Git / GitHub Workshop

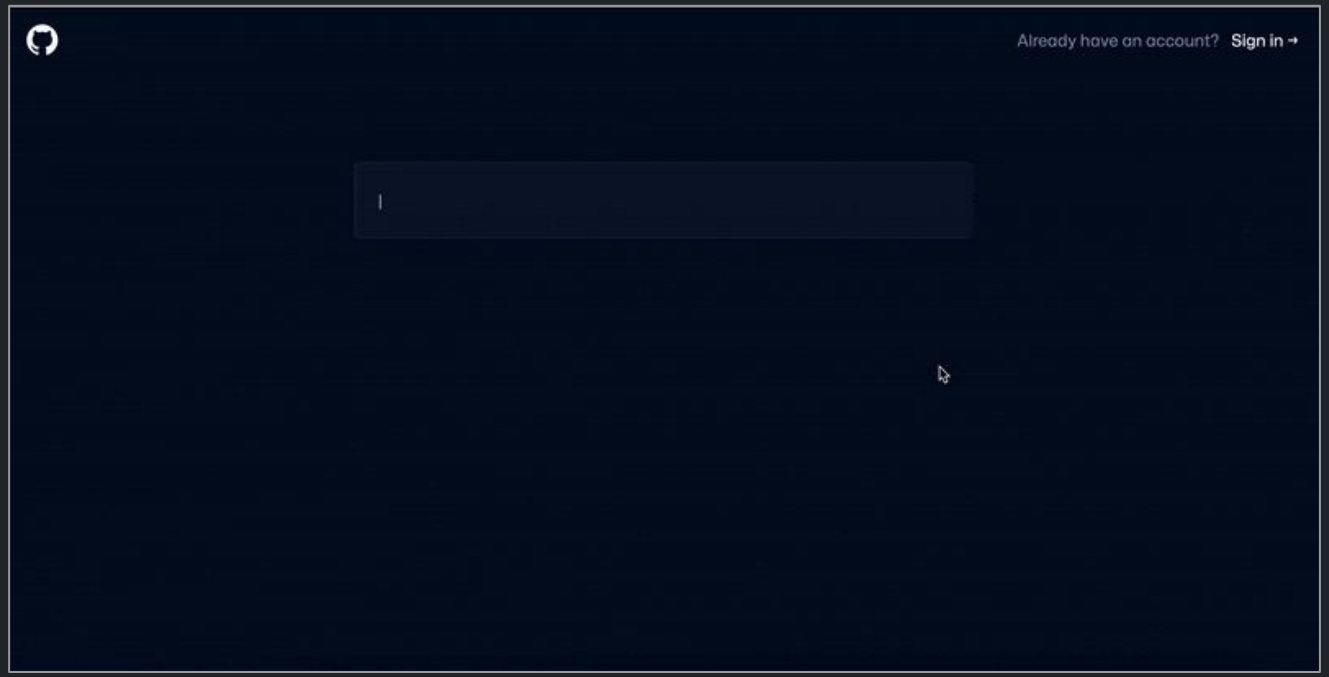
How To HackRPI

How to Set Up *GitHub*

Make a GitHub account

Use your personal email!

https://github.com/

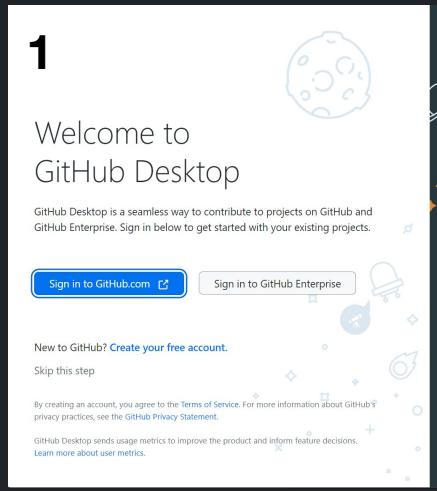


How to Install

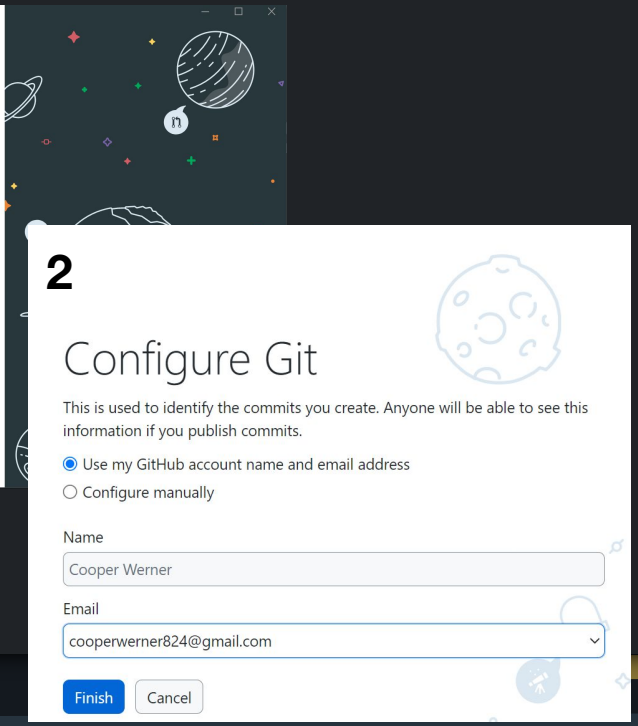
GitHub Desktop

A client to interact with GitHub from our computer.

https://desktop.github.com



^Download the installer from the website and use the installer to install the software.



Introduction

What are Git and GitHub?

- Git
  - Track changes to the codebase
  - Allows you to easily reverse changes
  - Improves coordination between developers
- GitHub
  - Storing our code on the web
  - Enables many Devs on one project
  - Makes deploying code easier

How do we use Git and GitHub

- A variety of commands and concepts
  - Commits
  - Branches
  - Pushes / Pulls
  - Forks
  - Pull Requests / Merges

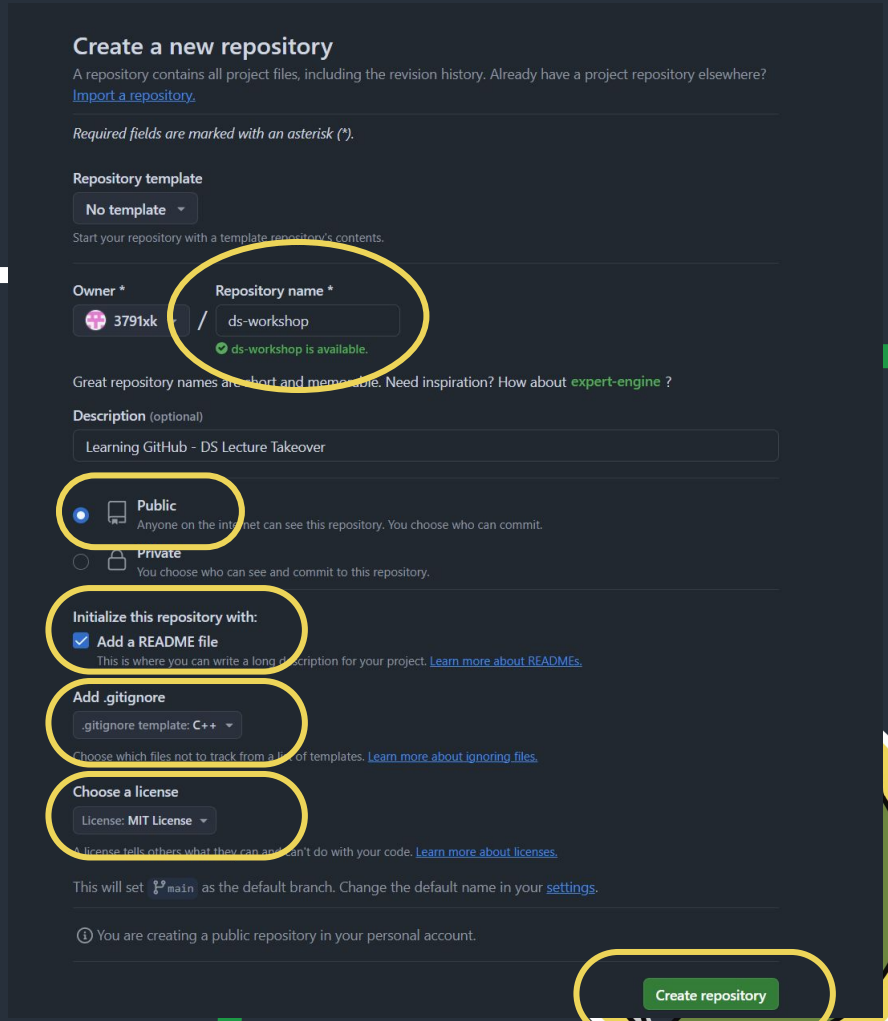
- We will go more in depth with these later but let's get setup first.

Using Git / GitHub

Create a Repository

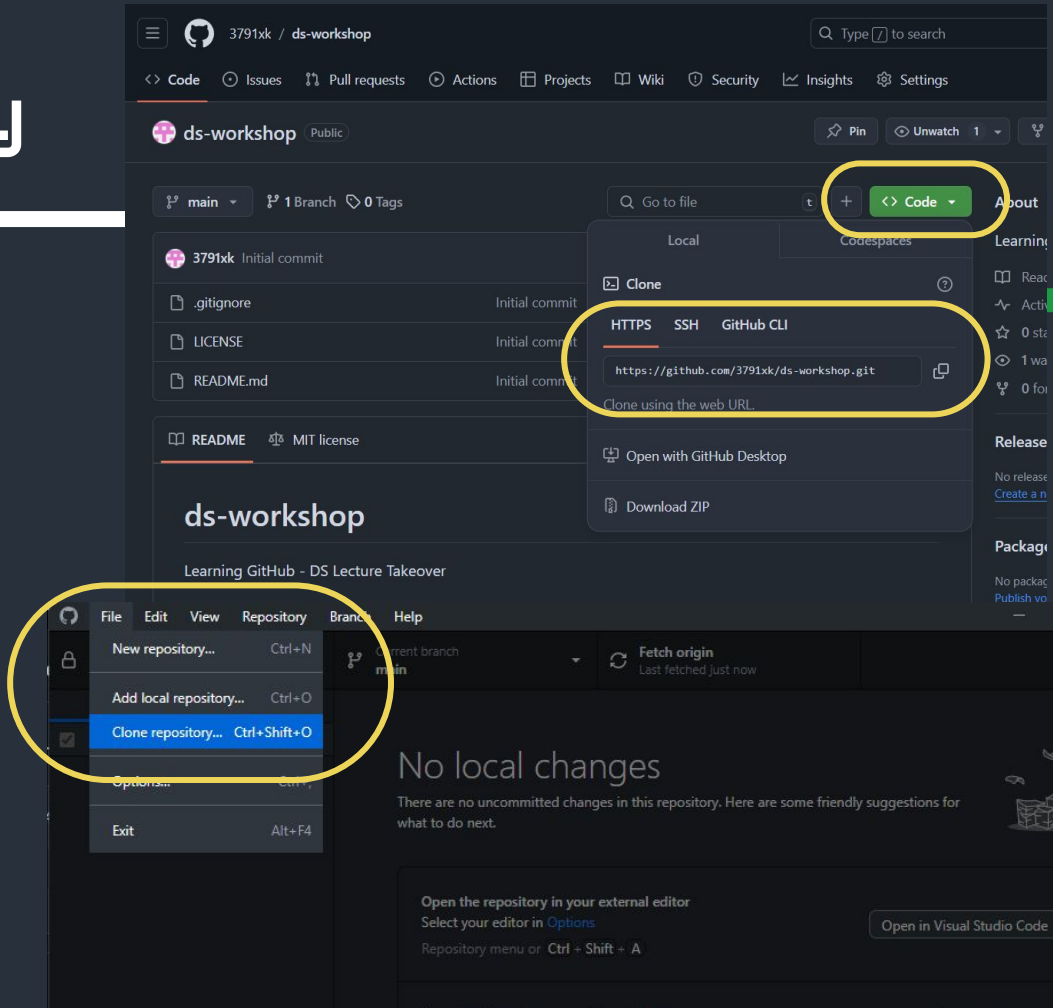
Repositories are a place to store code for a project

1. Go to https://github.com
2. Click the green “New” button.
3. And follow the prompts to create a new repository.
4. Make sure Repository is public



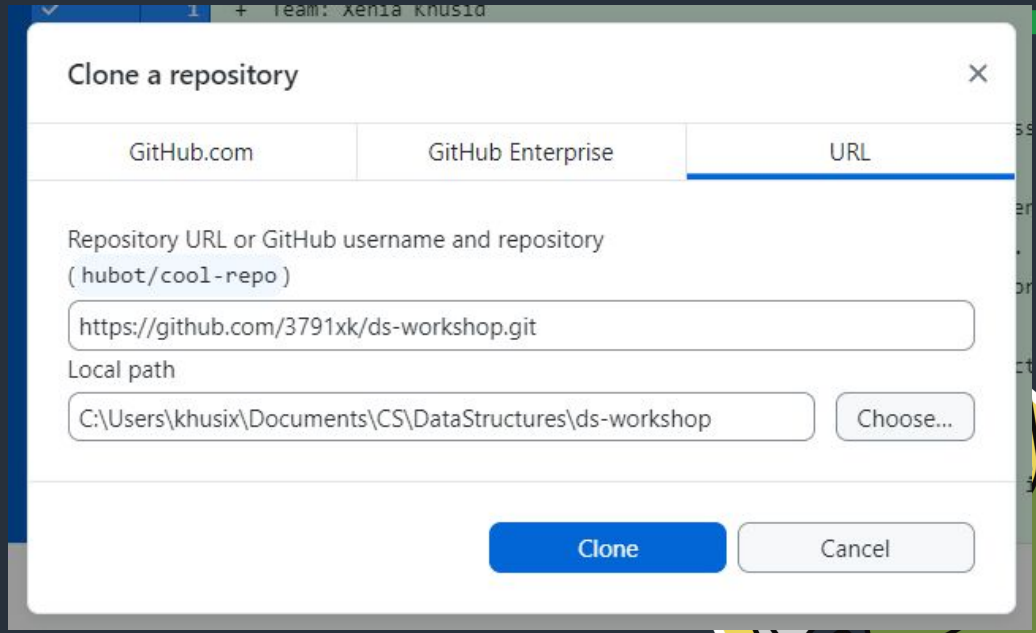
Clone a Repository

1. Creating a copy of the repository on our computer
2. On github.com click the green code button in your repository, copy the link to your repository.
3. Open GitHub Desktop and click file, then Clone repository.



Clone a repository continued:

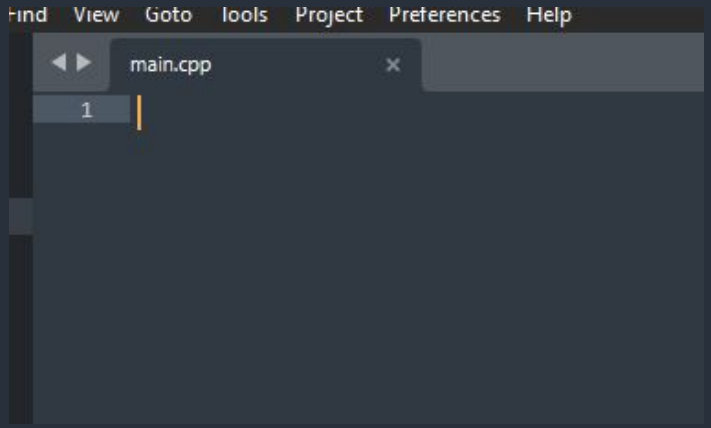
1. Paste the link from your github repository into the Repository URL field
2. Click the blue clone button

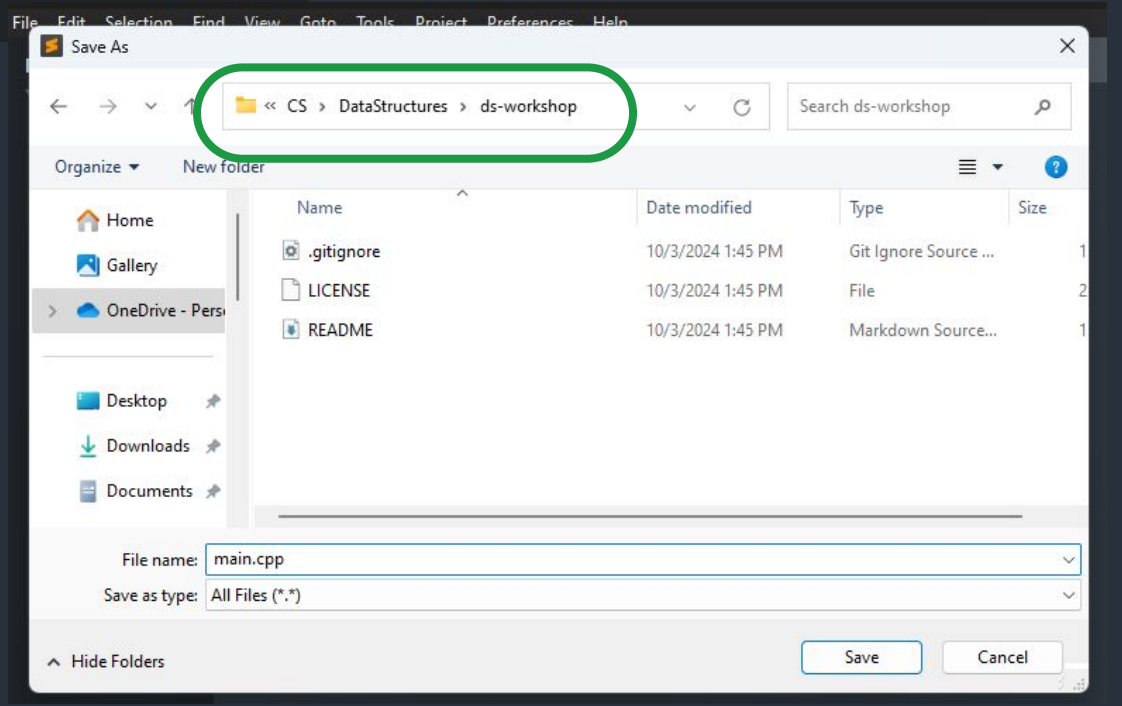


Start our code

H A C K

Make a .cpp file in that folder





Write a program:

- Accept command line arguments
- Put each object into a std::list of int values
- Use an iterator to print each value in the list

```
khusix@LAPTOP-NGACK2CL:/mnt/c/Users/khusix/Documents/CS/DataStructures/ds-workshop$ g++ main.cpp -g -Wall -Wextra
khusix@LAPTOP-NGACK2CL:/mnt/c/Users/khusix/Documents/CS/DataStructures/ds-workshop$ ./a.out 1 2 3 5 8
1
2
3
5
8
khusix@LAPTOP-NGACK2CL:/mnt/c/Users/khusix/Documents/CS/DataStructures/ds-workshop$
```

https://github.com/3791xk/ds-workshop

Example:

Will look something like this:

Or copy from my repo!

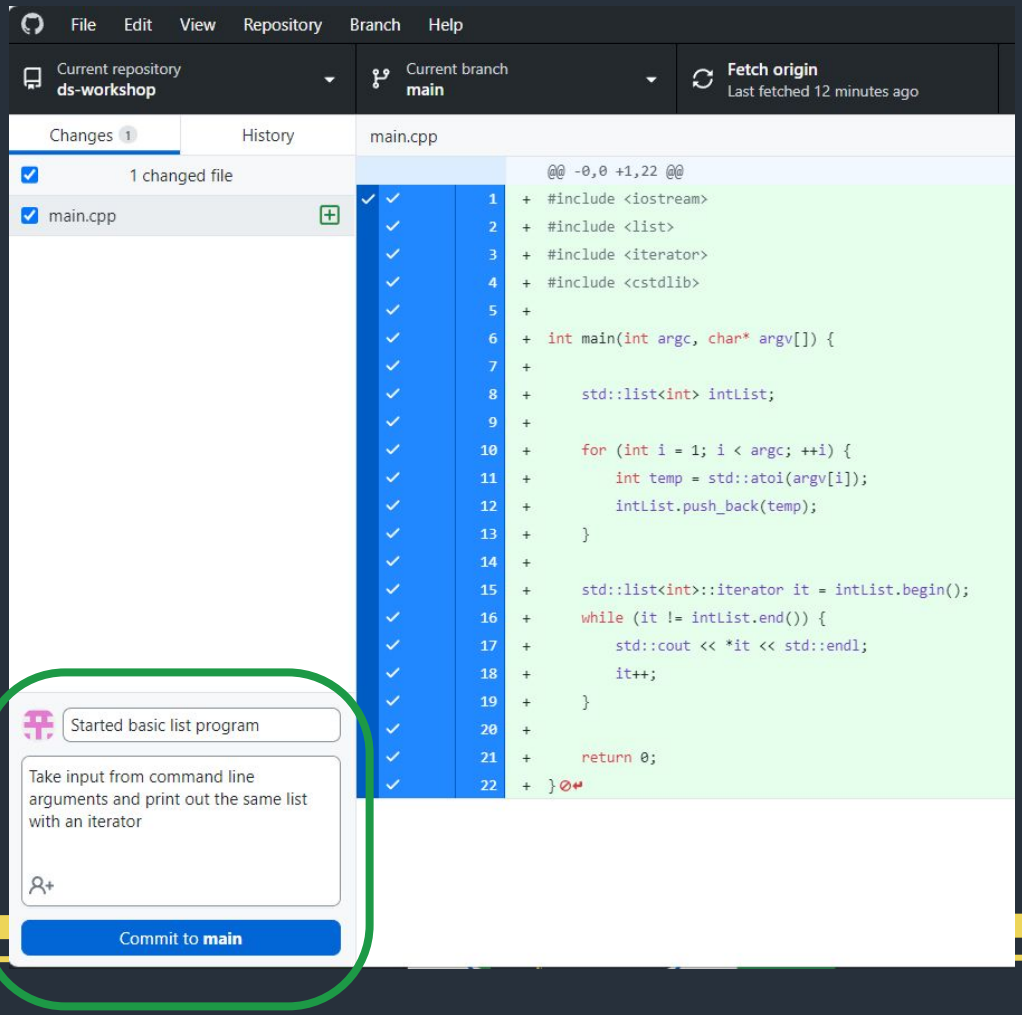
https://github.com/3791xk/ds-workshop

main.cpp

```cpp
#include <iostream>
#include <list>
#include <iterator>
#include <cstdlib>

int main(int argc, char* argv[]) {

    std::list<int> intList;

    for (int i = 1; i < argc; ++i) {
        int temp = std::atoi(argv[i]);
        intList.push_back(temp);
    }

    std::list<int>::iterator it = intList.begin();
    while (it != intList.end()) {
        std::cout << *it << std::endl;
        it++;
    }

    return 0;
}
```

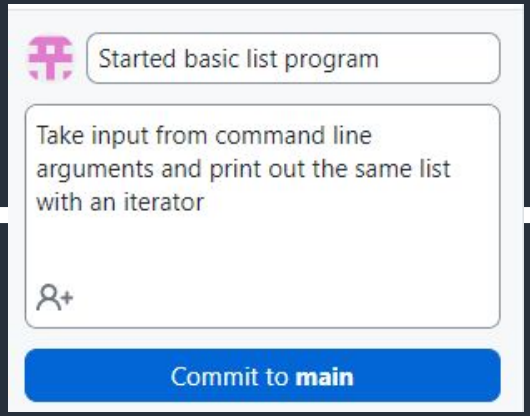
See your changes in GitHub Desktop

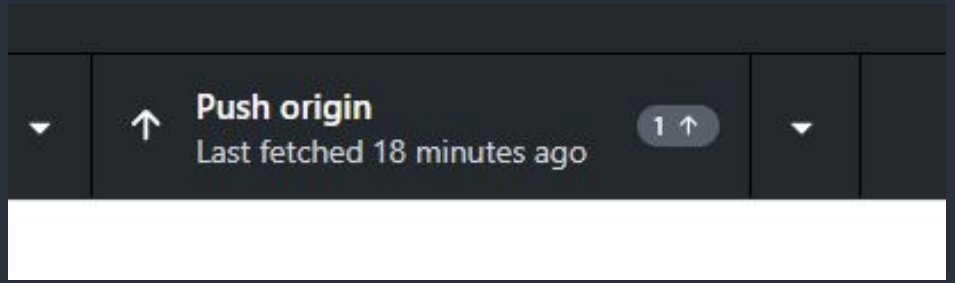
- Notice “Current branch” is “main”
- One changed file is the one you made and edited
- Write a **name** and **description**
- This is a **“staged commit”**



Commit to main

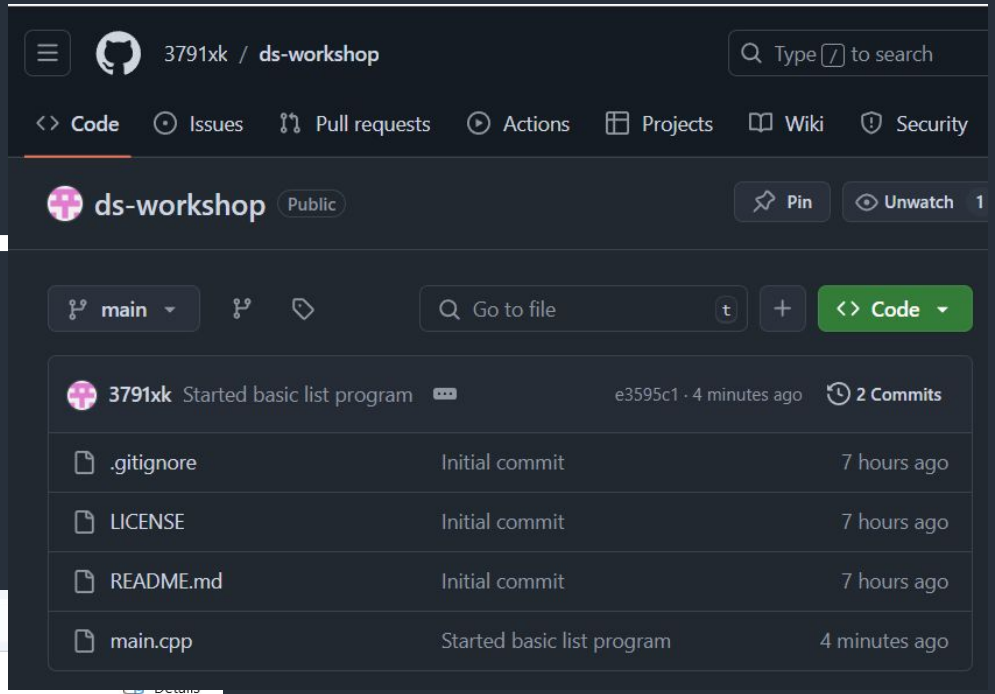
- Click the blue commit button
- Then click push to origin
- Your change will now be visible in your GitHub repository
- We'll find out why this isn't a best practice soon

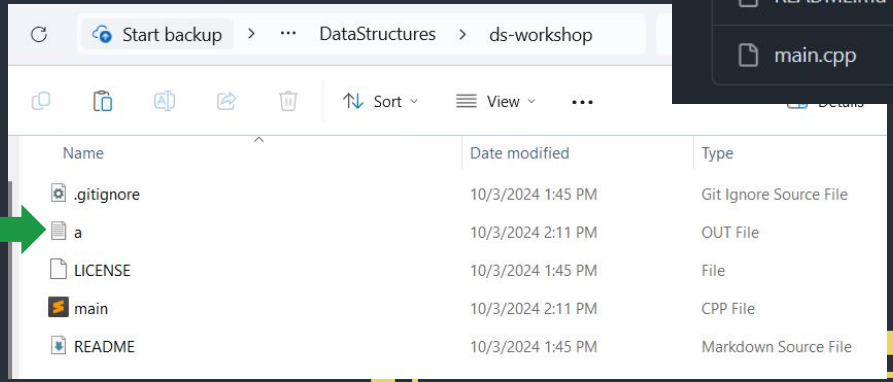




Note the differences

- Your **.gitignore** file keeps compiled output files out of your repository

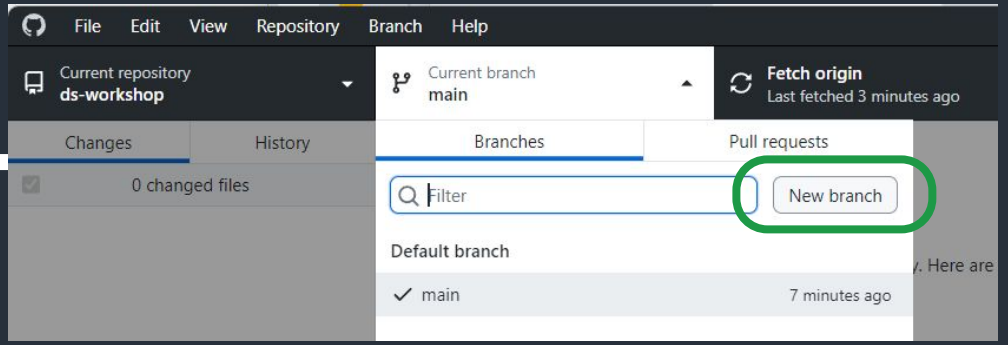


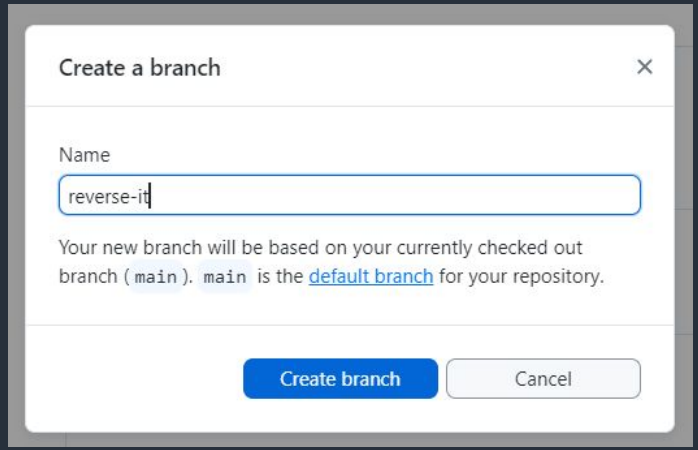


Branching

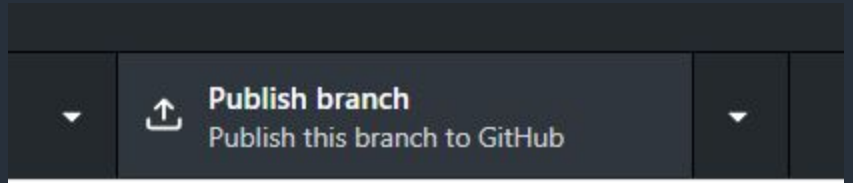
Make a branch

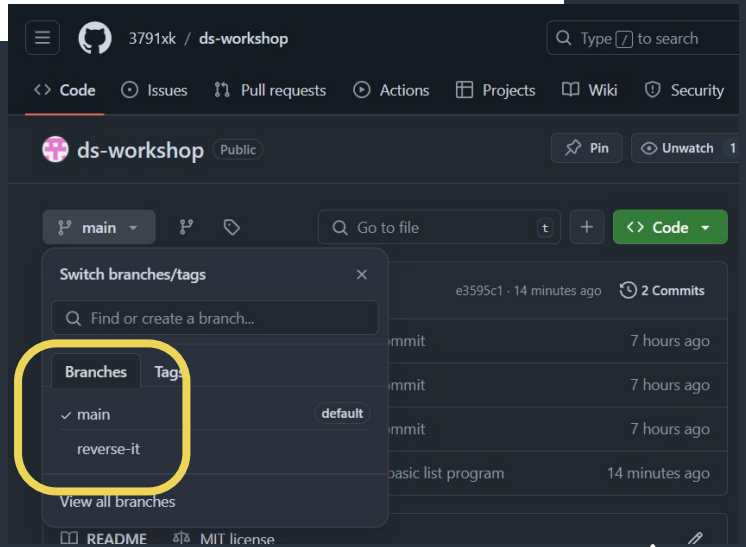
- A branch allows you to split off a version of the code to test things / add a feature
- ALWAYS pull repository first before creating
- Be specific when naming branch





See it on GitHub

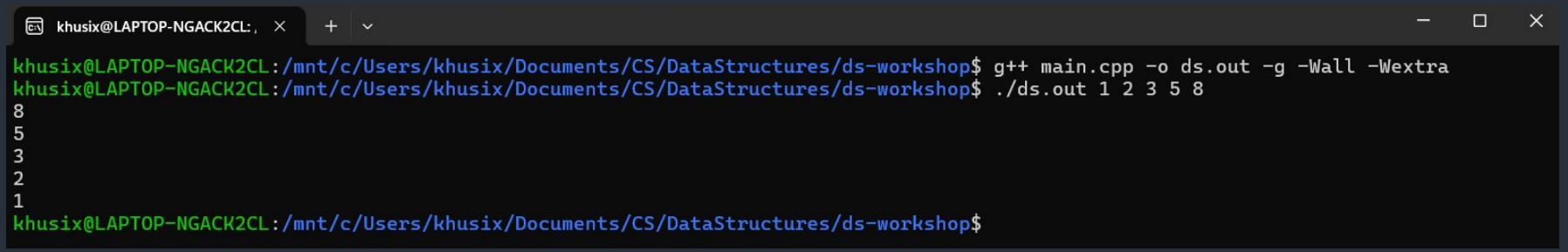




- Click “Publish branch”
- You can commit to a branch as much as you want locally, but it’s good to push the branch to the repository to save it and for collaboration
- Then look at the repository page, right now everything is up to date with main

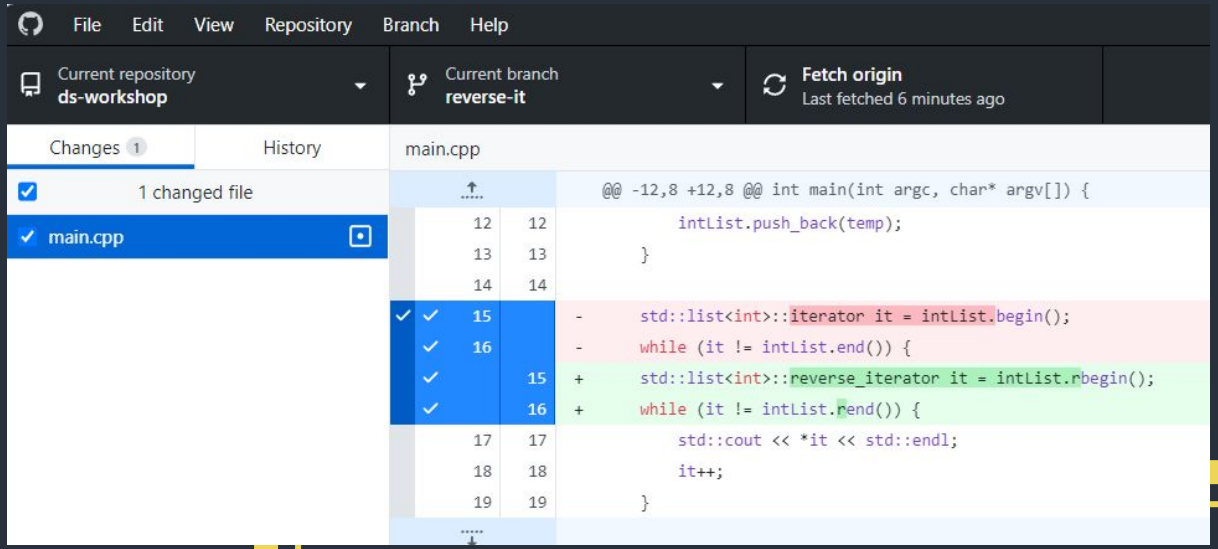
Reverse the iteration

- Swap the iterator out for a reverse iterator to iterate through the list backwards
- (Or just start at the end with the original iterator)



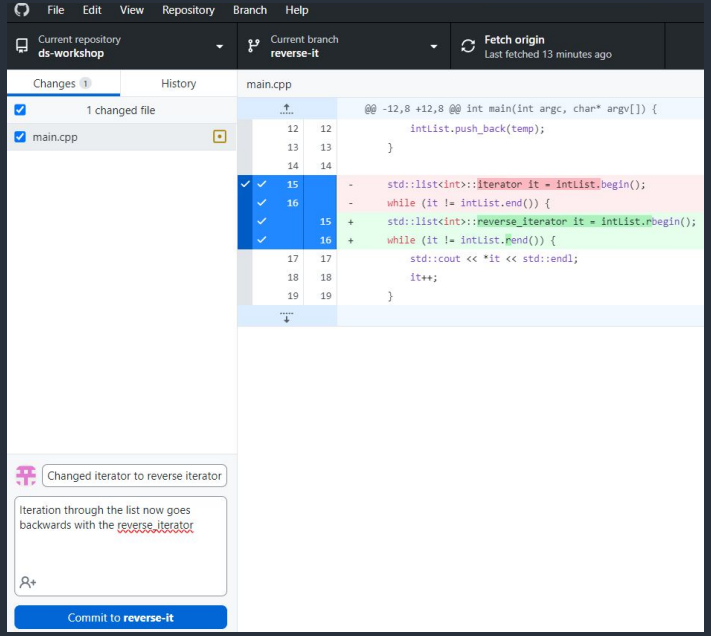
See the changes in GitHub Desktop

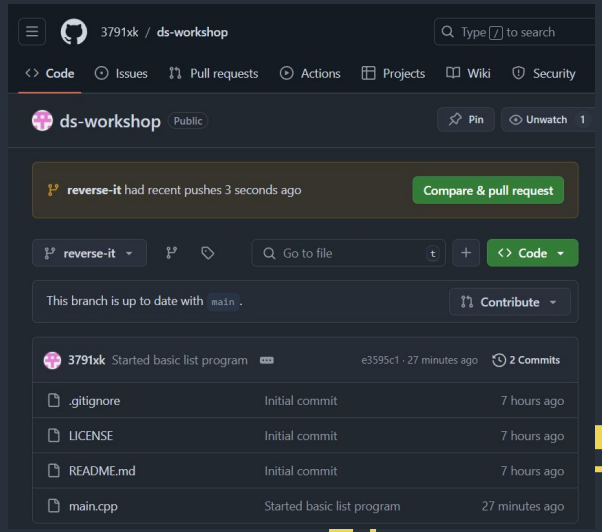
- Pretty straightforward, added parts are in green, removed parts are in red



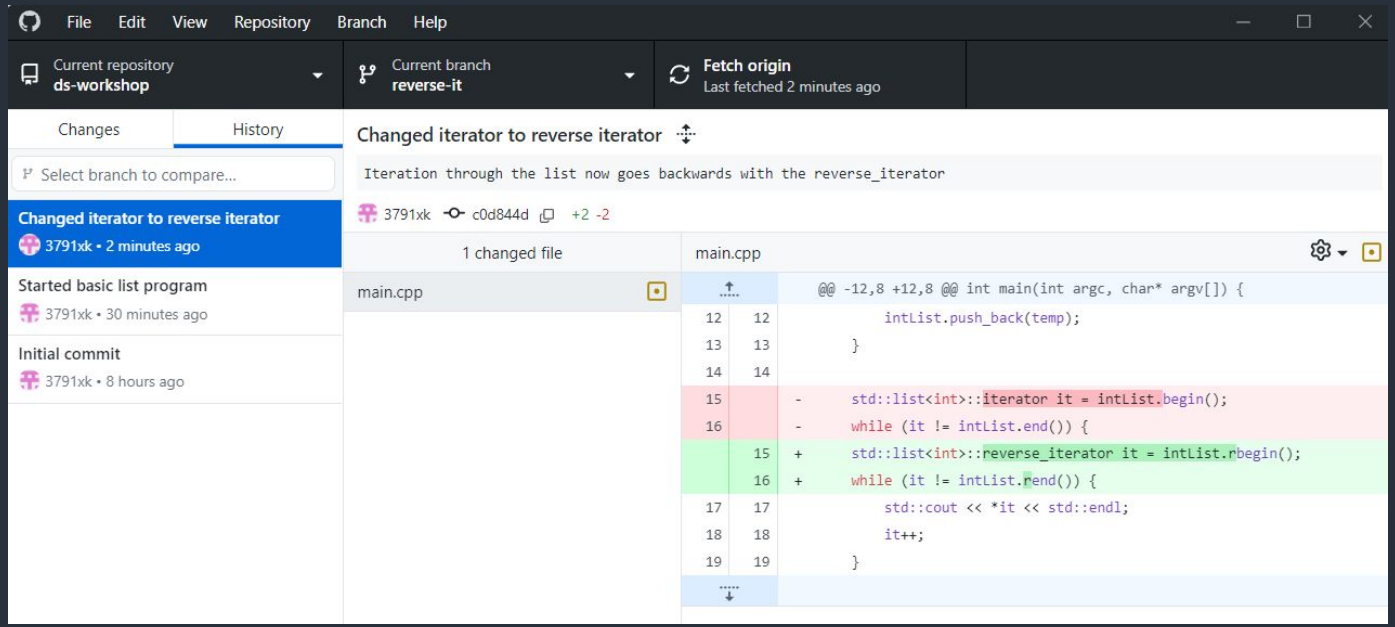
Commit your new changes

1. Write a commit name + description
2. Commit the changes to your branch
3. Push the changes to origin





See commit history

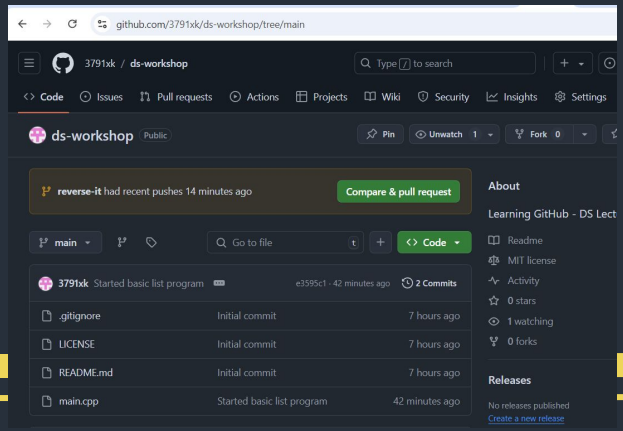


Pull requests

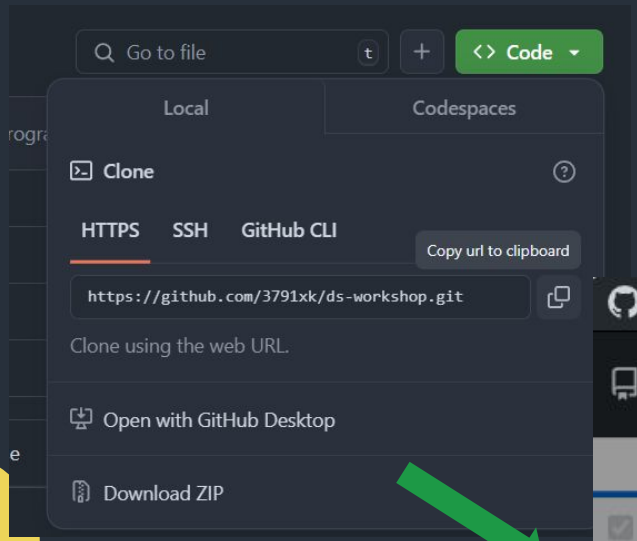
Collaborating With GitHub

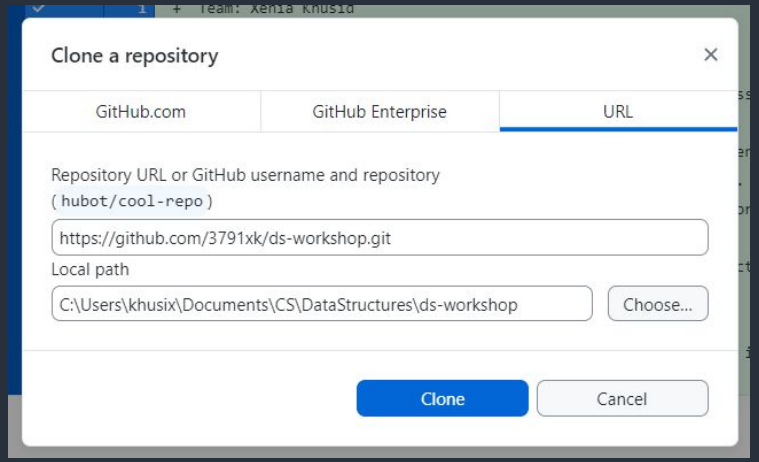
- Introduce yourself with the person sitting next to you
  - Pairs or groups
  - Pass the repositories down
- We going to be collaborating on each other's websites
- Go to their GitHub repository

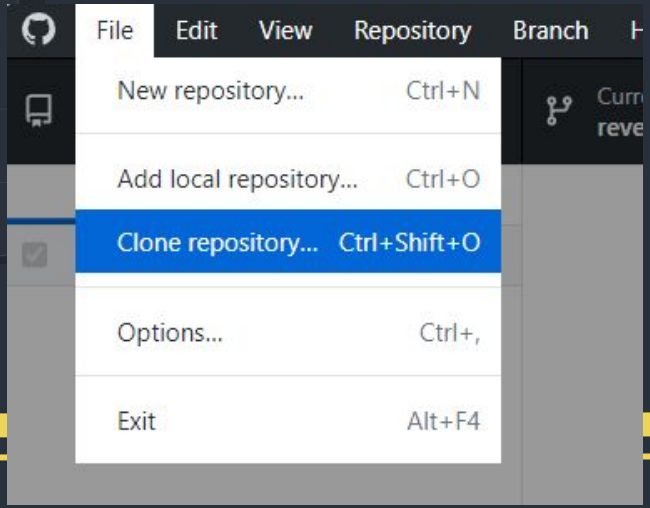
https://github.com/3791xk/ds-workshop



Clone their repository

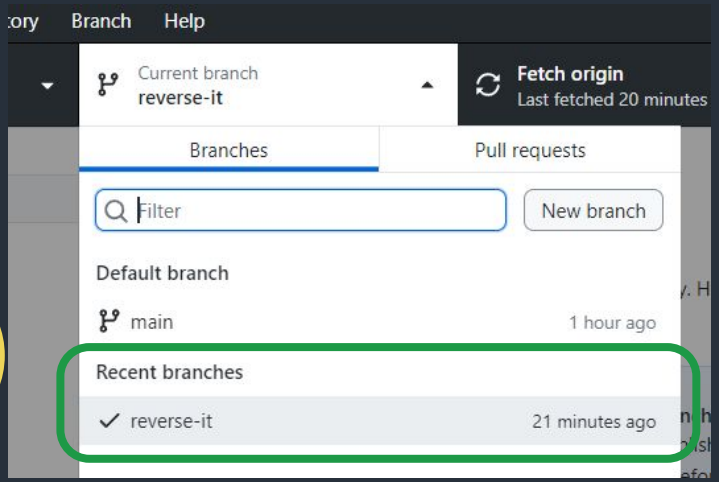


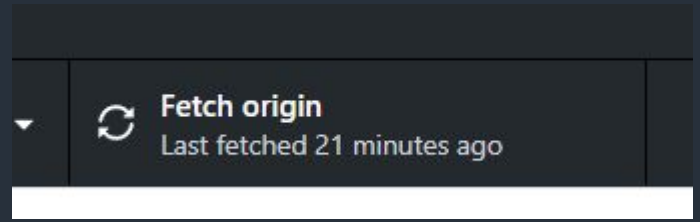




Switch to the reverse branch

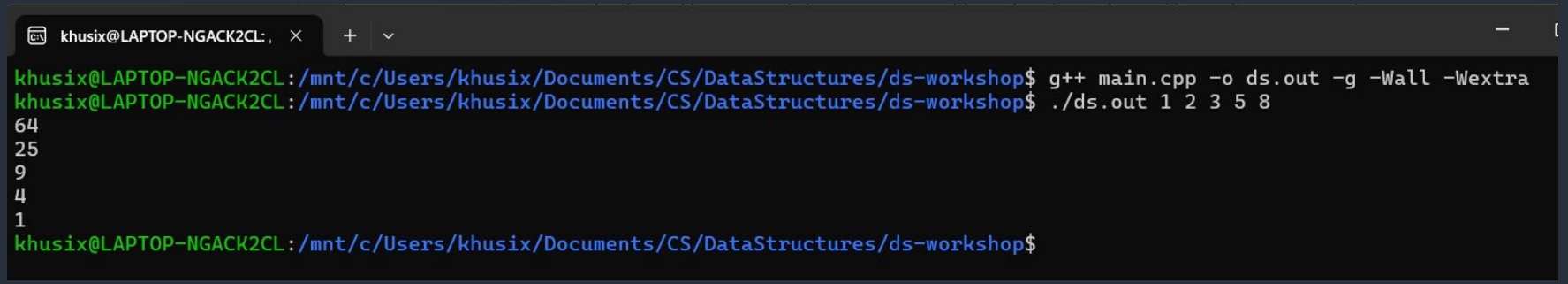
- Switch to their reverse branch
- Find the file in file explorer and open it in whatever IDE you use





Print out the square of the int

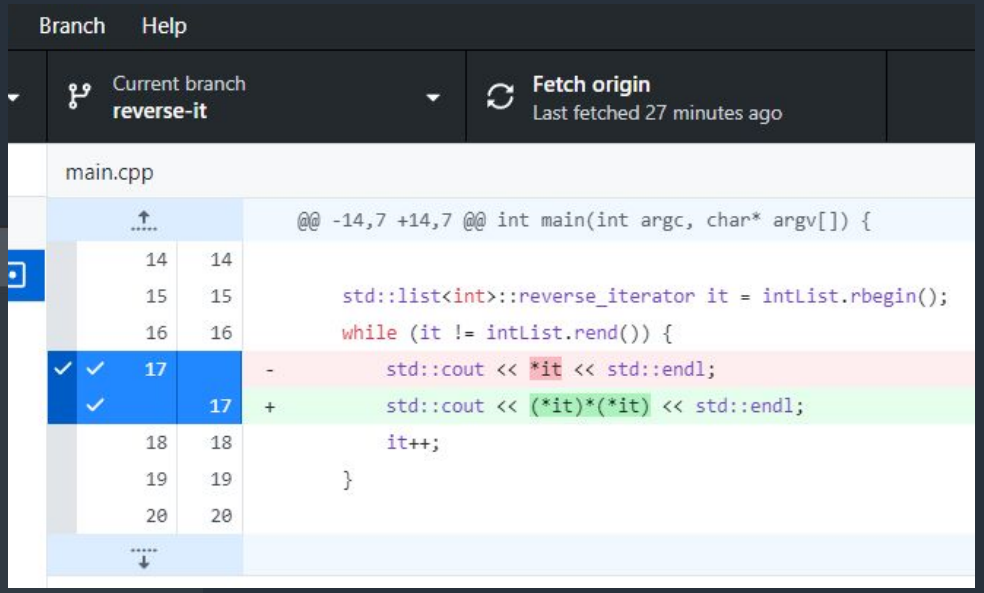
- Open up their code and make it print the squares of the integers instead of the integers



See the changes

main.cpp

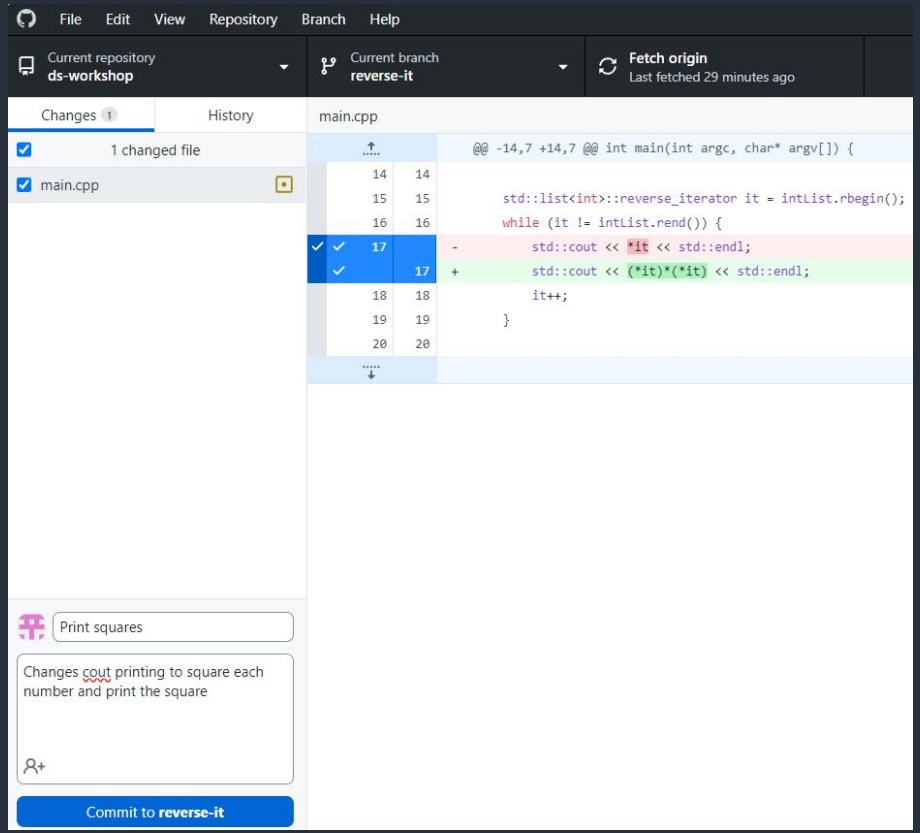
```cpp
#include <iostream>
#include <list>
#include <iterator>
#include <cstdlib>

int main(int argc, char* argv[]) {

    std::list<int> intList;

    for (int i = 1; i < argc; ++i) {
        int temp = std::atoi(argv[i]);
        intList.push_back(temp);
    }

    std::list<int>::reverse_iterator it = intList.rbegin();
    while (it != intList.rend()) {
        std::cout << (*it)*(*it) << std::endl;
        it++;
    }

    return 0;
}
```



Commit your changes

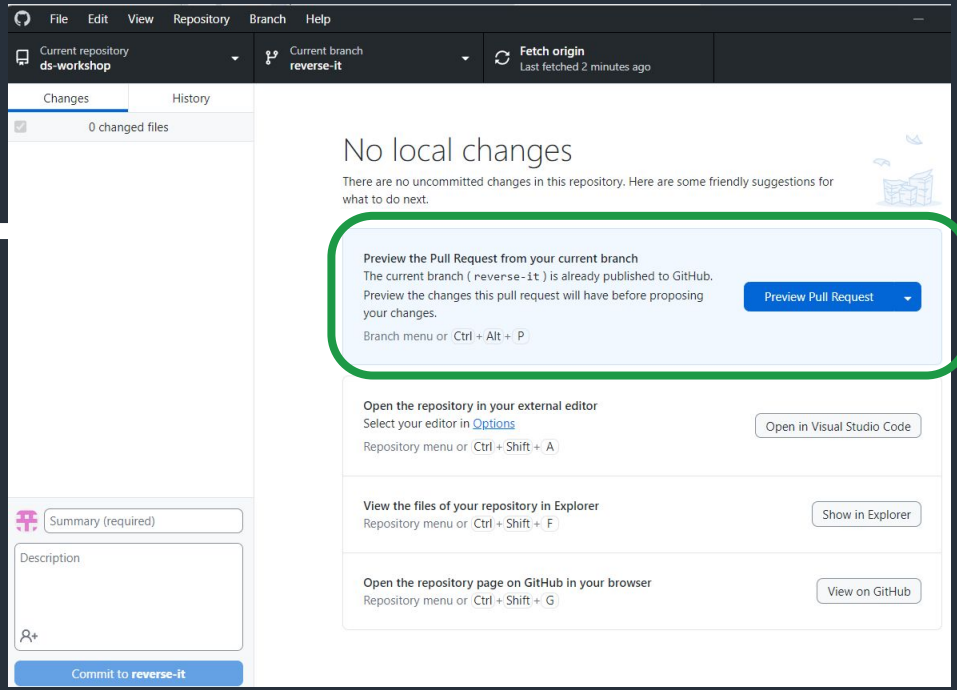
We hope you know how by now

1. Write a commit name + description
2. Commit the changes to your branch
3. Push the changes to origin

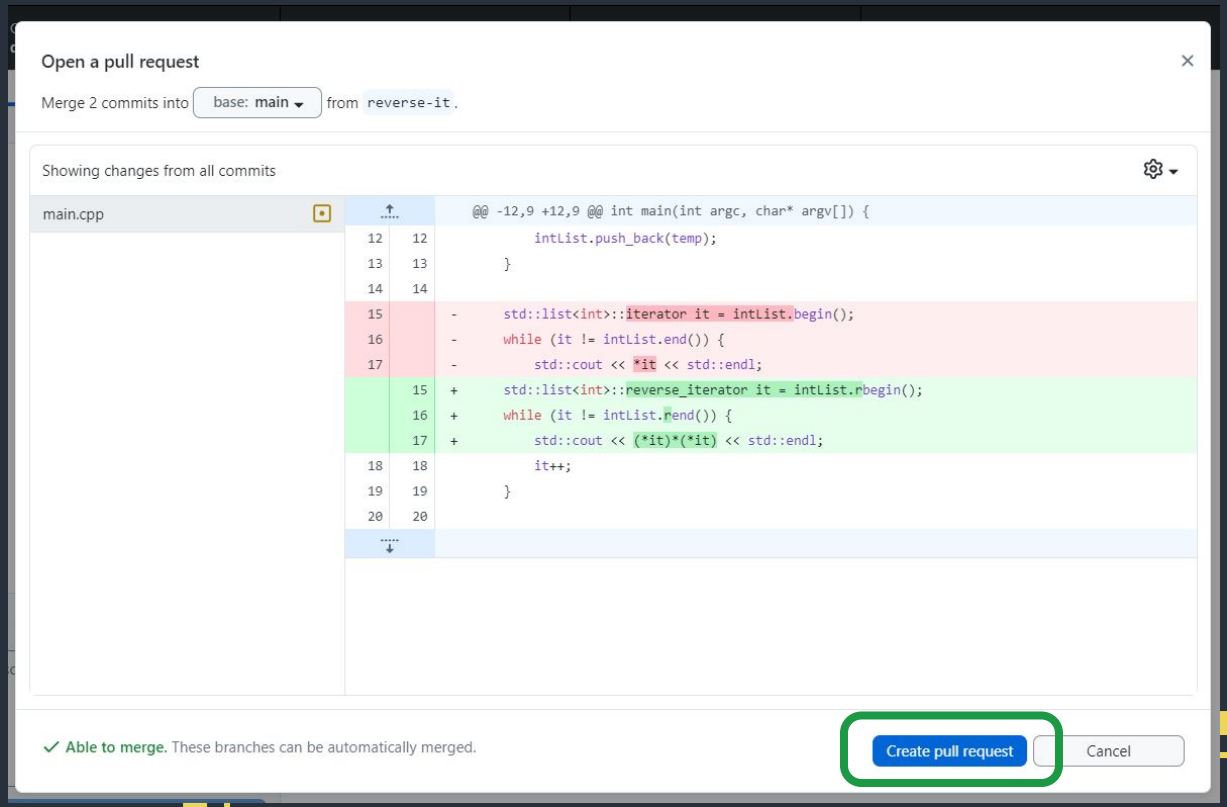


Pull Requests

- Allows your team to view your branch on github for review and merging
- Pull Requests show up after your have Pushed Commits and will show up in blue
- Select Create Pull Request to create a pull request

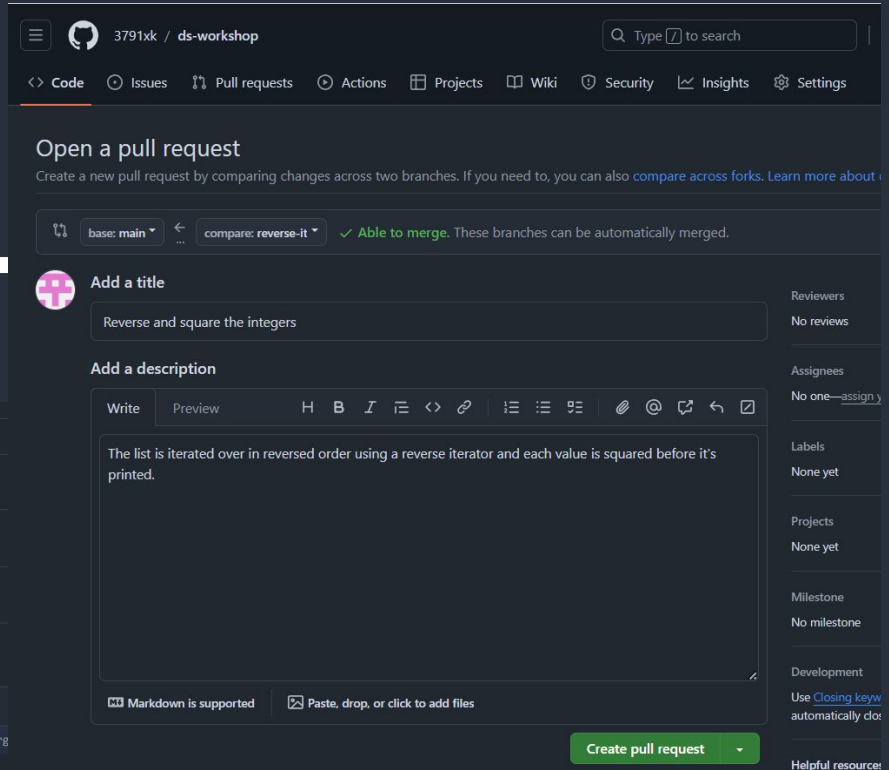


Click “Create pull request”



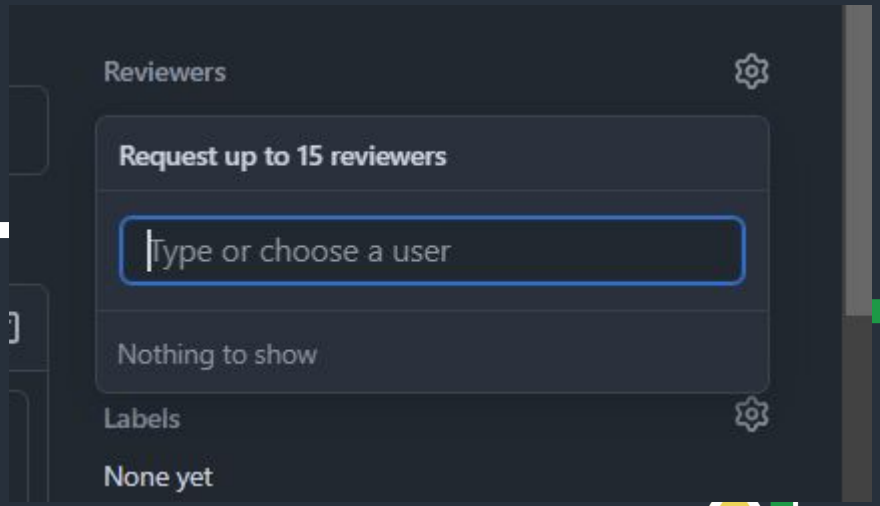
Creating a pull request

- Write a title and description
- See the commits
- See changes from the commits
- This is a request to have your changed merged



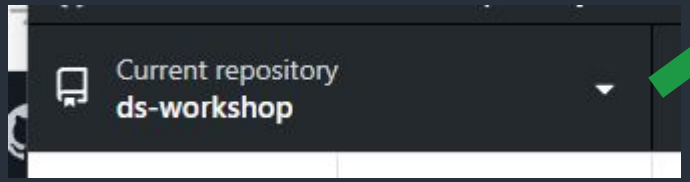
Request a review

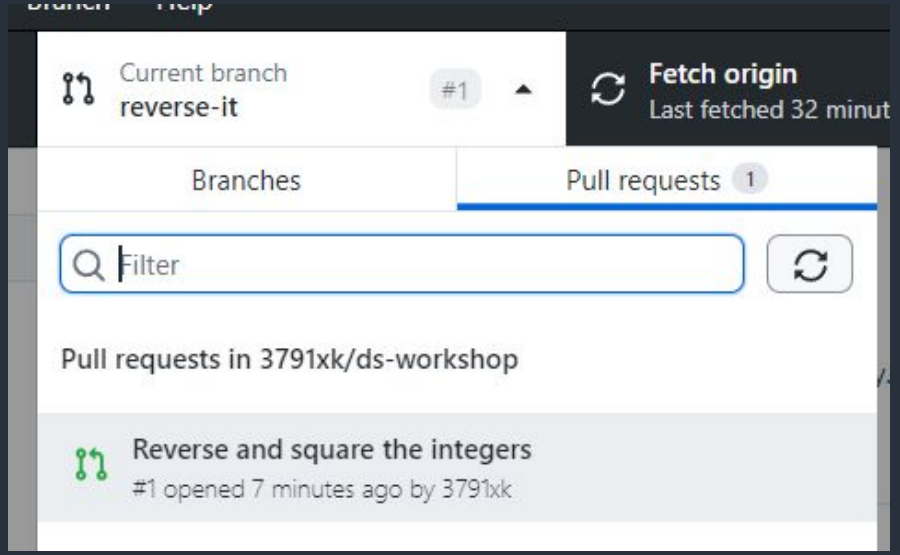
- Request the owners username to review
- Your partner can make comments on, request changes, close, or approve and merge a pull request.
- Click on the ‘Files Changed’ button to review the PR



Switch to your repository on GitHub Desktop

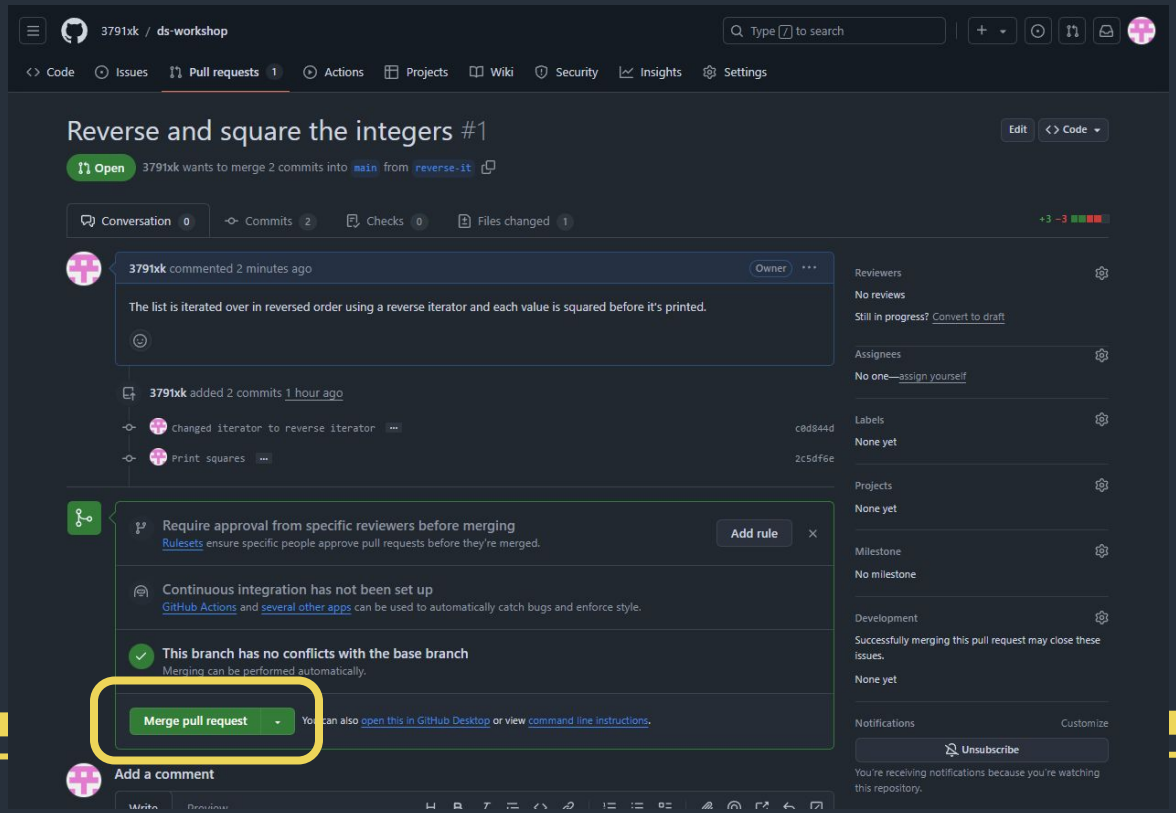
- Switch “Current repository” to your own repo
- Switch to the “Pull request” branch and fetch origin





Review the Pull Request

- Run their code to make sure it actually works
- Review on GitHub
- Optionally add comments
- Merge when ready, it'll turn purple



Questions/ Comments?

HackRPI 2024

November 9-10th

https://hackrpi.com/

How to HackRPI

October 4th

Instagram: @hack.rpi

Linkedin: @HackRPI

Website: hackrpi.com