

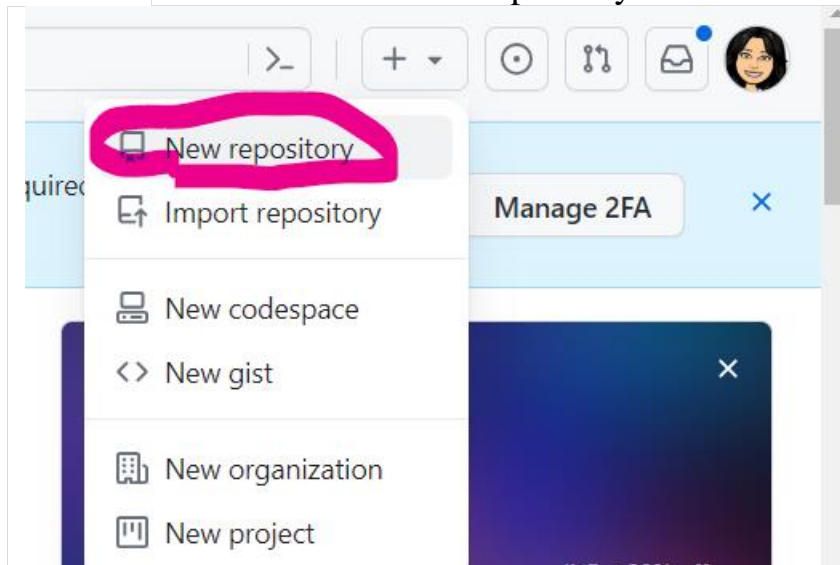
Uploading CSC110, Intro to Computer Programming! Lab Exercises to GitHub

1. Create a GitHub Account:

- If you don't already have a GitHub account, you can create one at <https://github.com/>.

2. Create a New Repository:

- Click on the '+' sign at the top right corner of your GitHub homepage and select "New Repository."



- Give your repository a name in proper title case, like

"CSC110--Intro-to-Computer-Programming-Lab-Exercises."

Create a new repository

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

Required fields are marked with an asterisk (*).

Repository template

No template

Start your repository with a template repository's contents.

Owner *

solmazsm

Repository name *

CSC110--Intro-to-Compu

✓ CSC110--Intro-to-Computer-Programming-Lab-Exercises. is available.

Great repository names are short and memorable. Need inspiration? How about [verbose-fiesta](#) ?


3. Initialize the Repository:


- Choose whether you want your repository to be public or private (based on your preference and GitHub plan).
- Optionally, add a README file to your repository (it's a good practice to have one).

Great repository names are short and memorable. Need inspiration? How about `verbose-fiesta` ?

Description (optional)

CSC110--Intro-to-Computer-Programming-Lab-Exercises.

☒  **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

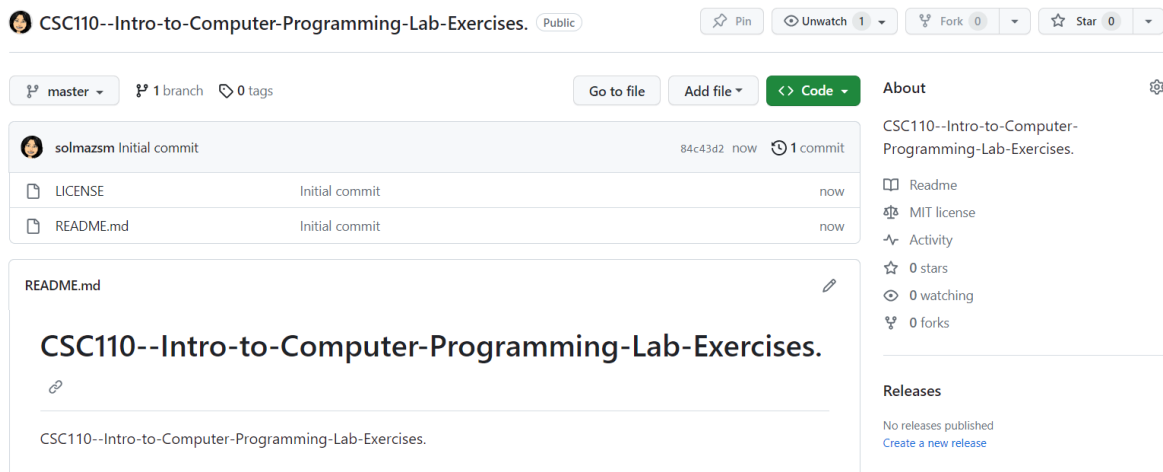
License: **MIT License**

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

This will set `master` as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

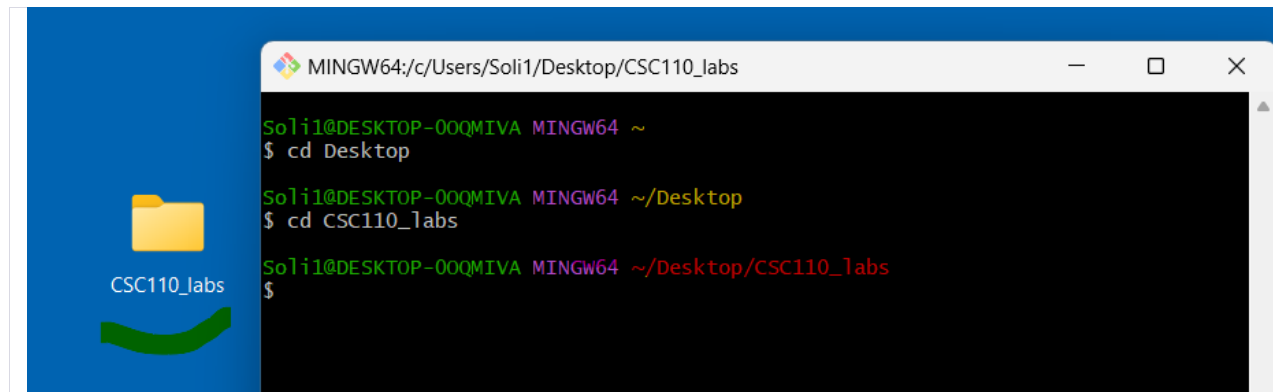
Create repository



The screenshot shows the GitHub repository page for 'CSC110--Intro-to-Computer-Programming-Lab-Exercises'. The repository is public and has 1 branch (master) and 0 tags. It contains 1 commit (84c43d2) with files LICENSE and README.md. The README.md file is displayed with the title 'CSC110--Intro-to-Computer-Programming-Lab-Exercises.' and the content 'CSC110--Intro-to-Computer-Programming-Lab-Exercises.' The right sidebar shows repository statistics: 0 stars, 0 watching, and 0 forks. There are also links for Readme, MIT license, Activity, and Releases.

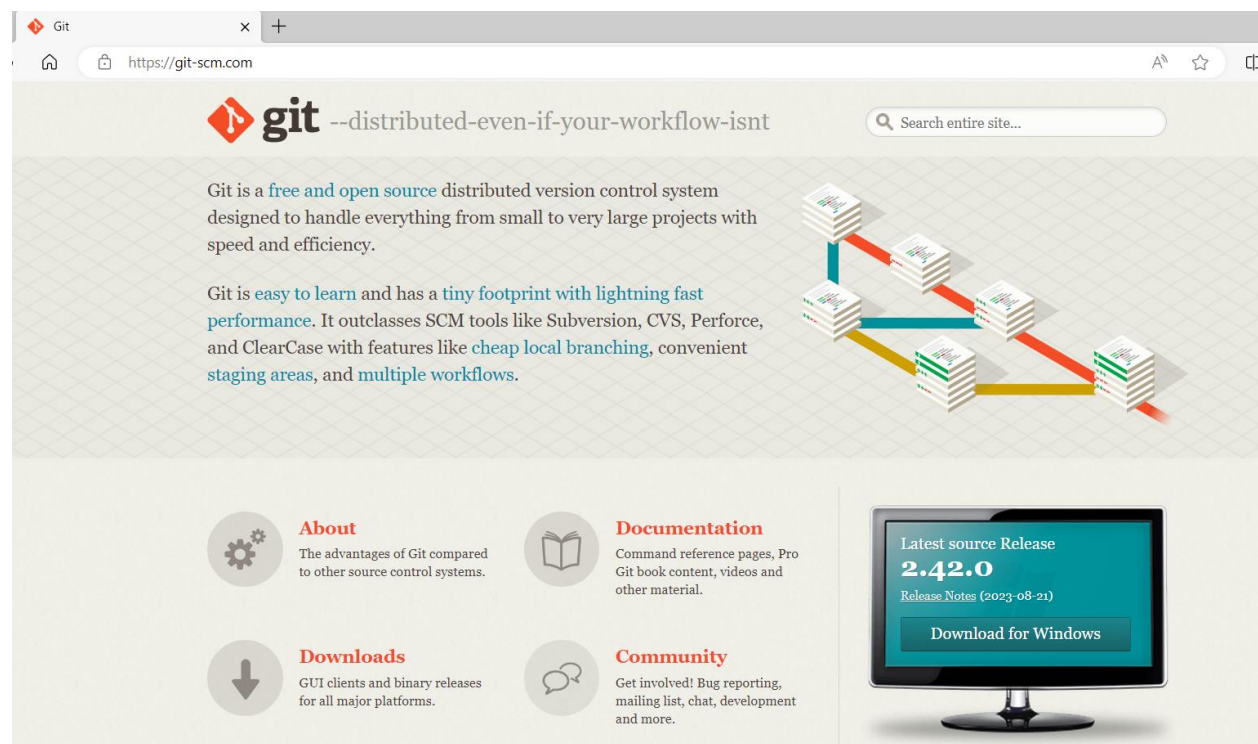
4. Create a Local Git Repository:

- Open a terminal or command prompt on your local computer.
- Navigate to the directory where your CSC110 lab exercises are stored using the `cd` command.



Installing Git Bash

Download the Git Bash setup from the official website: <https://git-scm.com/>



5. Initialize Git:

- Run the following commands to initialize a Git repository and connect it to your GitHub repository:

git init

Adds the file to your local repository and stages it for commit. To unstage a file, use

git add .

Commits the tracked changes and prepares them to be pushed to a remote repository.

git commit -m "Initial commit"

6. Add a Remote Repository:

- Link your local repository to the GitHub repository by running the following command, replacing `<your-username>` with your GitHub username and `<repository-name>` with your repository's name:

git remote add origin <https://github.com/<yourusername>/<repository-name>.git>

The screenshot displays a GitHub repository page for "CSC110--Intro-to-Computer-Programming-Lab-Exercises." The repository is public and has one branch, "master". The commit history shows an "Initial commit" by user "solmazsm" with files "LICENSE" and "README.md". A "Clone" dropdown menu is open, showing the "HTTPS" URL: `https://github.com/solmazsm/CSC110--Intro-to-`. Below the repository page, a terminal window shows the following commands and output:

```
MINGW64/c/Users/Soli1/Desktop/CSC110_labs
Solli1@DESKTOP-00QMIVA MINGW64 ~
$ cd Desktop
Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop
$ cd CSC110_labs
Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs
$ git init
Initialized empty Git repository in C:/Users/Soli1/Desktop/CSC110_labs/.git/
Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git add .
Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git commit -m "Initial commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)
Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git remote add origin https://github.com/solmazsm/CSC110--Intro-to-Computer-Programming-Lab-Exercises..git
```

```
MINGW64:/c:/Users/Soli1/Desktop/CSC110_labs
Solli1@DESKTOP-00QMIVA MINGW64 ~
$ cd Desktop

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop
$ cd CSC110_labs

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs
$ git init
Initialized empty Git repository in C:/Users/Soli1/Desktop/CSC110_labs/.git/

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git add .

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git commit -m "Initial commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$ git remote add origin https://github.com/solmazsm/CSC110--Intro-to-Computer-Programming-Lab-Exercises..git

Solli1@DESKTOP-00QMIVA MINGW64 ~/Desktop/CSC110_labs (master)
$
```

7. Push Your Code:

- Push your code to GitHub with the following command:

```
git push -u origin master
```

8. Organize Your Files:

- Make sure that all your lab exercises and files follow the capitalization pattern. Rename them if necessary.

9. Commit and Push Changes:

- After organizing your files, commit and push the changes to GitHub:

```
git add .
```

```
git commit -m "Organized file names"
```

```
git push
```

10. Repeat for Each Lab Exercise:

By following these steps, you can successfully upload your "CSC110, Intro to Computer Programming! Lab Exercises" to GitHub while ensuring proper capitalization in the repository name and file names. This will enable you to share your work and collaborate with others more effectively.

[Link](#)