

# Programming Memes

THIS IS GIT. IT TRACKS COLLABORATIVE WORK  
ON PROJECTS THROUGH A BEAUTIFUL  
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL  
COMMANDS AND TYPE THEM TO SYNC UP.  
IF YOU GET ERRORS, SAVE YOUR WORK  
ELSEWHERE, DELETE THE PROJECT,  
AND DOWNLOAD A FRESH COPY.

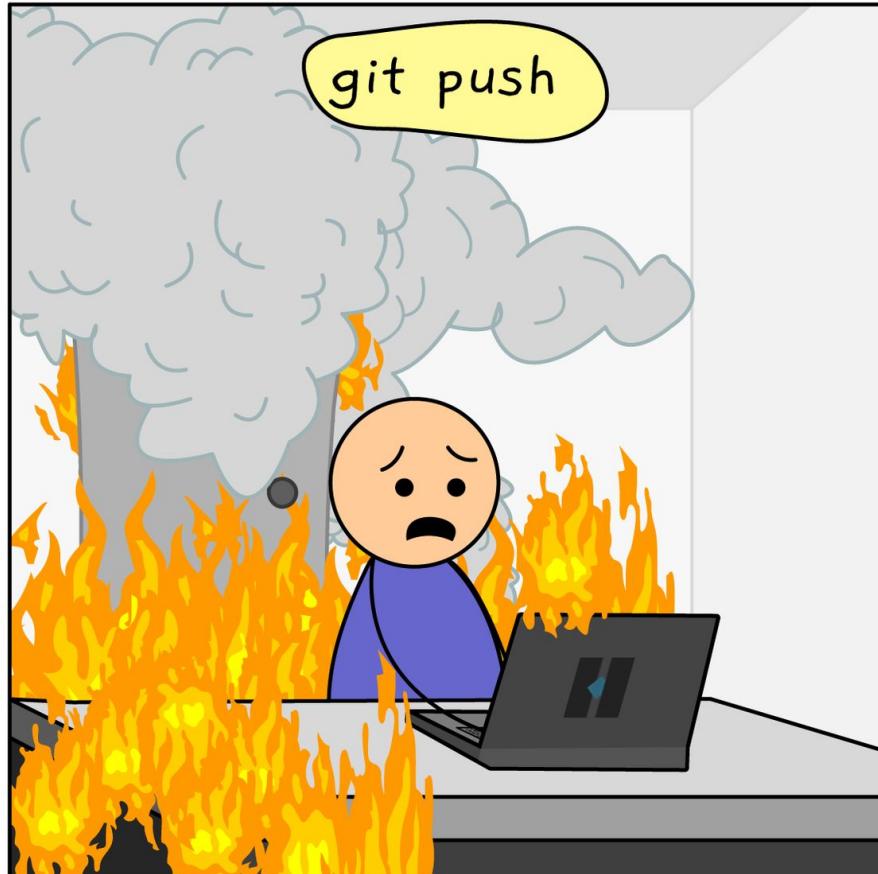


ME WHEN LOOKING AT THE CODE I WROTE TWO WEEKS AGO



I've never met this Code in my life.





# **Interview Questions**

# Hourglass

You have one 4-minute hourglass and one 7-minute hourglass.

How can you time 9 minutes exactly?



# Hourglass

## Solution:

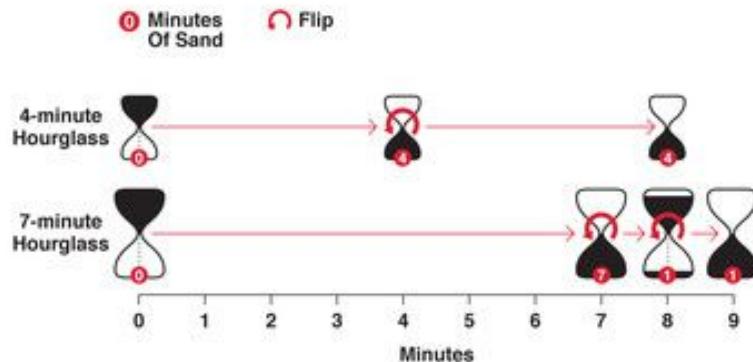
Start both timers at the same time.

When the 4 minute runs out, turn it over to start again.

When the 7 minute runs out, turn it over.

When the second 4 minute timer runs out, there will be 1 minute of sand in the 7 minute timer.

Turn it over and you will have your 9 minutes.



# **Resources**

# LinkedIn Learning

As a Humber student you have free access to all LinkedIn Learning courses:

<https://humber.ca/linkedinlearning/>

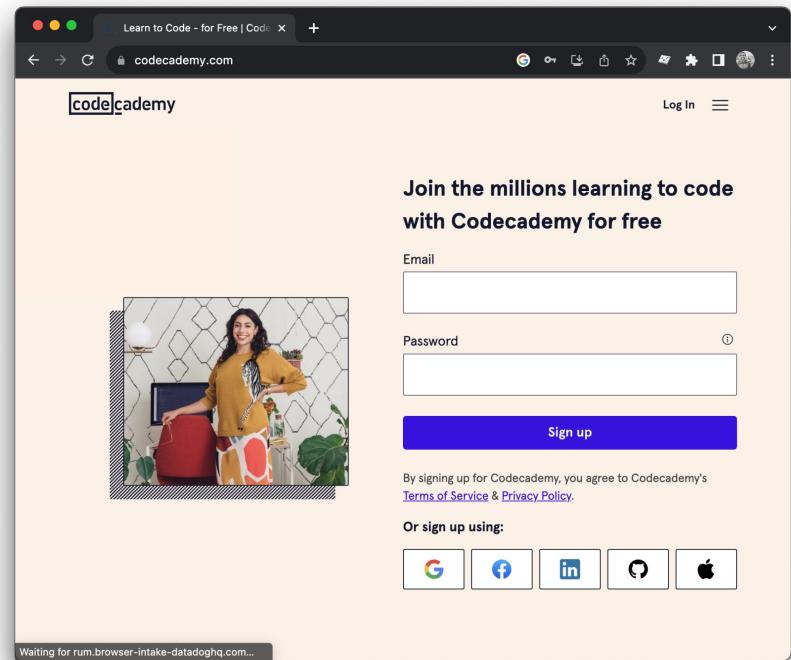


A screenshot of a web browser showing the LinkedIn Learning homepage. The URL in the address bar is 'humber.ca/linkedinlearning/'. The page features a dark background with a blue gradient overlay. At the top left is the LinkedIn logo and the word 'LEARNING'. Below it is the text 'WITH Lynda.com® CONTENT'. The main heading is 'Welcome to LinkedIn Learning'. A white box contains the text 'LINKEDIN LEARNING INFORMATION &amp; ACCESS INSTRUCTIONS'. It explains that users can sign in with their organization account (humber.ca) and notes that merging accounts is optional. At the bottom is a dark footer bar with icons for Program, Quick Links, Humber, Alerts, and Search, along with a red 'Info for' button.

# Codecademy

Codecademy is one of my favourite resources:

<https://www.codecademy.com/>



codecademy

# **Let's Fix Some Typos**

# Typos on Tasks

Let's fix some typos on the tasks website:

<https://tasks.brickmmo.com>

<https://www.w3.org/2002/01/spellchecker>

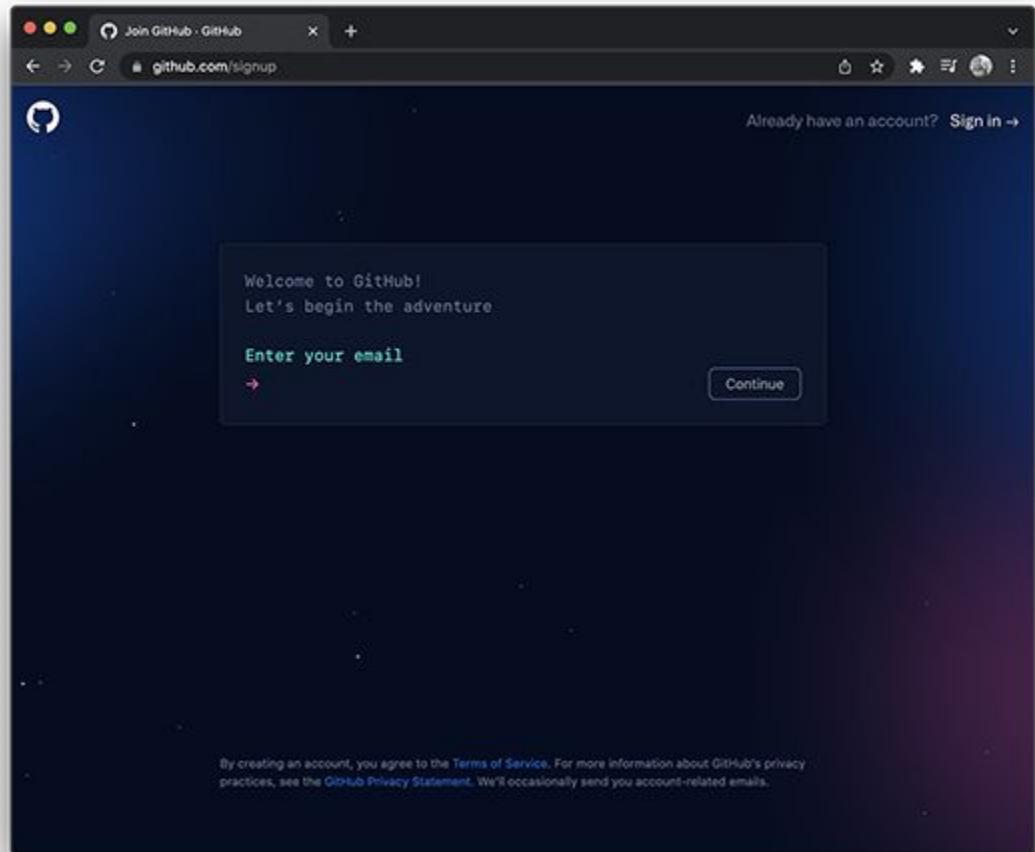
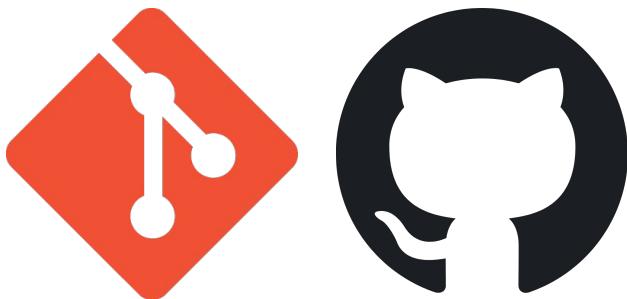
The screenshot shows the W3C Spell Checker website. It features a header with the W3C logo and the title "W3C Spell Checker". Below the header, there are sections for "Status" and "Usage". The "Status" section indicates that the tool supports English and French. The "Usage" section includes fields for the URL to check, language selection (set to English), and a checkbox for getting results. A "See also" section links to the QA Toolbox, and a "References" section lists the use of Aspell and Python. A "Known bugs and limitations" section notes that it doesn't handle language switching. At the bottom, there is a footer with authorship information.

The screenshot shows the tasks.brickmmo.com website. The page title is "tasks". It displays a message about the repository storing introductory tasks for students. A GitHub profile for Jason Alvarez (@0xBanana) is shown, featuring a list of commits. The commits are all identical: "I HATE PROGRAMMING". Below the commits, there is a note about due dates and a link to BrickMMO Flow. The "Phase One" section lists five tasks: Academic Integrity, Accounts, GitHub for Individuals, README.md Files, and Profile README.md. The first four tasks have been completed, while the fifth is pending.

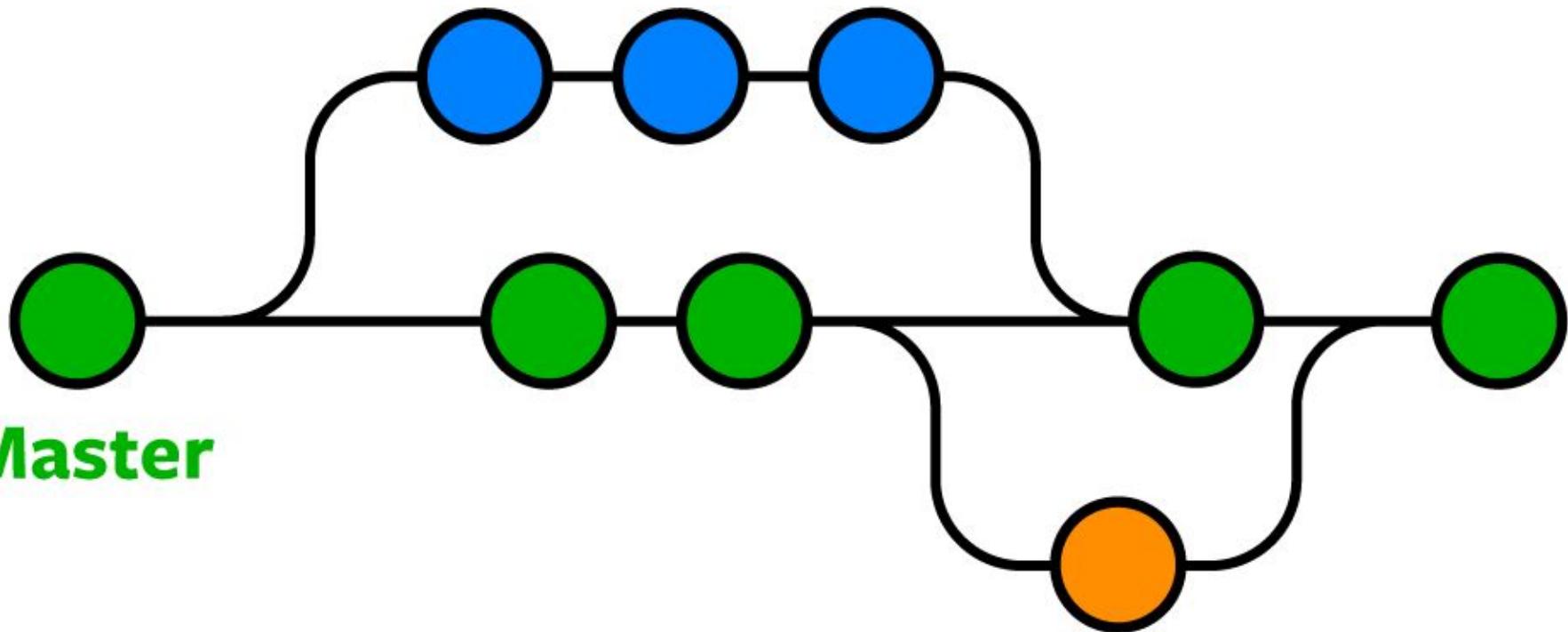
# **1. Git and GitHub**

# Git and GitHub

Version Control is a system for tracking changes to a file or files, which lets you access the various points in history of those changes.



## Your Work



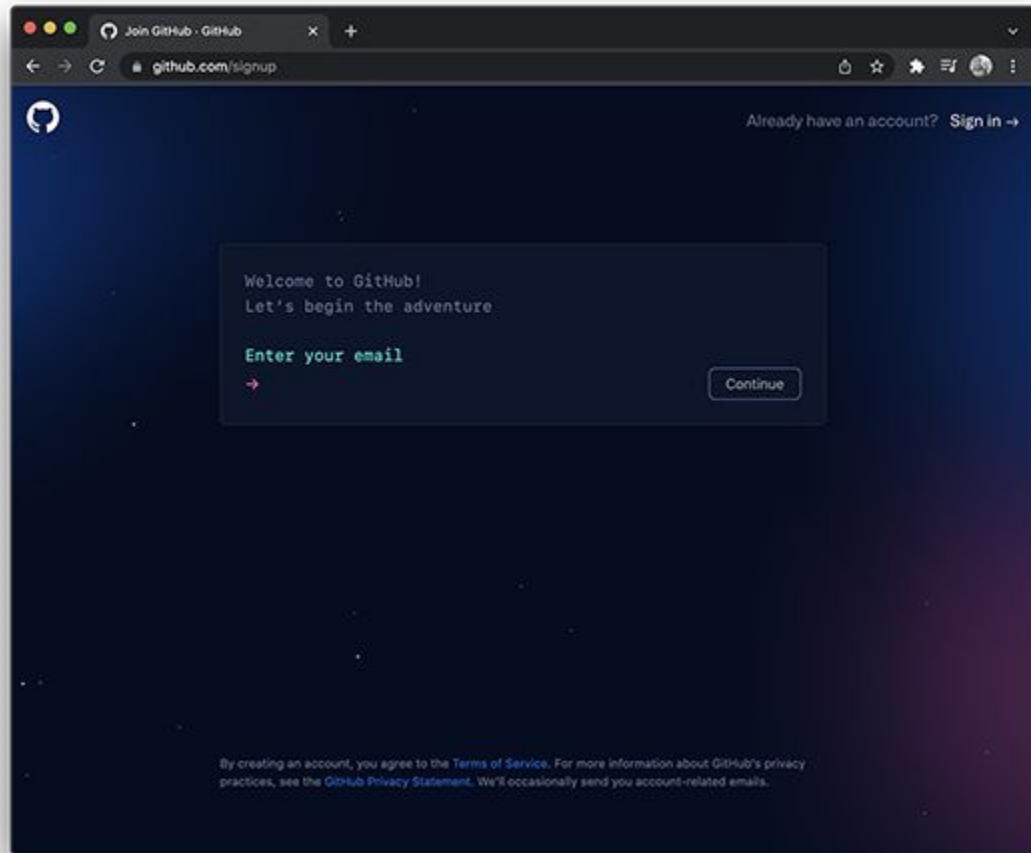
# Master

# Someone Else's Work

# GitHub Sign Up

Go to GitHub and register for an account.

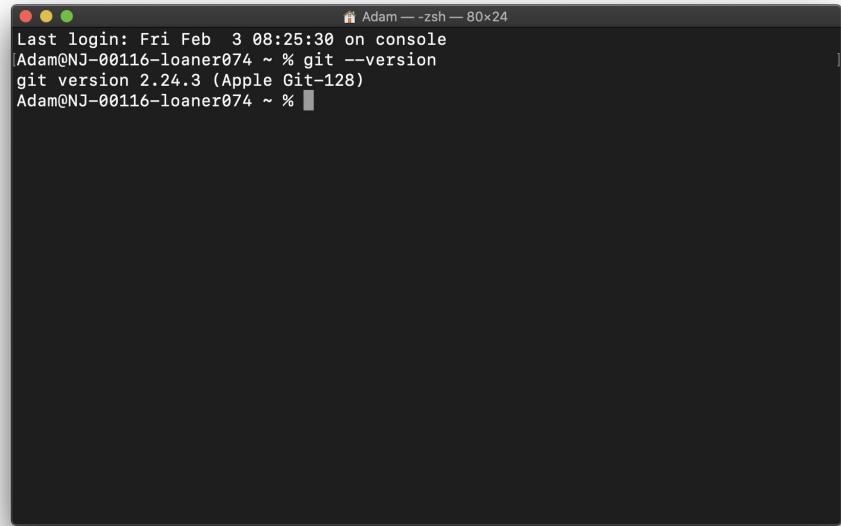
<https://github.com/signup>



# Git Installation

Run the following command to check if Git is installed:

```
$ git --version
```

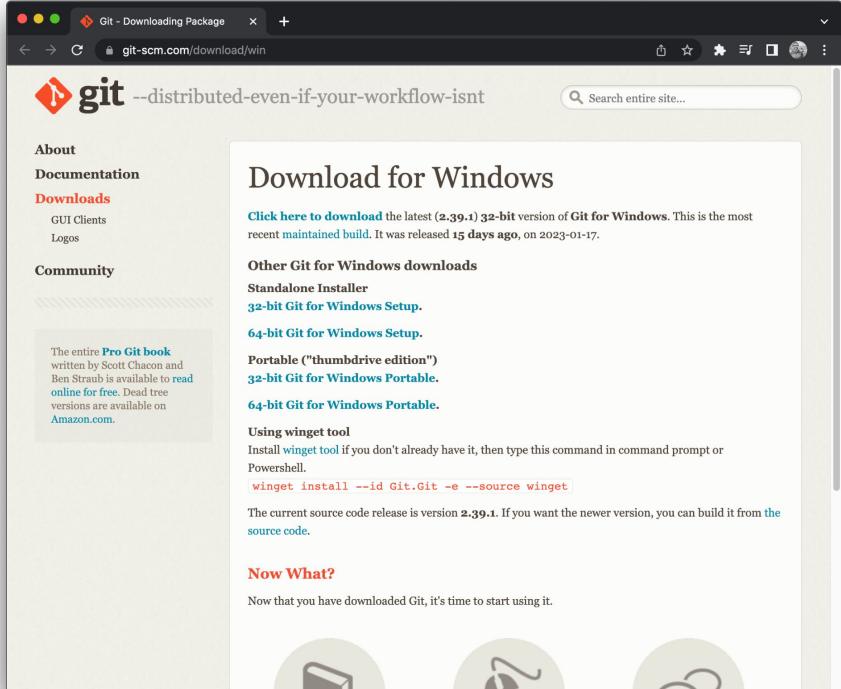
A screenshot of a macOS terminal window titled "Adam -- zsh -- 80x24". The window shows the output of the "git --version" command. The text is white on a black background. At the top, it says "Last login: Fri Feb 3 08:25:30 on console". Below that, it shows the command "git --version" and its output "git version 2.24.3 (Apple Git-128)". The prompt "Adam@NJ-00116-loaner074 ~ %" is at the bottom.

```
Last login: Fri Feb 3 08:25:30 on console  
Adam@NJ-00116-loaner074 ~ % git --version  
git version 2.24.3 (Apple Git-128)  
Adam@NJ-00116-loaner074 ~ %
```

# Git Installation: Windows

Navigate to the following URL and install Git:

<https://git-scm.com/download/win>



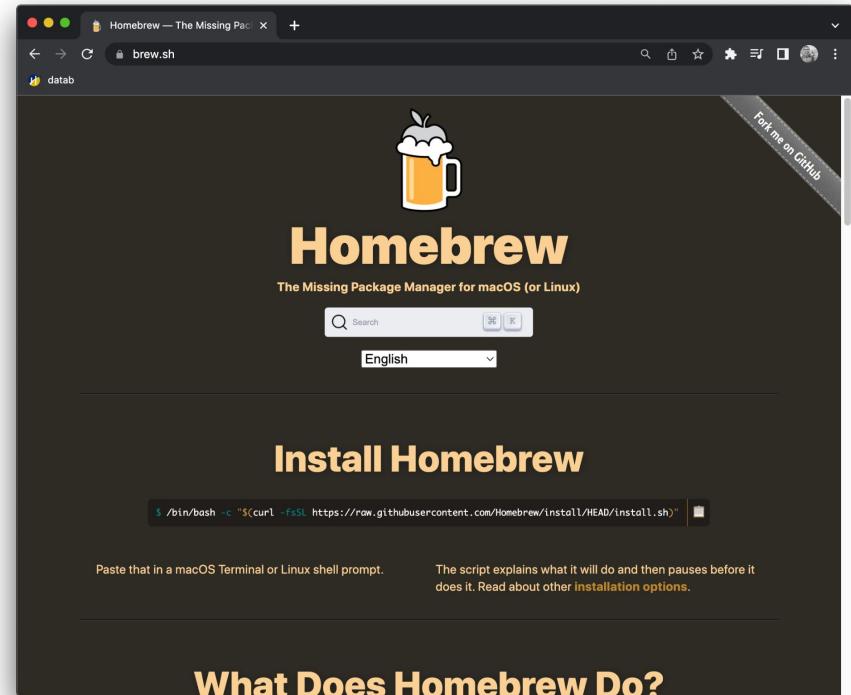
A screenshot of a web browser displaying the Git download page for Windows at [git-scm.com/download/win](https://git-scm.com/download/win). The page features the Git logo and the tagline "distributed-even-if-your-workflow-isn't". It includes sections for "About", "Documentation", "Downloads" (with links for "GUI Clients" and "Logos"), and "Community". A sidebar on the left contains links for the "Pro Git book" and "Using winget tool". The main content area is titled "Download for Windows" and provides a link to the latest 32-bit version of Git for Windows. It also lists other download options like "Standalone Installer", "32-bit Git for Windows Setup", "64-bit Git for Windows Setup", "Portable", "32-bit Git for Windows Portable", and "64-bit Git for Windows Portable". At the bottom, it shows the current source code release version 2.39.1 and instructions for building from source code. Three circular icons representing different Git clients are shown at the bottom right.

# Git Installation: Mac

Install Homebrew:

```
$ /bin/bash -c "$(curl  
-fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

<https://brew.sh/>



# Git Installation: Mac

Use Homebrew to install Git:

```
$ brew install git
```

A screenshot of a web browser window titled "git — Homebrew Formulae". The URL is "formulae.brew.sh/formula/git#default". The page content is as follows:

**git**

Install command: `brew install git`

Distributed revision control system

<https://git-scm.com>

License: [GPL-2.0-only](#)

Formula JSON API: </api/formula/git.json>

Bottle JSON API: </api/bottle/git.json>

Formula code: [git.rb](#) on GitHub

Bottle (binary package) installation support provided for:

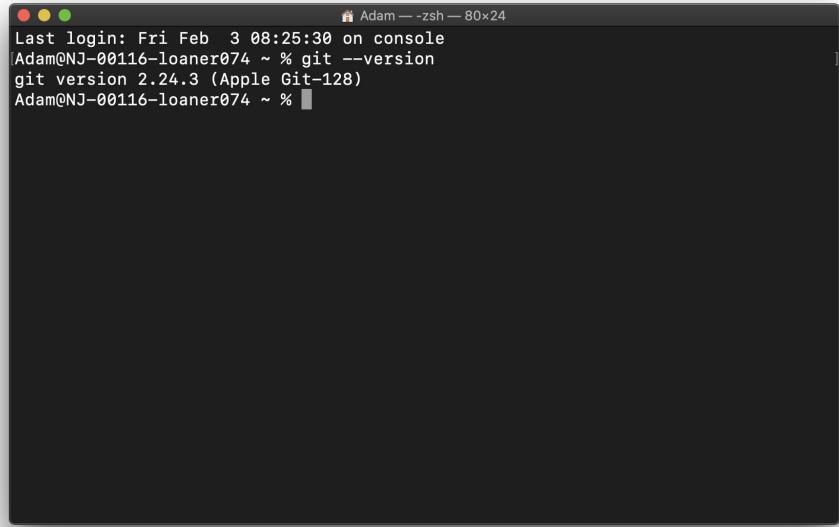
<b>Intel</b>	Ventura	✓
	Monterey	✓
	Big Sur	✓
	64-Bit Linux	✓
<b>Apple Silicon</b>	Ventura	✓
	Monterey	✓
	Big Sur	✓

Current versions:

# Git Installation

Confirm that Git is installed:

```
$ git --version
```

A screenshot of a macOS terminal window titled "Adam -- zsh -- 80x24". The window shows the command "git --version" being run and its output: "git version 2.24.3 (Apple Git-128)".

```
Last login: Fri Feb  3 08:25:30 on console  
Adam@NJ-00116-loaner074 ~ % git --version  
git version 2.24.3 (Apple Git-128)  
Adam@NJ-00116-loaner074 ~ %
```

# GitHub CLI

The GitHub CLI is a series of command line tool to make Git and GitHub easier.

<https://cli.github.com/>

```
$ gh --version
```

A screenshot of a macOS terminal window titled "Adam -- zsh -- 80x24". The window shows the following text:

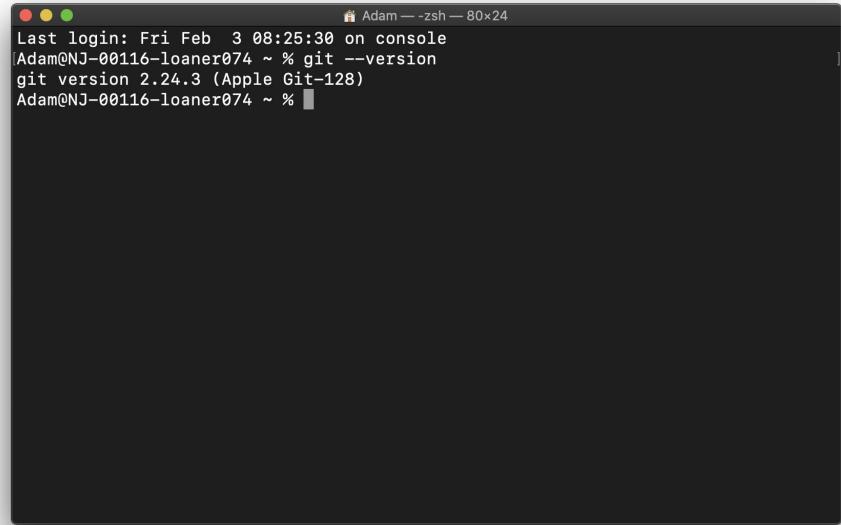
```
Last login: Fri Feb  3 08:25:30 on console  
Adam@NJ-00116-loaner074 ~ % git --version  
git version 2.24.3 (Apple Git-128)  
Adam@NJ-00116-loaner074 ~ %
```

## **2. Setup**

# GitHub CLI

Use the GitHub CLI to login:

```
$ gh auth login
```



A screenshot of a macOS terminal window titled "Adam -- zsh -- 80x24". The window shows the command "git --version" being run, with the output "git version 2.24.3 (Apple Git-128)". The terminal has a dark background and standard macOS window controls at the top.

# GitHub: Setup

Configure Git with your email and name:

```
$ git config --global user.email "name@email.com"  
$ git config --global user.name "First Last"
```



# GitHub: Setup

View your current Git configuration:

```
$ git config --list
```



```
[Adam@NJ-00116-loaner074 portfolio-draft % git config --list
credential.helper=osxkeychain
user.email=thomasadam83@hotmail.com
user.name=Adam Thomas
core.repositoryformatversion=0
core.filenode=true
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
core.precomposeunicode=true
remote.origin.url=https://github.com/codeadamca/portfolio-draft.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
Adam@NJ-00116-loaner074 portfolio-draft % ]
```

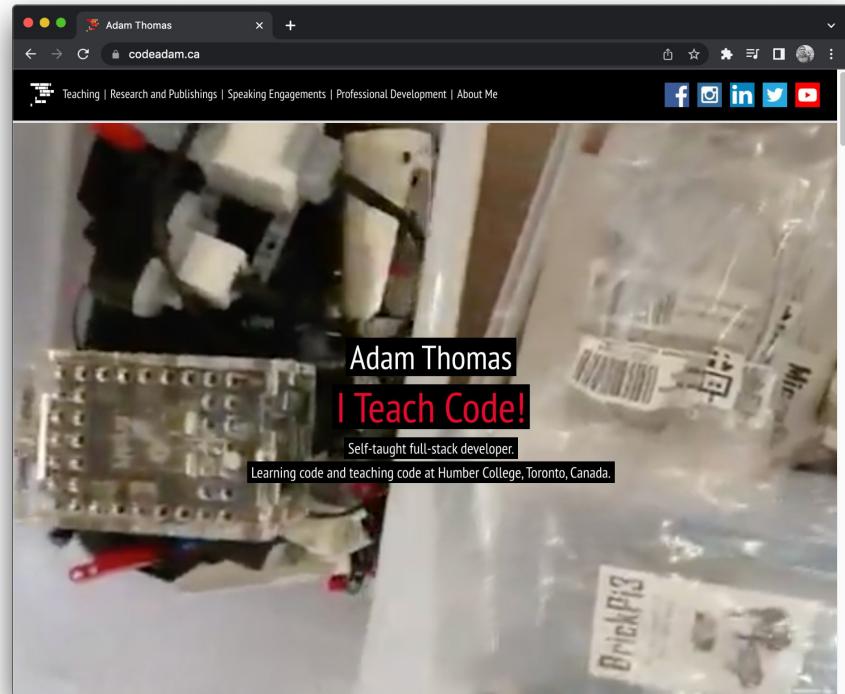
# Portfolio

Let's create a GitHub repository for this course.

We will call it:

http5114

Note: the recommended naming convention for GitHub repositories is kebab-case.



# GitHub: Setup

Copy your GitHub repository URL:

<https://github.com/codeadamca/http5114.git>

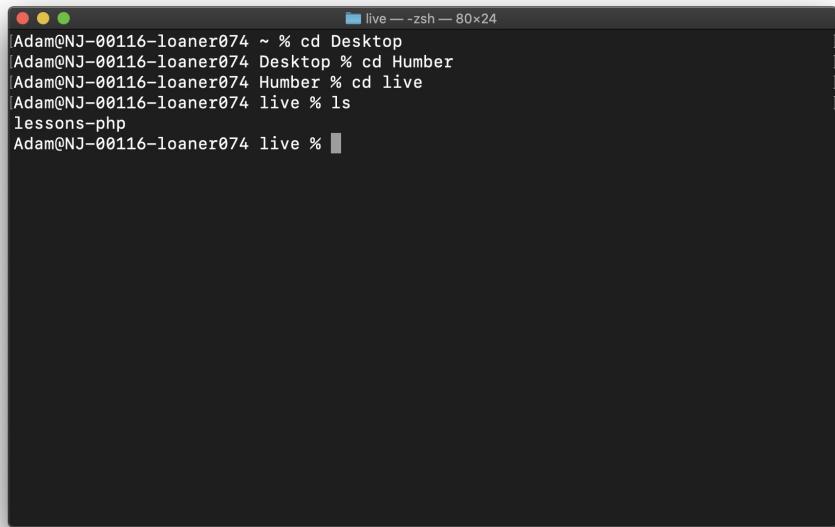
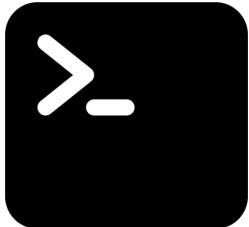


A screenshot of a GitHub repository page for "codeadamca/portfolio-draft". The URL in the browser bar is "github.com/codeadamca/portfolio-draft". The page shows a single file, "README.md", with the content "portfolio-draf". Below the file is a "Clone" section with links for "HTTPS", "SSH", and "GitHub CLI", and a "Use Git or checkout with SVN using the web URL" link. To the right, there's an "About" section with a description "A basic HTML draft of my portfolio.", and sections for "Releases", "Packages", and footer links including "Terms", "Privacy", "Security", "Status", "Docs", "Contact GitHub", "Pricing", "API", "Training", "Blog", and "About".

# Command Line Basics

Open a terminal and navigate to your working folder:

```
$ cd Desktop/Humber/live
```



A screenshot of a terminal window titled "live — -zsh — 80x24". The window shows the following command history:

```
[Adam@NJ-00116-loaner074 ~ % cd Desktop
[Adam@NJ-00116-loaner074 Desktop % cd Humber
[Adam@NJ-00116-loaner074 Humber % cd live
[Adam@NJ-00116-loaner074 live % ls
lessons-php
[Adam@NJ-00116-loaner074 live % ]
```

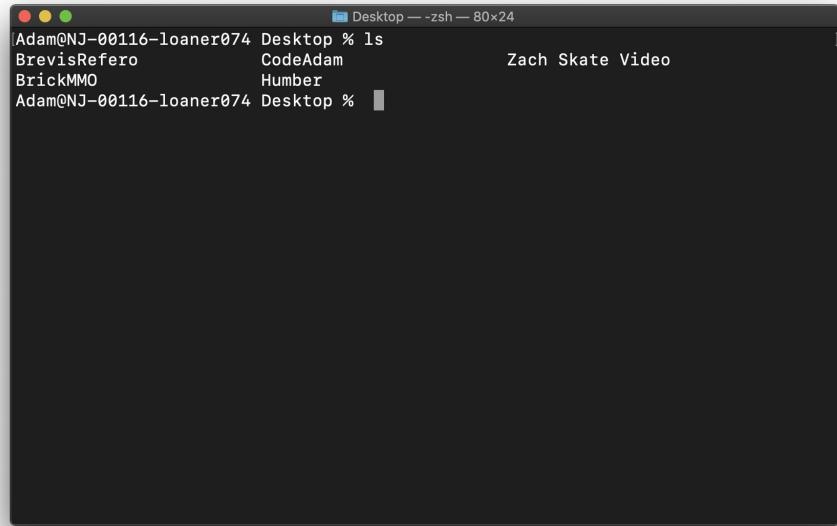
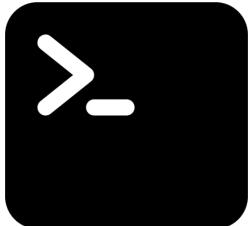
# Command Line Basics

At anytime you can view the contents of the folder you're in. On a Mac:

```
$ ls
```

Or on a PC:

```
$ dir
```

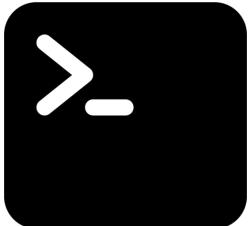
A screenshot of a Mac OS X terminal window titled "Desktop — zsh — 80x24". The window shows the command "ls" being run in the directory "/Users/Adam/Desktop". The output lists several files and folders: BrevisRefero, BrickMMO, CodeAdam, Humber, Zach, Skate, and Video. The prompt at the bottom right indicates the user is still in the terminal session.

```
[Adam@NJ-00116-loaner074 Desktop % ls
BrevisRefero           CodeAdam
BrickMMO                Humber
Zach      Skate  Video
Adam@NJ-00116-loaner074 Desktop % ]
```

# Command Line Basics

At anytime you can view the full location of the current folder:

```
$ pwd
```

A screenshot of a terminal window titled "Desktop — -zsh — 80x24". The window shows the command "pwd" being run, followed by the output "/Users/Adam/Desktop".

```
[Adam@NJ-00116-loaner074 Desktop % pwd  
/Users/Adam/Desktop  
Adam@NJ-00116-loaner074 Desktop % ]
```

# GitHub: Setup

Clone your repo:

```
$ git clone https://github.com/codeadamca/http5114.git
```

Change the directory to your new repo:

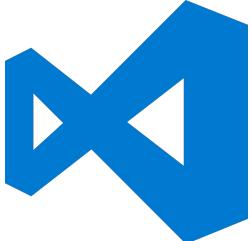
```
$ cd http5114
```



# Visual Studio

Point Visual Studio Code to the new folder.

Create an index.html page. Add some content. Save.



The screenshot shows the Visual Studio Code interface. In the Explorer sidebar, there is a folder named 'portfolio-draft' containing 'index.html', 'lessons-php', and 'README.md'. The 'index.html' file is selected in the Explorer. In the main editor area, the code for 'index.html' is displayed:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Adam Thomas</title>
</head>
<body>
<h1>My Portfolio</h1>
</body>
</html>
```

At the bottom of the interface, the status bar shows: Ln 11, Col 26 Spaces: 4 UTF-8 LF PHP LiveCode Prettier

# **3. Committing Changes**

# GitHub: Commit

Check the repo current status:

```
$ git status
```



```
[Adam@NJ-00116-loaner074 portfolio-draft % git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
Adam@NJ-00116-loaner074 portfolio-draft % ]
```

# GitHub: Commit

Add all current changes to the next commit:

```
$ git add .
```



```
[Adam@NJ-00116-loaner074 portfolio-draft % git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
[Adam@NJ-00116-loaner074 portfolio-draft % git add .
[Adam@NJ-00116-loaner074 portfolio-draft % ]
```

# GitHub: Commit

Commit the changes:

```
$ git commit -am "Added  
initial home page"
```



```
[Adam@NJ-00116-loaner074 portfolio-draft % git status  
On branch main  
Your branch is up to date with 'origin/main'.  
  
Untracked files:  
  (use "git add <file>..." to include in what will be committed)  
    index.html  
  
nothing added to commit but untracked files present (use "git add" to track)  
[Adam@NJ-00116-loaner074 portfolio-draft % git add .  
[Adam@NJ-00116-loaner074 portfolio-draft % git commit -am "Added initial home pag  
e"  
[main 38a40bb] Added initial home page  
  1 file changed, 14 insertions(+)  
  create mode 100644 index.html  
Adam@NJ-00116-loaner074 portfolio-draft %
```

# GitHub: Commit

Upload local repo content to your GitHub repo:

```
$ git push origin main
```



```
portfolio-draft -- zsh -- 80x24
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
[Adam@NJ-00116-loaner074 portfolio-draft % git add .
[Adam@NJ-00116-loaner074 portfolio-draft % git commit -am "Added initial home pag
e"
[main 38a40bb] Added initial home page
  1 file changed, 14 insertions(+)
  create mode 100644 index.html
[Adam@NJ-00116-loaner074 portfolio-draft % git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 489 bytes | 489.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/codeadamca/portfolio-draft.git
  736f1e8..38a40bb  main -> main
[Adam@NJ-00116-loaner074 portfolio-draft % ]
```

# GitHub: Online Repo

Refresh your GitHub repo to view your online files.

A screenshot of a GitHub repository page. The URL in the address bar is "github.com/codeadamca/portfolio-draft". The repository name is "codeadamca / portfolio-draft" (Public). The "Code" tab is selected. In the commit list, there are two commits: "Added initial home page" (38a40bb, 8 minutes ago) and "Initial commit" (3 days ago). The "index.html" file is highlighted with a red circle. The repository description is "A basic HTML draft of my portfolio." The sidebar shows "About" (0 stars, 1 watching, 0 forks), "Releases" (No releases published, Create a new release), "Packages" (No packages published, Publish your first package), and "Languages" (HTML 100.0%).

codeadamca / portfolio-draft (Public)

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

Code Adamca / portfolio-draft (Public) Pin Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code About

codeadamca Added initial home page 38a40bb 8 minutes ago 2 commits

Initial commit 3 days ago

index.html Added initial home page 8 minutes ago

README.md

**portfolio-draft**

A basic HTML draft of my portfolio.

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Languages

HTML 100.0%

# **4. GitHub for Individuals**

Create a repo each course.

codeadamca (Adam Thomas) +

github.com/codeadamca?tab=repositories

codeadamca

Overview Repositories 188 Projects Packages Stars 1

Find a repository... Type Language Sort New

**react-console** Public

Testing an admin console using React and Material Design.

react

Updated 3 weeks ago

**javascript-dom** Public

A basic example of using JavaScript to make changes to a webpage using the DOM.

javascript dom learning-code

HTML Updated 3 weeks ago

**javascript-mouse-events** Public

A basic example of using JavaScript to connect mouse events to DOM changes.

javascript mouse-events learning-code

HTML Updated 3 weeks ago

Adam Thomas  
codeadamca

I Teach Code! Self-taught full-stack developer. Learning code and teaching code at Humber College, Toronto, Canada.

Edit profile

100 followers · 5 following

<https://codeadam.ca>

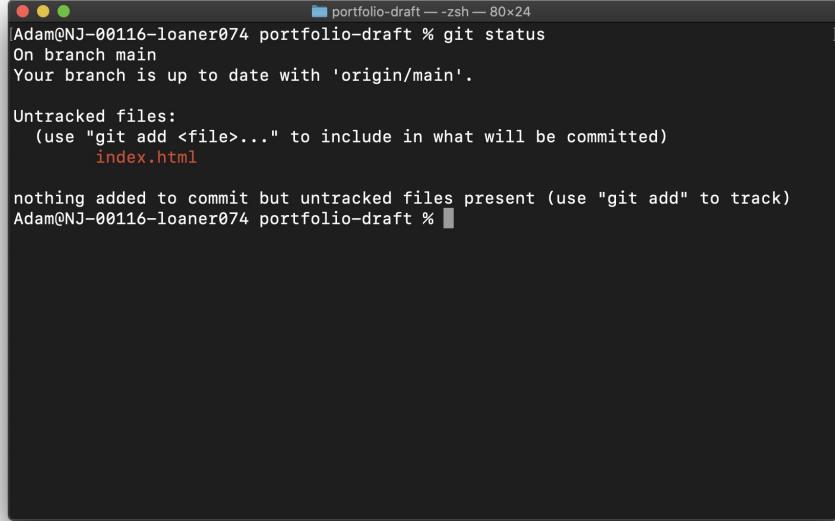
@codeadamca

## **5. README.md**

# README.md Files

Each repo should have a README.md file:

- Repo description
- Instructions
- Helpful links



```
[Adam@NJ-00116-loaner074 portfolio-draft % git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
Adam@NJ-00116-loaner074 portfolio-draft % ]
```

A screenshot of a terminal window titled 'portfolio-draft -- zsh -- 80x24'. The window shows the output of a 'git status' command. It indicates that the user is on branch 'main' and their branch is up to date with 'origin/main'. There are untracked files, specifically 'index.html', which are highlighted in red. A message at the bottom says 'nothing added to commit but untracked files present (use "git add" to track)'. The terminal window has a dark background and light-colored text.

# README.md Files

Make a README.me for your new repo:

<https://www.markdownguide.org/>



The screenshot shows a GitHub repository page for the '\_readme' repository. The main content is the README.md file, which contains the following text:

```
_readme

A set of guidelines for GitHub repos, README.md files, and README.md assets to a GitHub repo.

This repo is available to view at
https://readme.codeadam.ca
```

Below the README, there are sections for 'Project Stack' (using only Markdown) and 'Repo Resources' (links to \_readme, GitHub - About README.md files, and Markdown). On the right side of the page, there are sections for 'Releases', 'Packages', and 'Deployments'.

# \_readme

\_readme is a set of standards  
BrickMMO follows when formatting  
README.md files and assets:

<https://readme.codeadam.ca/>



The screenshot shows a GitHub repository page for the '\_readme' project. The main content is the README.md file, which contains the title '\_readme' and a brief description: 'A set of guidelines for GitHub repos, README.md files, and README.md assets to a GitHub repo.' It also includes a link to the website: <https://readme.codeadam.ca/>. To the right of the README, there are sections for 'Releases', 'Packages', and 'Deployments'. The 'Releases' section shows 'No releases published' and a 'Create a new release' button. The 'Packages' section shows 'No packages published' and a 'Publish your first package' button. The 'Deployments' section shows one deployment named 'github-pages' from 2 weeks ago and '+ 46 deployments'.

# Markdown Resume

Create a new public repo named "resume". Include a README.md file.

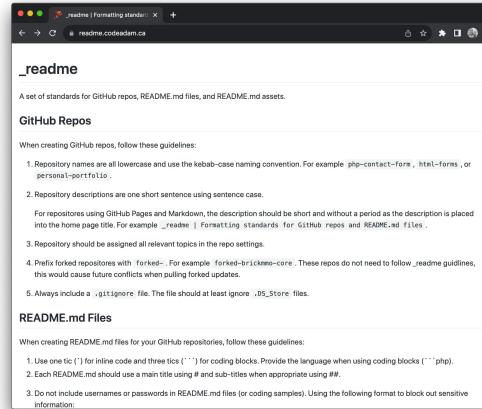
Add the following to your README.md:

- A level one heading (your name).
- Level two headings (about, projects, education, work experience, contact)
- A paragraph (a short bio under your name).
- A profile image
- A table with a list of your projects (title, URL, technology).
- A list for education and work experience.
- A list of social media accounts with links.

All Markdown must follow \_readme guidelines:

<https://readme.codeadam.ca/>

Note: You do NOT need to include the readme.css file.



# **6. README.md**

Create a README.md file each course.

codeadamca/react-console

github.com/codeadamca/react-console

main 1 branch 0 tags Go to file Add file Code About

codeadamca Update README.md 2e533c1 3 weeks ago 2 commits

README.md Update README.md 3 weeks ago

README.md

# React Console

Creating an admin console using React.

## Repo Resources

- Visual Studio Code



Testing an admin console using React and Material Design.

react

Readme Activity 0 stars 1 watching 0 forks

## Releases

No releases published Create a new release

## Packages

No packages published Publish your first package

© 2023 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About



**END**

# **Unused Slides**

### **3. Auth Using a Token**

# GitHub: Setup

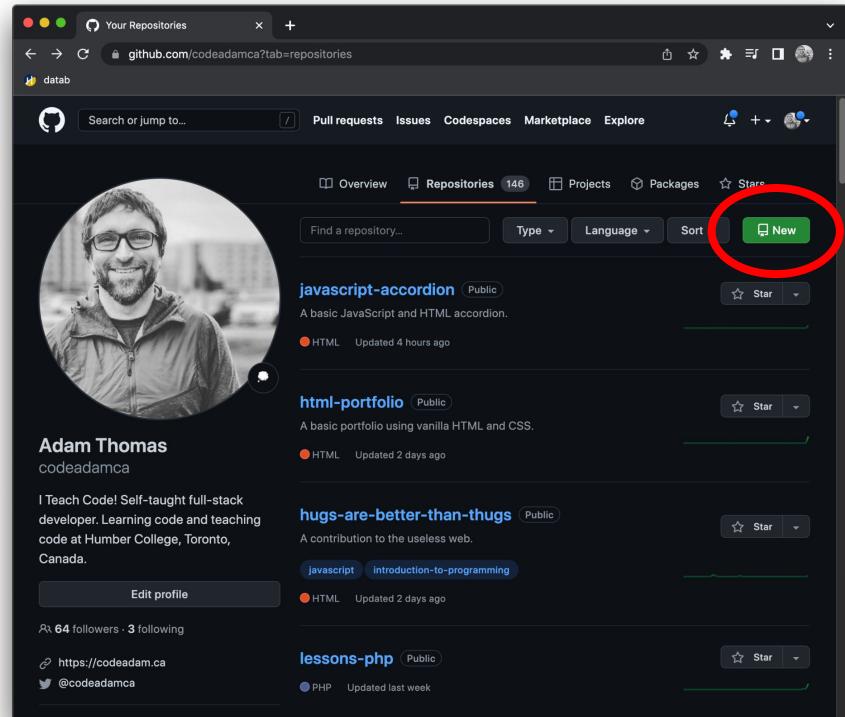
Log in to your GitHub account and create a new repo:

- 1) Go to your profile

<https://github.com/codeadamca>

- 2) Click "Repositories"

- 3) Click the  button

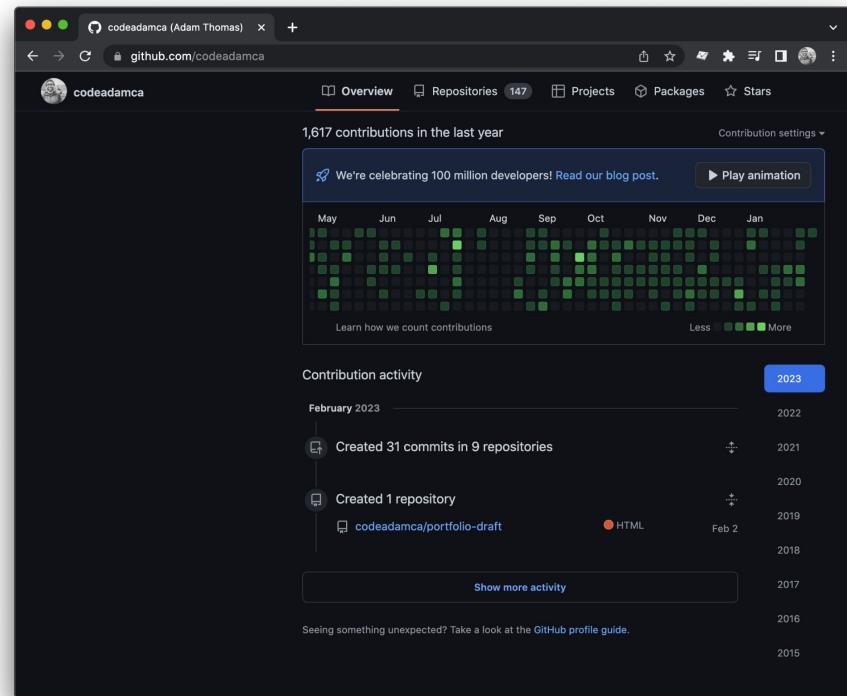


A screenshot of a GitHub profile page for a user named Adam Thomas. The profile picture shows a smiling man with glasses and a beard. The page displays four repositories: 'javascript-accordion' (public), 'html-portfolio' (public), 'hugs-are-better-than-thugs' (public), and 'lessons-php' (public). The 'repositories' tab is selected, showing 146 repos. A green 'New' button with a plus sign is highlighted with a red circle. The GitHub interface includes a search bar, pull requests, issues, codespaces, marketplace, and explore tabs at the top.

# GitHub: Contributions

Get your contributions up!

- Create a repo for each class.
- Create a repo for each project.
- Create a repo for any concept you are learning.
- Create a repo for each version of your portfolio.



# GitHub: Public Profile

You have a special repo:  
codeadamca/codeadamca



The screenshot shows a GitHub repository page for the user 'codeadamca'. The repository name is 'codeadamca/codeadamca'. The page includes a search bar, navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore, and a header showing 1 unwatched repository, 0 forks, and 0 stars. The main content area displays a file named 'README.md' with a commit history from Oct 4, 2022, showing updates to the README. A sidebar on the right highlights that this is a special repository and provides links to edit the README and visit the profile. The profile section features a bio for Adam Thomas, a link to his website, and social media icons for LinkedIn, Facebook, YouTube, Twitter, and Instagram. The repository stats show 0 stars, 1 watching, and 0 forks. The releases section indicates no releases have been published.

# MarkDown

Use MarkDown to create your profile page.

<https://www.markdownguide.org/>



The screenshot shows the homepage of the Markdown Guide website at markdownguide.org. The header features the site's logo and navigation links for 'Markdown Guide', 'Get Started', 'Cheat Sheet', 'Basic Syntax', 'Extended Syntax', 'Hacks', 'Tools', and 'Book'. A search bar is also present. The main content area has a blue background with a large 'M↓' logo. Below it, a text block describes the site as a free, open-source reference guide for Markdown. A 'Get Started' button is located in the bottom right corner of the blue section. At the very bottom, there is a small yellow banner with the text 'Never miss a service outage again: Real-time'.

# MarkDown

Check out some other great profile pages!

<https://github.com/abhisheknaiidu/awesome-github-profile-readme>



The screenshot shows a GitHub repository page for 'abhisheknaiidu / awesome-github-profile-readme'. The page has a dark theme. At the top, there's a search bar and navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore. Below the header, the repository name 'abhisheknaiidu / awesome-github-profile-readme' is shown with a 'Public' status. The 'Code' tab is selected, showing the master branch with 15 branches and 0 tags. A list of files includes '.github', 'assets', 'LICENSE', 'README.md', and 'contributing.md', all updated 2 years ago. The 'About' section describes it as a curated list of awesome GitHub Profile READMEs, with a link to 'bit.ly/awesome-gh'. It lists tags like 'github', 'portfolio', 'awesome', 'awesome-list', 'github-readme', 'profile-readme', and 'github-profile-readme'. The 'README.md' file is expanded, showing its content: 'Awesome GitHub Profile README' and a 'Product Hunt' badge with a rank of 74. The 'Releases' section indicates no releases have been published. The bottom of the page shows social sharing icons for GitHub, awesome, arbeitnow, If Useful, and Join Community, along with an online status of 33 users.

# GitHub: Setup

- 1) Name the repo:  
portfolio-draft
- 2) Provide a brief description:  
A basic HTML draft of my portfolio.
- 3) Check the **Public** and the **Add a README file** option.



The screenshot shows the GitHub 'Create a New Repository' interface. The 'Repository name' field is filled with 'portfolio-draft'. The 'Description (optional)' field contains 'A basic HTML draft of my portfolio.' The 'Public' radio button is selected. The 'Add a README file' checkbox is checked. The 'Create repository' button is at the bottom.

# GitHub: Setup

Create a Personal Access Token.

- 1) Click on your avatar and then **Settings**.
- 2) At the bottom click **Developer Settings**.
- 3) Click **Personal access tokens**.
- 4) Click **Tokens (classic)**.
- 5) Click **Generate new token (classic)**.



The screenshot shows the GitHub developer settings page with the "Personal access tokens (classic)" section selected. A red circle highlights the "Generate new token (classic)" button, which is located above a list of existing tokens. The tokens listed include "BrickPI", "CodeAdamCa", and "BrevisRefero", each with their respective scopes and usage details.

# GitHub: Setup

- 1) Name the token.
- 2) Choose **No expiration**.
- 3) Select the **repo** scope.
- 4) Click **Generate token**.
- 5) Copy your token and put it somewhere!!!



The screenshot shows the GitHub settings interface for creating a new personal access token. The left sidebar has sections for GitHub Apps, OAuth Apps, Personal access tokens (selected), Fine-grained tokens (Beta), and Tokens (classic). The main area is titled "New personal access token (classic)". It includes fields for "Note" (set to "Humber Class"), "Expiration" (set to "No expiration" with the note "The token will never expire!"), and a warning about not setting an expiration date. Below this is a "Select scopes" section with a note about scopes defining access. The "repo" scope is selected, granting full control of private repositories, commit status, deployment status, public repositories, repository invitations, and security events. Other scopes listed include workflow, write:packages, read:packages, and delete:packages.

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note: Humber Class

What's this token for?

Expiration: No expiration The token will never expire!

GitHub strongly recommends that you set an expiration date for your token to help keep your information secure. [Learn more](#)

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

Scope	Description
<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry

# GitHub: Setup

Use your Personal Access Token to log in:

```
$ git remote set-url origin  
https://ghp_classic_token@github.com/username/repo-name.  
git
```

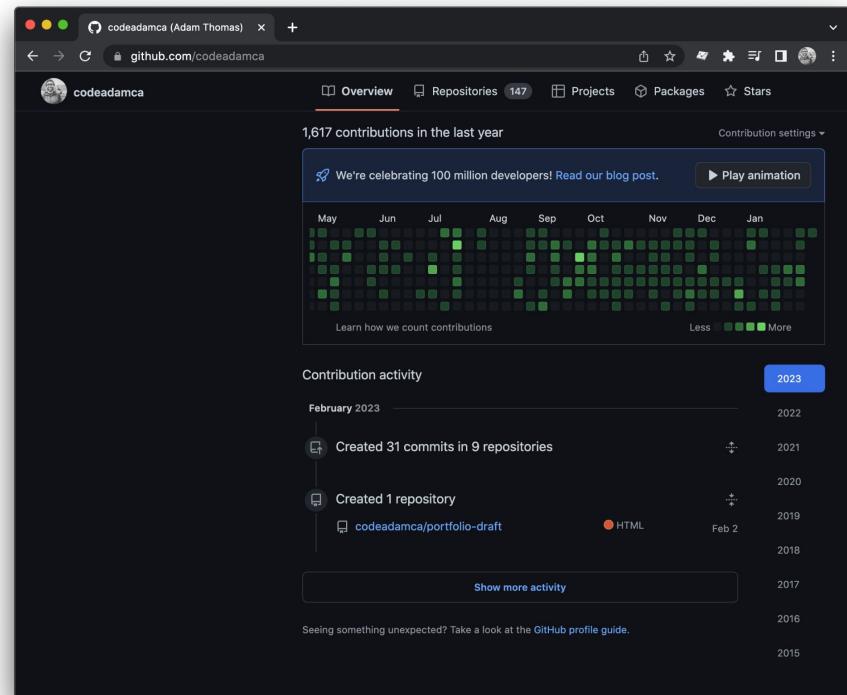
You only need to do this for your first repository! Your next repository will use the same credentials.

### **3. GitHub Profile**

# GitHub: Contributions

Get your contributions up!

- Create a repo for each class.
- Create a repo for each project.
- Create a repo for any concept you are learning.
- Create a repo for each version of your portfolio.



# GitHub: Public Profile

You have a special repo:  
codeadamca/codeadamca



The screenshot shows a GitHub repository page for the user 'codeadamca'. The repository name is 'codeadamca/codeadamca'. The page includes a search bar, navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore, and a header showing 1 unwatched repository, 0 forks, and 0 stars. The main content area displays a file named 'README.md' with a commit history from Oct 4, 2022, showing 36 commits. A note on the right states: 'codeadamca/codeadamca is a special repository. Its README.md will appear on your public profile.' Below the file list, there's a section for 'About' with a bio: 'I Teach Code! Self-taught full-stack developer. Learning code and teaching code at Humber College, Toronto, Canada.' It also lists social media links for LinkedIn, Facebook, YouTube, Twitter, and Instagram. The sidebar on the right shows repository statistics: 0 stars, 1 watching, and 0 forks. There is also a 'Releases' section indicating 'No releases published'.

# MarkDown

Use MarkDown to create your profile page.

<https://www.markdownguide.org/>



The screenshot shows the homepage of the Markdown Guide website at markdownguide.org. The header features the site's logo and navigation links for 'Markdown Guide', 'Get Started', 'Cheat Sheet', 'Basic Syntax', 'Extended Syntax', 'Hacks', 'Tools', and 'Book'. A search bar is also present. The main content area has a blue background with a large 'M↓' logo. Below it, a text block describes the site as a free, open-source reference guide for Markdown. A 'Get Started' button is located in the bottom right corner of the blue section. At the very bottom, there is a small yellow banner with the text 'Never miss a service outage again: Real-time'.

# MarkDown

Check out some other great profile pages!

<https://github.com/abhisheknaiidu/awesome-github-profile-readme>



The screenshot shows a GitHub repository page for 'abhisheknaiidu / awesome-github-profile-readme'. The page has a dark theme. At the top, there's a search bar and navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore. Below the header, the repository name 'abhisheknaiidu / awesome-github-profile-readme' is shown with a 'Public' status. The 'Code' tab is selected, showing the master branch with 15 branches and 0 tags. A list of files includes '.github', 'assets', 'LICENSE', 'README.md', and 'contributing.md', all updated 2 years ago. The 'About' section describes it as a curated list of awesome GitHub Profile READMEs, with a link to 'bit.ly/awesome-gh'. It lists tags like 'github', 'portfolio', 'awesome', 'awesome-list', 'github-readme', 'profile-readme', and 'github-profile-readme'. The 'Readme' section contains the text 'Awesome GitHub Profile README' and a 'Product Hunt' badge with a rank of 74. The 'Releases' section indicates 'No releases published'. The URL at the bottom is <https://github.com/abhisheknaiidu/awesome-github-profile-readme/actions>.