

## GUIDE

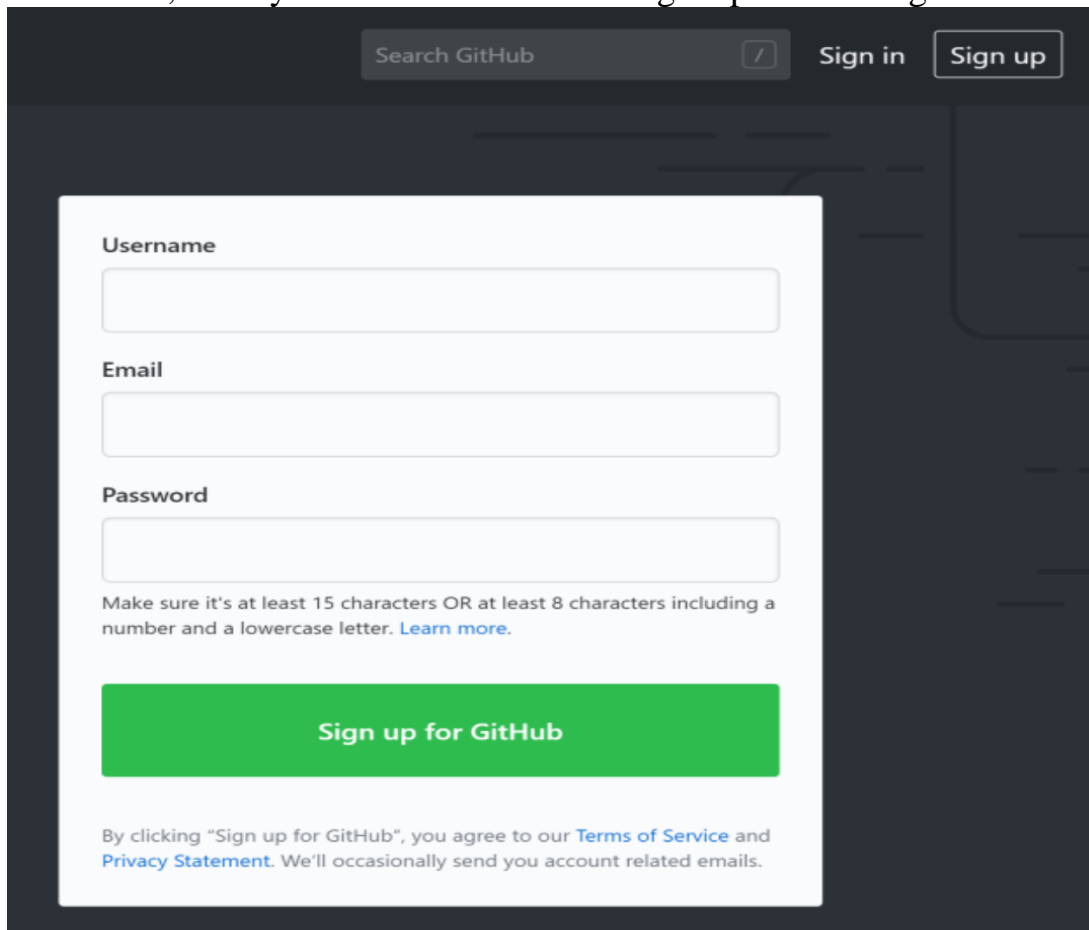
### TO DEPLOYING A REACT APP TO GITHUB PAGES

GitHub Pages is a free service from GitHub that allows you to host a website directly from a repository on GitHub. It is a great tool for deploying static websites such as blogs, personal pages, or project documentation.

#### I. Create a Repository on GitHub.

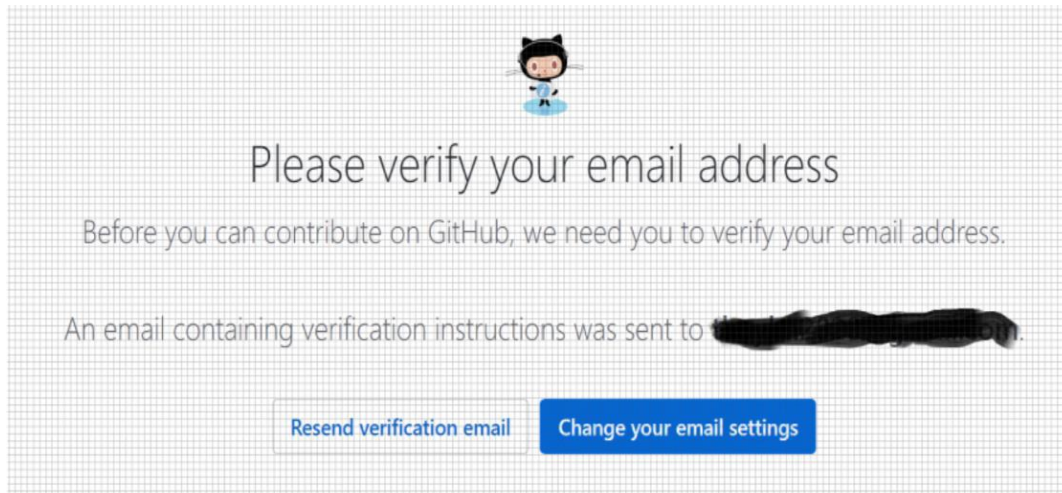
##### 1. Create an account and log in to GitHub.

Go to <https://github.com/> to log in or sign up for an account. If you don't have an account, enter your information in the Sign Up form to register.

A screenshot of the GitHub Sign Up form. At the top, there is a search bar labeled "Search GitHub" and buttons for "Sign in" and "Sign up". The main form is a white box with a dark background. It contains three input fields: "Username", "Email", and "Password". Below the "Password" field, there is a note: "Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)". At the bottom of the form is a large green button labeled "Sign up for GitHub". Below the button, there is a line of text: "By clicking 'Sign up for GitHub', you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails."

After providing all the required information and completing the registration process.

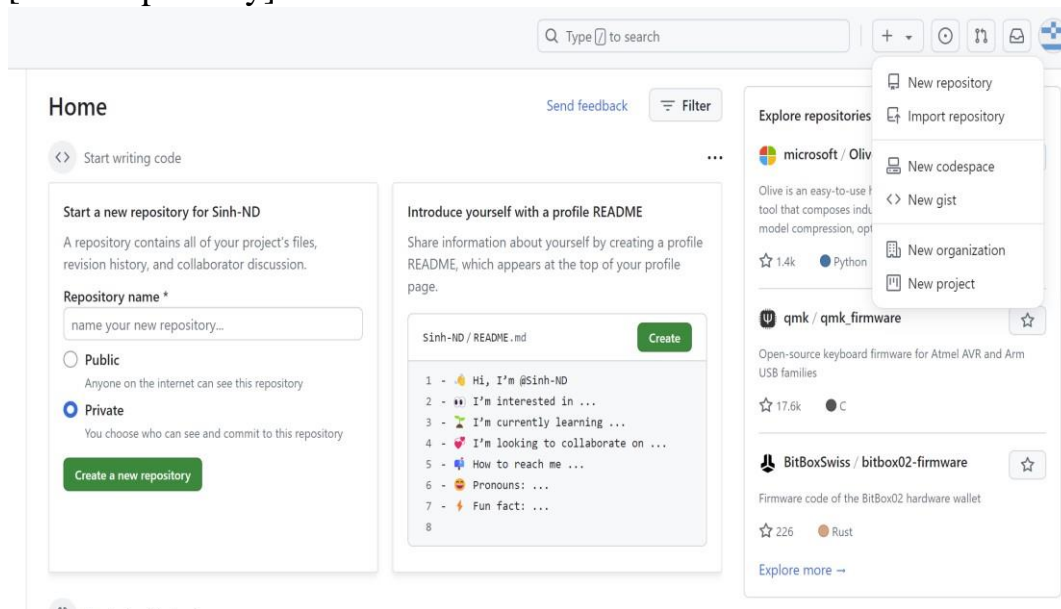
# GITHUB SETUP



Check your email to see if GitHub has sent a verification link. If so, click on [Verify email address] to proceed with verification. If you haven't received it, go back to the GitHub homepage and click [Resend verification email].

## 2. Create a New Repository.

- Click the plus sign icon in the upper-right corner of the screen and select [New Repository].




- Name your repository, for example: climbing36e
- Choose between Public or Private depending on your needs.
- Click Create repository to complete the process.

## 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 \*

 dohung95 ▾

Repository name \*

Nobel Prize Winner

✓ Your new repository will be created as Nobel-Prize-Winner-.

The repository name can only contain ASCII letters, digits, and the characters ., -, and \_.

Great repository names are short and memorable. Need inspiration? How about **sturdy-octo-spork** ?

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.

## II. Deploying a React App to GitHub Pages

Open the terminal in Visual Studio Code:

### 1. Install GitHub Pages

```
npm install --save gh-pages
```

### 2. Add to package.json File

```
"homepage": "https://dohung95.github.io/Nobel-Prize-Winner/",
```

```
"predeploy": "npm run build",
```

```
"deploy": "gh-pages -d build".
```

```
{
  "homepage": "https://dohung95.github.io/Nobel-Prize-Winner",
  "name": "session04-parents-child",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@emotion/react": "^11.13.3",
    "@emotion/styled": "^11.13.0",
    "@mui/material": "^6.1.2",
    "@testing-library/jest-dom": "^5.17.0",
    "@testing-library/react": "^13.4.0",
    "@testing-library/user-event": "^13.5.0",
    "aos": "^2.3.4",
    "bootstrap": "^5.3.3",
    "chart.js": "^4.4.4",
    "leaflet": "^1.9.4",
    "react": "^18.3.1",
    "react-bootstrap": "^2.10.4",
    "react-chartjs-2": "^5.2.0",
    "react-dom": "^18.3.1",
    "react-leaflet": "^4.2.1",
    "react-responsive": "^10.0.0",
```

```
"scripts": {  
  "predeploy": "npm run build",  
  "deploy": "gh-pages -d build",  
  "start": "react-scripts start",  
  "build": "react-scripts build",  
  "test": "react-scripts test",  
  "eject": "react-scripts eject"  
},  
"eslintConfig": {  
  "extends": [  
    "react-app",  
    "react-app/jest"  
  ]  
},  
"browserslist": {  
  "production": [  
    ">0.2%",  
    "not dead",  
    "not op_mini all"  
  ],  
  "development": [  
    "last 1 chrome version",  
    "last 1 firefox version",  
    "last 1 safari version"  
  ]  
}
```

### 3. Push Code to GitHub

- git init
- git add .
- git commit -m "first commit"
- git branch -M main
- git remote add origin <https://github.com/dohung95/Nobel-Prize-Winner.git>
- git push -u origin main

### 4. Run the deploy command

- npm run deploy

## GITHUB SETUP

After a successful deployment, go back to GitHub: Open your repository on GitHub, click on the "Settings" tab at the top, and scroll down to the "Pages" section. Wait for about 1-5 minutes and you should see the results as shown below.

