Intro to GitHub

By: Aditya Makineni, Nicholas Burczyk

What is Git?

A version control system that:

- Tracks changes made to your files over time
- If you mess up (or something breaks)...it lets you go back to the previous version

What is a repository?

- A repository is just like a folder on your computer, where you can store your files, other folders, and code that you want.
- Repositories also store the full Git history of your project.

Key Takeaway: *Git manages you changes, and a repository stores your belongings and their evolution history*

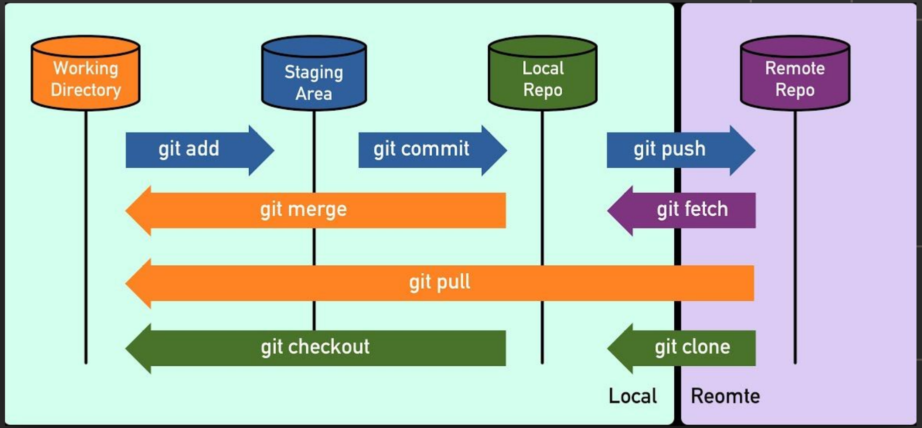
What is GitHub

Git + Repository = Git Repository (Git Repo)

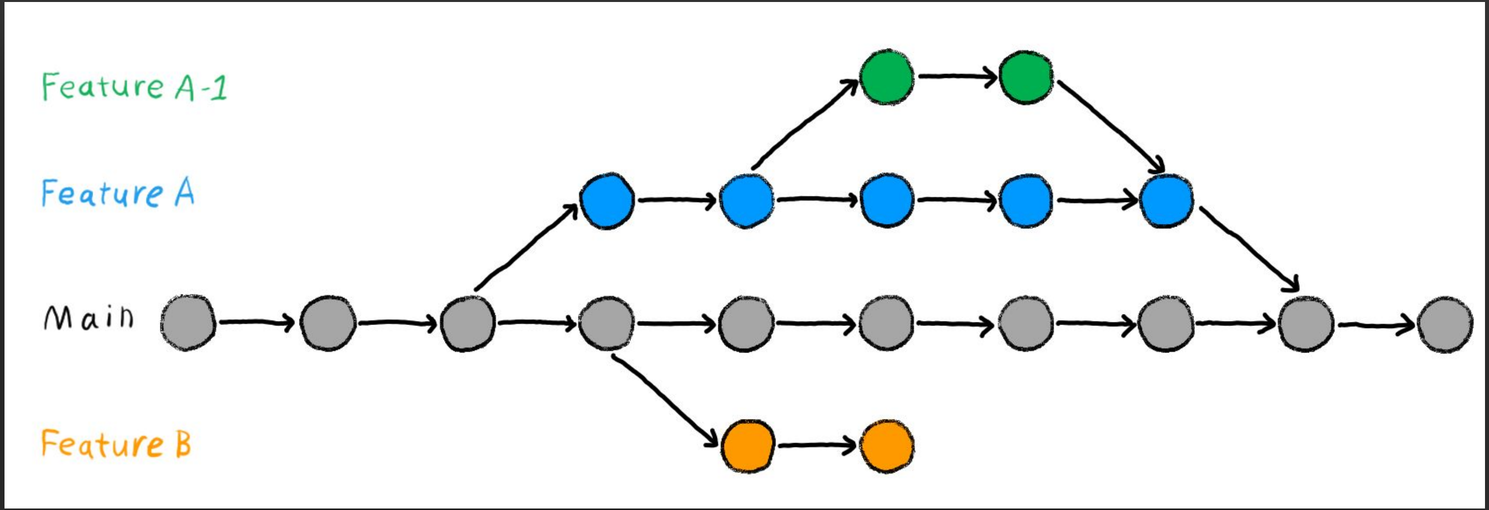
GitHub = **A platform for storing and sharing your git repositories!**

*There are two different types of repositories:

- A local repository
- A remote repository



GitHub Branches

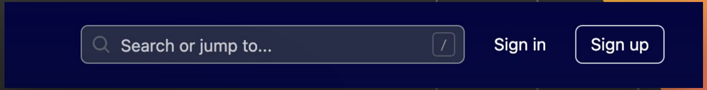


Key takeaway: *A copy of your project where you can try new ideas without affecting you main code.*

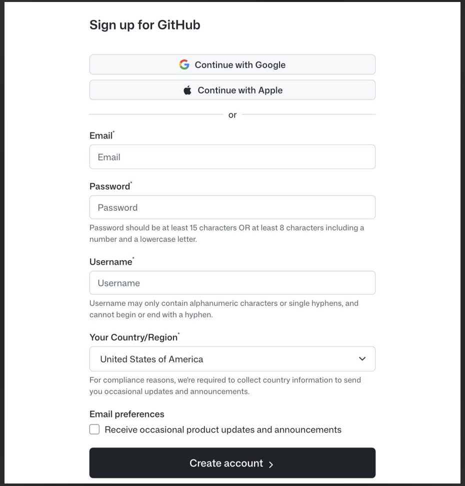
Creating a GitHub account

Visit: www.github.com

Click: “Sign Up” in the top right corner



Create your free account!



Sign up for GitHub

Email*

Email

Password*

Password

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*

Username

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your Country/Region*

United States of America

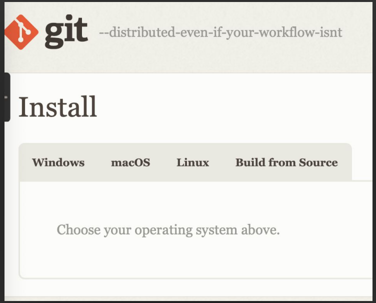
For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Email preferences

☐ Receive occasional product updates and announcements

Installing Git

Visit: https://git-scm.com/install/

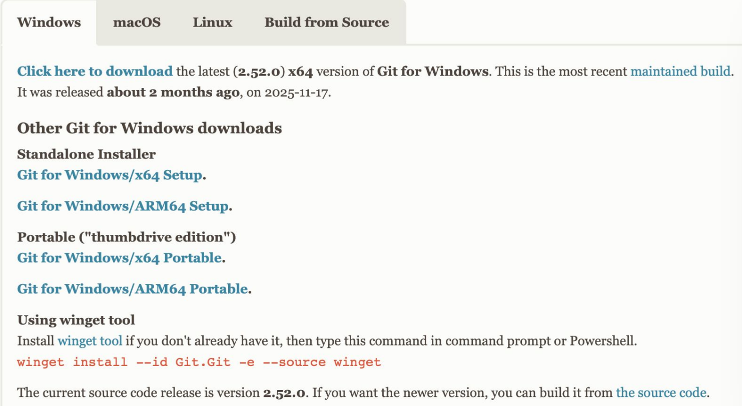


Install

Choose your operating system above.

Click on which OS you have

Follow the instructions on how to install git



Click here to download the latest (**2.52.0**) **x64** version of **Git for Windows**. This is the most recent maintained build. It was released **about 2 months ago**, on 2025-11-17.

Other Git for Windows downloads

Standalone Installer
Git for Windows/x64 Setup.

Git for Windows/ARM64 Setup.

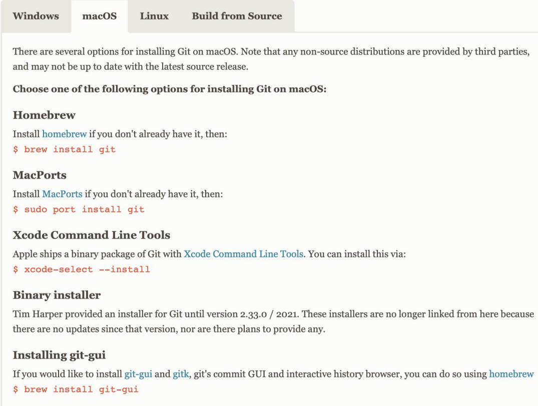
Portable ("thumbdrive edition")
Git for Windows/x64 Portable.

Git for Windows/ARM64 Portable.

Using winget tool
Install winget tool if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.52.0**. If you want the newer version, you can build it from the source code.



There are several options for installing Git on macOS. Note that any non-source distributions are provided by third parties, and may not be up to date with the latest source release.

Choose one of the following options for installing Git on macOS:

Homebrew
Install homebrew if you don't already have it, then:

```
$ brew install git
```

MacPorts
Install MacPorts if you don't already have it, then:

```
$ sudo port install git
```

Xcode Command Line Tools
Apple ships a binary package of Git with Xcode Command Line Tools. You can install this via:

```
$ xcode-select --install
```

Binary installer
Tim Harper provided an installer for Git until version 2.33.0 / 2021. These installers are no longer linked from here because there are no updates since that version, nor are there plans to provide any.

Installing git-gui
If you would like to install git-gui and gitk, git's commit GUI and interactive history browser, you can do so using homebrew

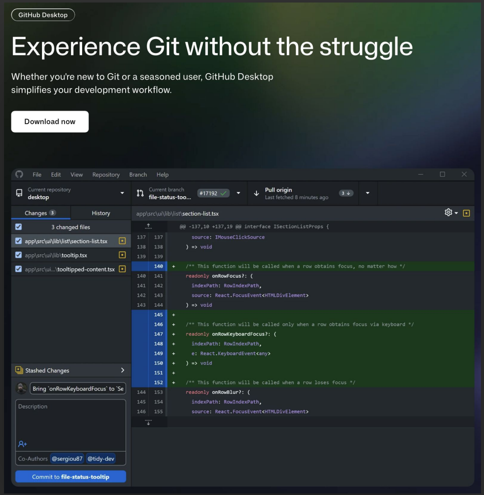
```
$ brew install git-gui
```

(Optional) Git clients

Different tools that make git actions easier to execute

GitHub Desktop

All in one git app, integrated with GitHub too



Experience Git without the struggle

Whether you're new to Git or a seasoned user, GitHub Desktop simplifies your development workflow.

☑ 3 changed files

☑ app\src\ui\lib\list\section-list.tsx

☑ app\src\ui\lib\tooltip.tsx

☑ app\src\ui...\tooltipped-content.tsx

Bring `onRowKeyboardFocus` to `Se

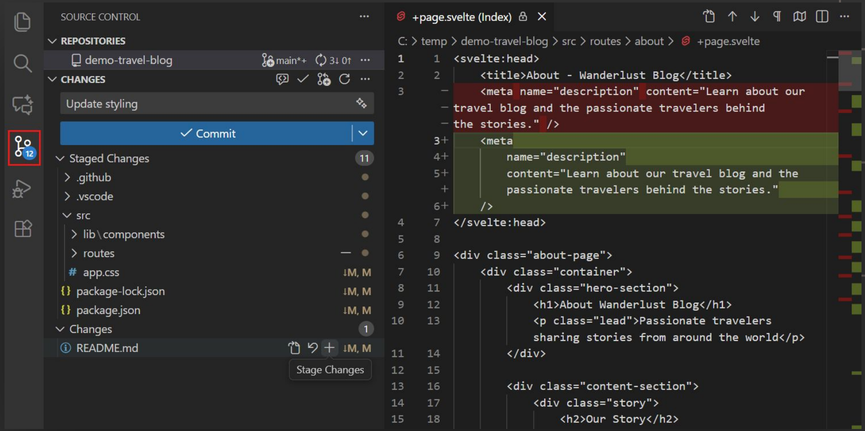
app\src\ui\lib\list\section-list.tsx

```
@@ -137,10 +137,19 @@ interface ISectionListProps {
      source: IMouseClickSource
    ) => void

+   /** This function will be called when a row obtains focus, no matter how */
    readonly onRowFocus?: (
      indexPath: RowIndexPath,
      source: React.FocusEvent<HTMLDivElement>
    ) => void
+
+   /** This function will be called only when a row obtains focus via keyboard */
+   readonly onRowKeyboardFocus?: (
+     indexPath: RowIndexPath,
+     e: React.KeyboardEvent<any>
+   ) => void
+
+   /** This function will be called when a row loses focus */
    readonly onRowBlur?: (
      indexPath: RowIndexPath,
      source: React.FocusEvent<HTMLDivElement>
```

IDE Integrated Plugins

Usually comes with IDE, or downloadable on plugin market



Update styling

```
<svelte:head>
    <title>About - Wanderlust Blog</title>
-   <meta name="description" content="Learn about our
-travel blog and the passionate travelers behind
-the stories." />
+   <meta
+       name="description"
+       content="Learn about our travel blog and the
+       passionate travelers behind the stories."
+   />
</svelte:head>

<div class="about-page">
    <div class="container">
        <div class="hero-section">
            <h1>About Wanderlust Blog</h1>
            <p class="lead">Passionate travelers
            sharing stories from around the world</p>
        </div>

        <div class="content-section">
            <div class="story">
                <h2>Our Story</h2>
```

Intro to GitHub

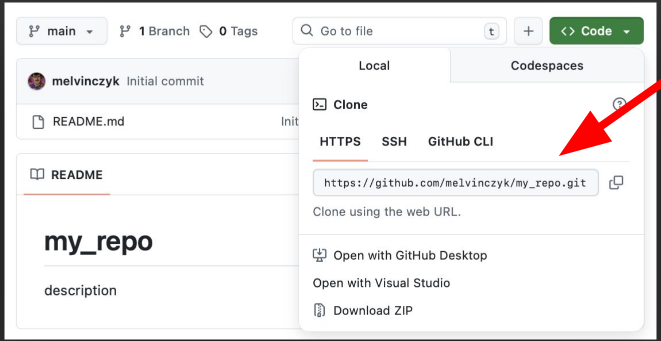
Creating or Publishing to GitHub

You can create and publish a repository to GitHub in 3 ways

On GitHub

1. Click on the “New” button →

2. Follow the instructions and “Create repository”
3. Clone the repo locally



melvinczyk Initial commit

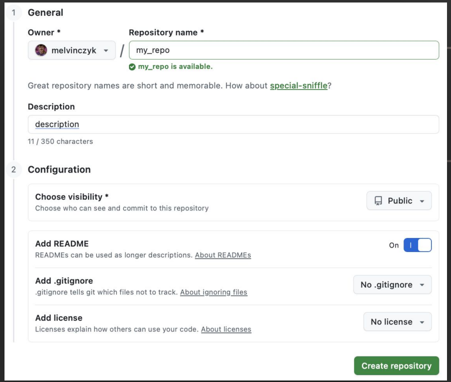
README.md Init

my_repo

description

`https://github.com/melvinczyk/my_repo.git`

Clone using the web URL.



1 General

Owner *: melvinczyk / Repository name *: my_repo

my_repo is available.

Great repository names are short and memorable. How about special-sniffle?

Description: description

11 / 350 characters

2 Configuration

Choose visibility * — Choose who can see and commit to this repository: Public

Add README — READMEs can be used as longer descriptions. About READMEs: On

Add .gitignore — .gitignore tells git which files not to track. About ignoring files: No .gitignore

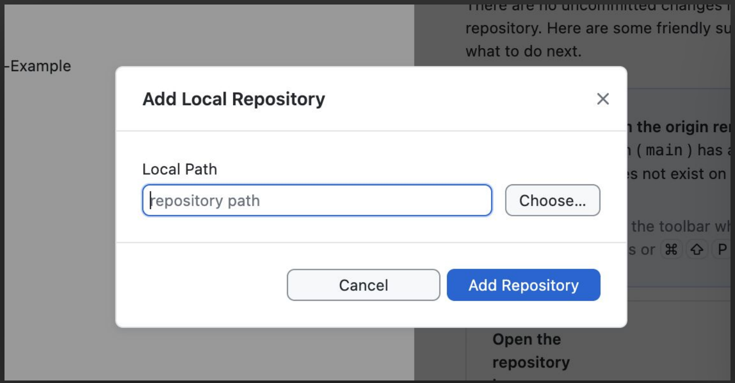
Add license — Licenses explain how others can use your code. About licenses: No license

Creating or Publishing to GitHub

You can create and publish a repository to GitHub in 3 ways

Publishing from GitHub Desktop

1. Create a repository locally `$ git init`
2. Add the files `$ git add .`
3. Use GitHub Desktop to publish to GitHub



There are no uncommitted changes i
repository. Here are some friendly su
what to do next.

-Example

Add Local Repository

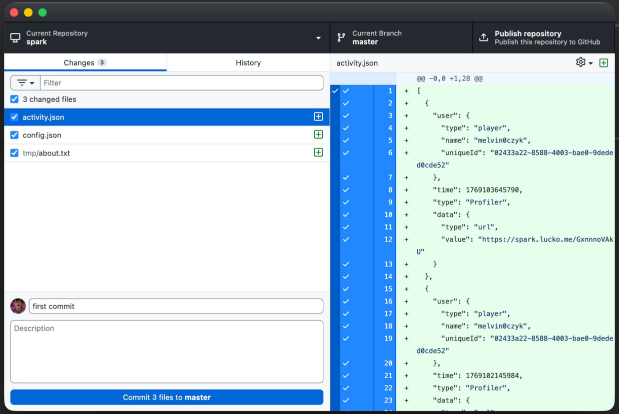
Local Path

repository path

n the origin re
n (main) has a
es not exist on

the toolbar wh
s or ⌘ ⇧ P

Open the
repository



Current Repository
spark

Current Branch
master

Publish repository
Publish this repository to GitHub

☑ 3 changed files

☑ activity.json

☑ config.json

☑ tmp/about.txt

first commit

activity.json

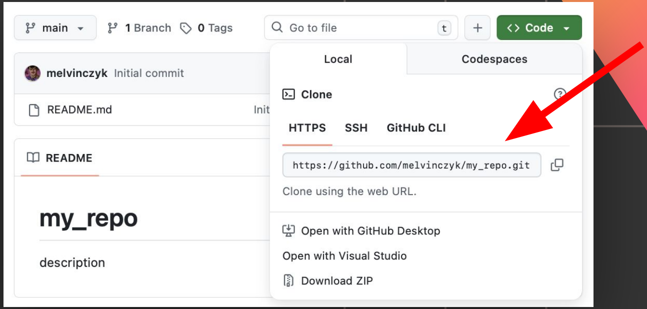
```
@@ -0,0 +1,28 @@
+ [
+   {
+     "user": {
+       "type": "player",
+       "name": "melvin0czyk",
+       "uniqueId": "02433a22-8588-4003-bae0-9deded0cde52"
+     },
+     "time": 1769103645790,
+     "type": "Profiler",
+     "data": {
+       "type": "url",
+       "value": "https://spark.lucko.me/GxnnnoVAkU"
+     }
+   },
+   {
+     "user": {
+       "type": "player",
+       "name": "melvin0czyk",
+       "uniqueId": "02433a22-8588-4003-bae0-9deded0cde52"
+     },
+     "time": 1769102145984,
+     "type": "Profiler",
+     "data": {
```

Creating or Publishing to GitHub

You can create and publish a repository to GitHub in 3 ways

Publishing from local

1. Initialize a repo on GitHub with no README
2. Create a repo locally `$ git init`
3. Add the files `$ git add .`
4. Change the origin to the new repo from GitHub `$ git remote add origin <link>`
5. Make a commit `$ git commit -m “message”`
6. Change branch `$ git branch -M main`
7. Push changes `$ git push -u origin main`



https://github.com/melvinczyk/my_repo.git

Clone using the web URL.

Basic commands

git clone - Copy (clone) a remote repository to your local computer

git status - See all the files that were changed

git add - Moves changes from your project directory to the staging area

git commit - Saves the staged changes to the local repository

git push - Sends your commits from local repository to the remote repository (on GitHub)

More at: *https://education.github.com/git-cheat-sheet-education.pdf* (Git Cheat Sheet)

Committing Changes

A commit is a snapshot of changes made to the project. You should make a commit every time you make a notable change or finish a task.

To track a new file created, execute `$ git add <file_name>`

Every commit needs a message with it, this is usually a short explanation of what changes were made.

To make a commit, execute `$ git commit -m “message”`

Pushing Changes

Once you have made a commit to your repo, you need to push it to GitHub

To push the commit, execute `$ git push -u origin main`

Assignment

1. Create a git repository for your group
2. One person in the group should start a group project with a document
3. Each member of the group should add their name to the document using commits.
4. Submit the link to the repository
5. Invite us to your repositories: Dr. Wagner - awagne30, Aditya - amakieni04, Nick - melvinczyk