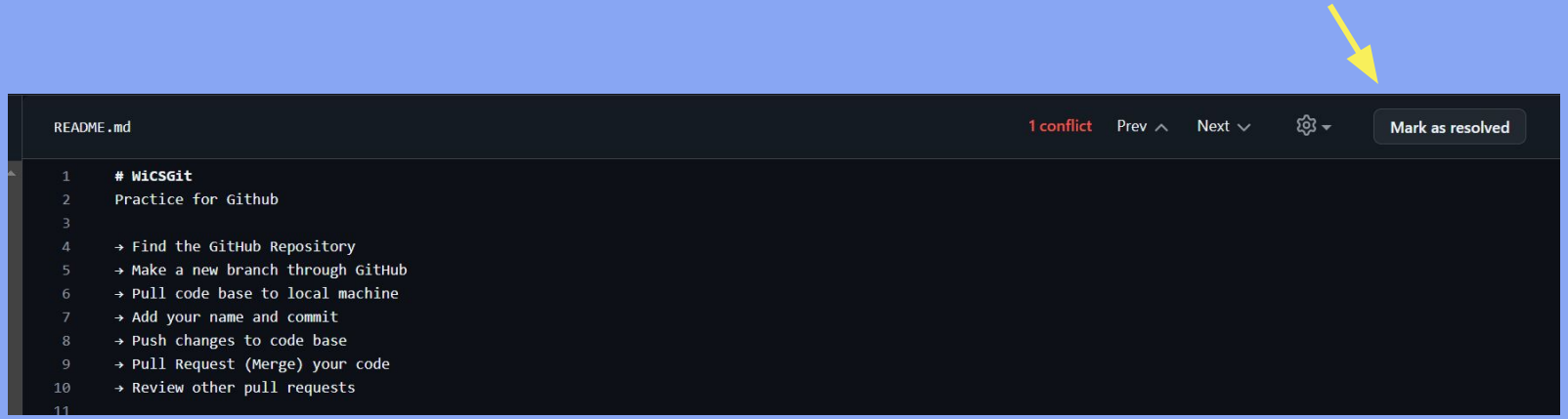
My branch version

Main Version

README.md

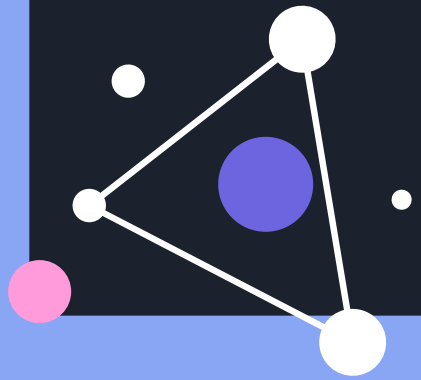
```
1   # WiCSGit
2   Practice for Github
3
4   <<<<<<< kimbrnach
5   MERGE CONFLICT
6   =======
7   → Find the GitHub Repository
8   → Make a new branch through GitHub
9   → Pull code base to local machine
10  → Add your name and commit
11  → Push changes to code base
12  → Pull Request (Merge) your code
13  → Review other pull requests
14  >>>>>>> main
15
```

# Watch out for Merge Conflicts



bit.ly/wicsgit

# Activity!



→ Find the GitHub Repository
→ Make a new **branch** through GitHub
→ **Pull** code base to local machine
→ Checkout your branch
→ Add your name and **commit**
→ **Push** changes to code base
→ **Pull Request** (Merge) your code
→ Review other pull requests

Watch out for merge conflicts!!!

# Resources

- WiCS Git Cheat Sheet
- https://ohshitgit.com/
- Atlassian Git Cheat Sheet
- https://learngitbranching.js.org/
- https://blog.codeanalogies.com/2019/05/26/prs-explained/
- https://git-school.github.io/visualizing-git/