

mlab



Refilwe Mokgothu

CODETRIBE SKILLS
ACCELERATOR

Course - Introduction to Html

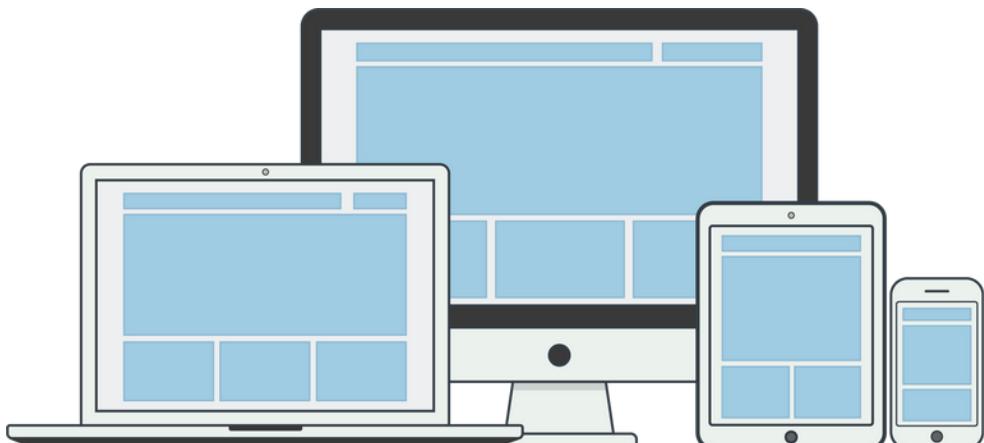
Bootcamp

Purpose

Is to up-skill and equip participants or everyone on these call with the foundation knowledge and practical skills needed to become a competent web developer

Web Development

Is the process of creating and maintaining websites & web applications



Why you should learn HTML

HTML = HyperText Markup Language

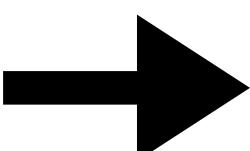
- Most basic building block of the web
- Allows you to add/change website content

HTML is useful for:

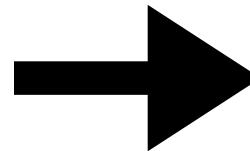
- Web developers
- Software developers
- Marketing professionals
- Sales
- Business owners
- Freelancers

Why you should Learn HTML

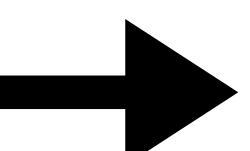
HTML



CSS



JS



Getting started:

What you will need

- Computer
- Text editor (VS Code, Notepad)
- Internet Browser
- Internet access (optional - for help debugging)

Environment Setup

- Installing a text editor (VS Code)
- VS Code into your Pc/Laptop

Creating your first project

- Create a new folder
- Create an HTML file, index.html
- Open the file in the web browser

Basic Structure of HTML

```
<!DOCTYPE html>
```

```
<html>
```

-

```
<head>
```

```
  <title>Title Goes Here</title>
```

```
</head>
```

```
<body>
```

```
  Page Content Goes Here
```

```
</body>
```



```
</html>
```

Key Learning Content

- 
- 1 Introduction to HTML
 - 2 HTML Tags
 - 3 Creating Links and Images
 - 4 List and Tables
 - 5 Forms and Inputs Elements
 - 6 Semantic HTML
 - 7 HTML 5

Introduction to Github

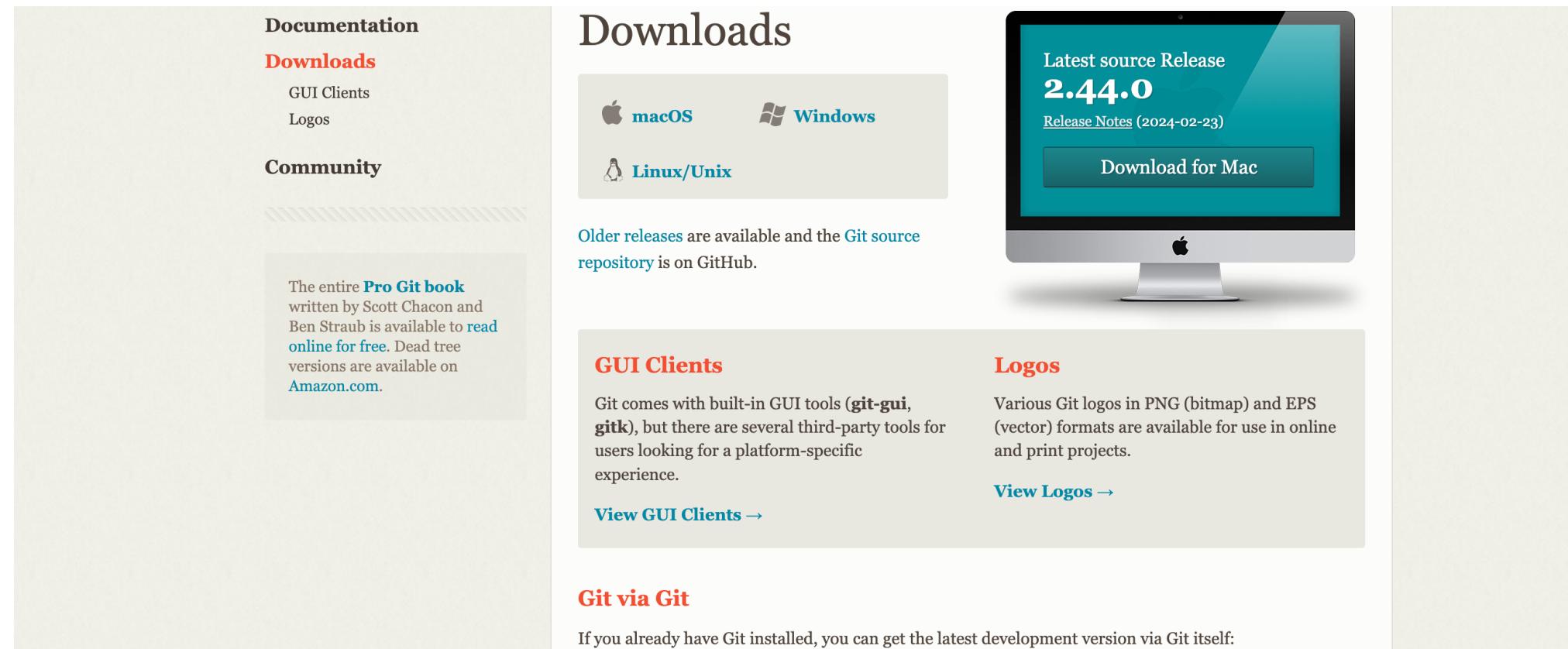
Git is a version control system

It's a system that records changes to our files over time

Many people can easily collaborate on a project and
have their own version of project files on their computer

STEP 1

Let us get started with the installation of github. Please refer to the git official website(<https://git-scm.com>). Download gitbash (<https://git-scm.com>)



You should see a screen like the one above. if you are using a windows machine, the computer screen at the bottom right will be a windows machine. Simply click on download and it will initiate the download of git.

After download has completed, make sure to click on the git file that you would have downloaded to start the installation process. Accept all so to complete the installation

STEP 2

We will be using our terminal to work with git. If you are working with a windows machine, your terminal is your CMD(Command prompt terminal). It can be located by clicking on your search bar on the bottom left corner of your screen then typing cmd on the search and clicking enter. You should then see a screen like the one on the **next page**.



- 1) On your terminal (git bash, Windows CMD, or preferred CLI)
- 2) We want to first initialize our GitHub account globally on our workstation(computer/laptop). Therefore we are required to add the username, email and password.
In your terminal(CMD);
 - Enter git config --global user.name "Refilwe". Where there is John, place your GitHub username. Make sure the username is in quotation marks.
 - Enter git config --global user.email "Refilwe@mlab.co.za". Make sure the email address is in quotation marks
 - Enter git config --global user.password "*****". Make sure the password in quotation marks.

We done with git installion , NOW WE NEED TO PUSH OUR UI PROJECT TO GITHUBLOG IN GITHUB ACCOUNT .

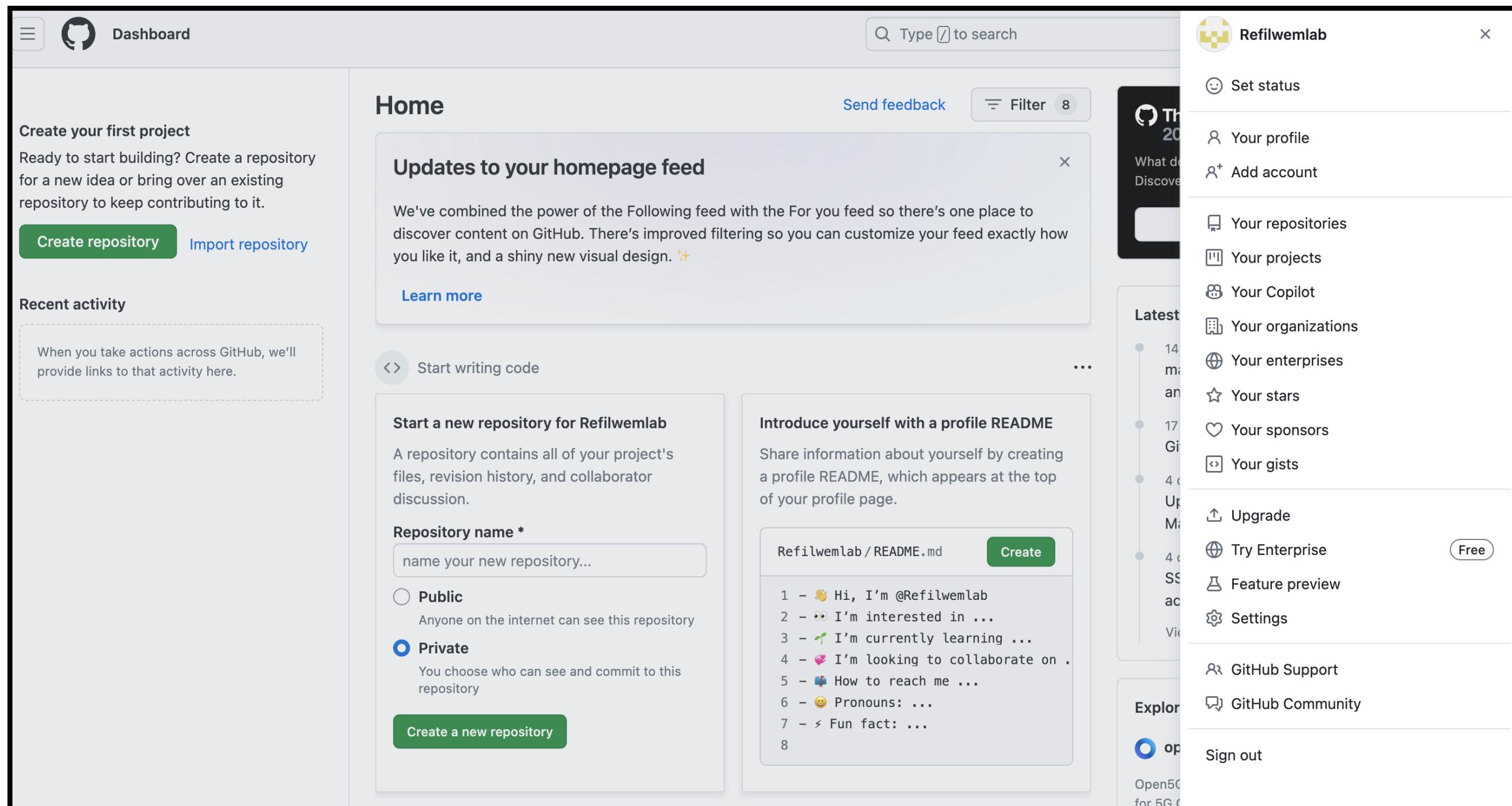
A screenshot of a Google search results page for the query "github". The search bar at the top shows the query "github". Below the search bar, there are filters for "All", "Images", "Videos", "News", "Shopping", and "More". To the right of these filters are "Tools" and "SafeSearch" dropdown menus. The search results indicate "About 785,000,000 results (0.30 seconds)". The first result is the official GitHub homepage, titled "GitHub: Let's build from here · GitHub". The description below the title reads: "GitHub is where over 100 million developers shape the future of software, together. Contribute to the open source community, manage your Git repositories, ...". Below the main result, there are several other links: "Login", "Join", "GitHub Desktop", "Explore GitHub", and "About". On the right side of the search results, there is a detailed sidebar with the following information:

GitHub	
Software developer	:
 github.com	
GitHub is a developer platform that allows developers to create, store, manage and share their code. It uses Git software, providing the distributed version control of Git plus access control, bug tracking, software feature requests, task management, continuous integration, and wikis for every project. Wikipedia	
Founders: Chris Wanstrath , Scott Chacon , Tom Preston-Werner , P. J. Hyett	
Headquarters: San Francisco, California, United States	
CEO: Thomas Dohmke (15 Nov 2021–)	
Founded: 2008, San Francisco, California, United States	
Subsidiary: npm, Inc.	
Employees: 5,595	
Parent: Microsoft	

Disclaimer

STEP 3

After you have logged into your github account. You should see a screen similar to the screen below. If not then go to the top right section of your screen, click on the last dropdown. Select "Your Profile". This will be where you are able to see all the repositories you created.



STEP 4

Now to create your repository, simply click on the add icon on the top right corner then select "New Repository". You should see a screen similar to the one below.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner * Refilwemlab | **Repository name *** Bootcamp Bootcamp is available.

Great repository names are short and memorable. Need inspiration? How about [supreme-octo-umbrella](#) ?

Description (optional)

 **Public** Anyone on the internet can see this repository. You choose who can commit.
  **Private** You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

 You are creating a public repository in your personal account.

The repository name field(Always Required) is where you will set the name of your repository. For this scenario we will name our repository as "Bootcamp". In the description input enter "My first repo ". Refer to the screenshot below.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner * Repository name *

 Refilwemlab | Bootcamp

 Bootcamp available.

Great repository names are short and memorable. Need inspiration? How about [supreme-octo-umbrella](#) ?

Description (optional)

 **Public**
Anyone on the internet can see this repository. You choose who can commit.

 **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file
This is where you can write a long description for your project. [Learn more about READMEs](#).

Now simply click on the "Create repository" button.

Owner * Repository name *

 Refilwemlab / 
 Bootcamp  available.

Great repository names are short and memorable. Need inspiration? How about [supreme-octo-umbrella](#) ?

Description (optional)

 **Public**
Anyone on the internet can see this repository. You choose who can commit.

 **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file
This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore



Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license



A license tells others what they can and can't do with your code. [Learn more about licenses](#).

 You are creating a public repository in your personal account.

Congratulations you have created your first github repo(repository). In this scenario, we have no need to create the readme file. The screen you should see now should be like the one below.

Quick setup — if you've done this kind of thing before

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/Refilwemlab/Bootcamp.git> [Copy](#)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# Bootcamp" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Refilwemlab/Bootcamp.git
git push -u origin main
```

[Copy](#)

...or push an existing repository from the command line

```
git remote add origin https://github.com/Refilwemlab/Bootcamp.git
git branch -M main
git push -u origin main
```

[Copy](#)

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

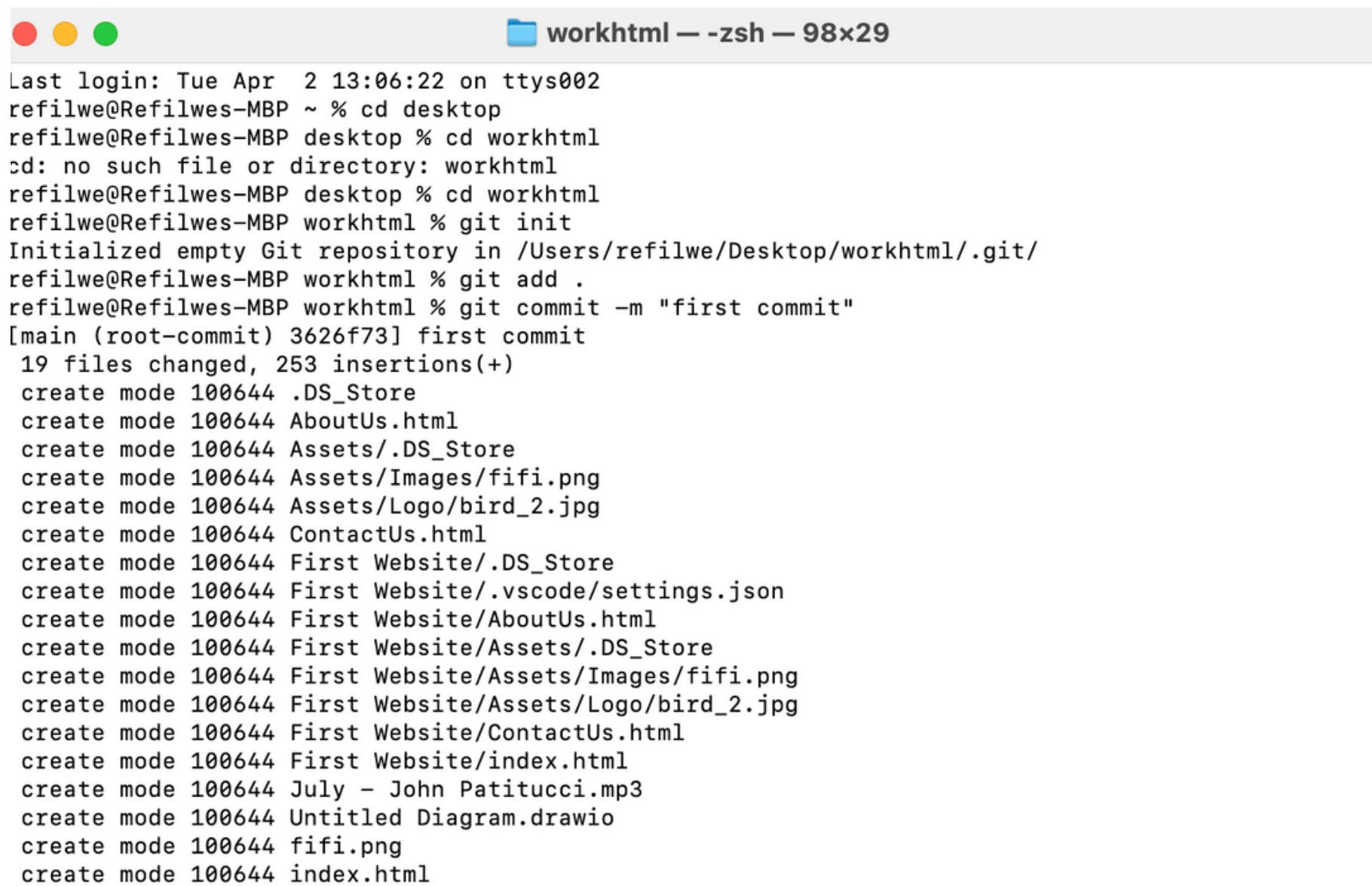
[Import code](#)

 **ProTip!** Use the URL for this page when adding GitHub as a remote.

NOW CODETRIBERS NAVIGATE TO A DIRECTORY WHERE YOUR PROJECTED IS LOCATED ON CMD

FOR EXAMPLE IN MY CASE MY PROJECT IS ON DESKTOP UNDER A FOLDER CALLED BootCamp...SO ON CMD

- cd desktop
- cd BootCamp



The screenshot shows a terminal window with the title "workhtml — zsh — 98x29". The window contains the following text:

```
Last login: Tue Apr  2 13:06:22 on ttys002
refilwe@Refilwes-MBP ~ % cd desktop
refilwe@Refilwes-MBP desktop % cd workhtml
cd: no such file or directory: workhtml
refilwe@Refilwes-MBP desktop % cd workhtml
refilwe@Refilwes-MBP workhtml % git init
Initialized empty Git repository in /Users/refilwe/Desktop/workhtml/.git/
refilwe@Refilwes-MBP workhtml % git add .
refilwe@Refilwes-MBP workhtml % git commit -m "first commit"
[main (root-commit) 3626f73] first commit
 19 files changed, 253 insertions(+)
 create mode 100644 .DS_Store
 create mode 100644 AboutUs.html
 create mode 100644 Assets/.DS_Store
 create mode 100644 Assets/Images/fifi.png
 create mode 100644 Assets/Logo/bird_2.jpg
 create mode 100644 ContactUs.html
 create mode 100644 First Website/.DS_Store
 create mode 100644 First Website/.vscode/settings.json
 create mode 100644 First Website/AboutUs.html
 create mode 100644 First Website/Assets/.DS_Store
 create mode 100644 First Website/Assets/Images/fifi.png
 create mode 100644 First Website/Assets/Logo/bird_2.jpg
 create mode 100644 First Website/ContactUs.html
 create mode 100644 First Website/index.html
 create mode 100644 July - John Patitucci.mp3
 create mode 100644 Untitled Diagram.drawio
 create mode 100644 fifi.png
 create mode 100644 index.html
```

ON YOUR CMD TYPE THE FOLLOWING COMMAND STEP BY STEP

- git init
- git add *
- git commit -m "first commit"
- git remote add origin <https://github.com/Refilwemlab/TESTRE.git> -----NB -----PUT
YOUR STUFF ON URL
- git push -u origin master

WE DONE-----WELL DONE

