Walkthrough of Installing Everything Needed for AI-Assisted-Coding-In-R Workshop

2025-11-04

Table of contents

**If you are having any issues with installing anything for the workshop, think about** [**submitting a consulting request**](https://dlab.berkeley.edu/consulting/submit-consulting-request) **where D-Lab consultants can help you get prepared for your workshop.**

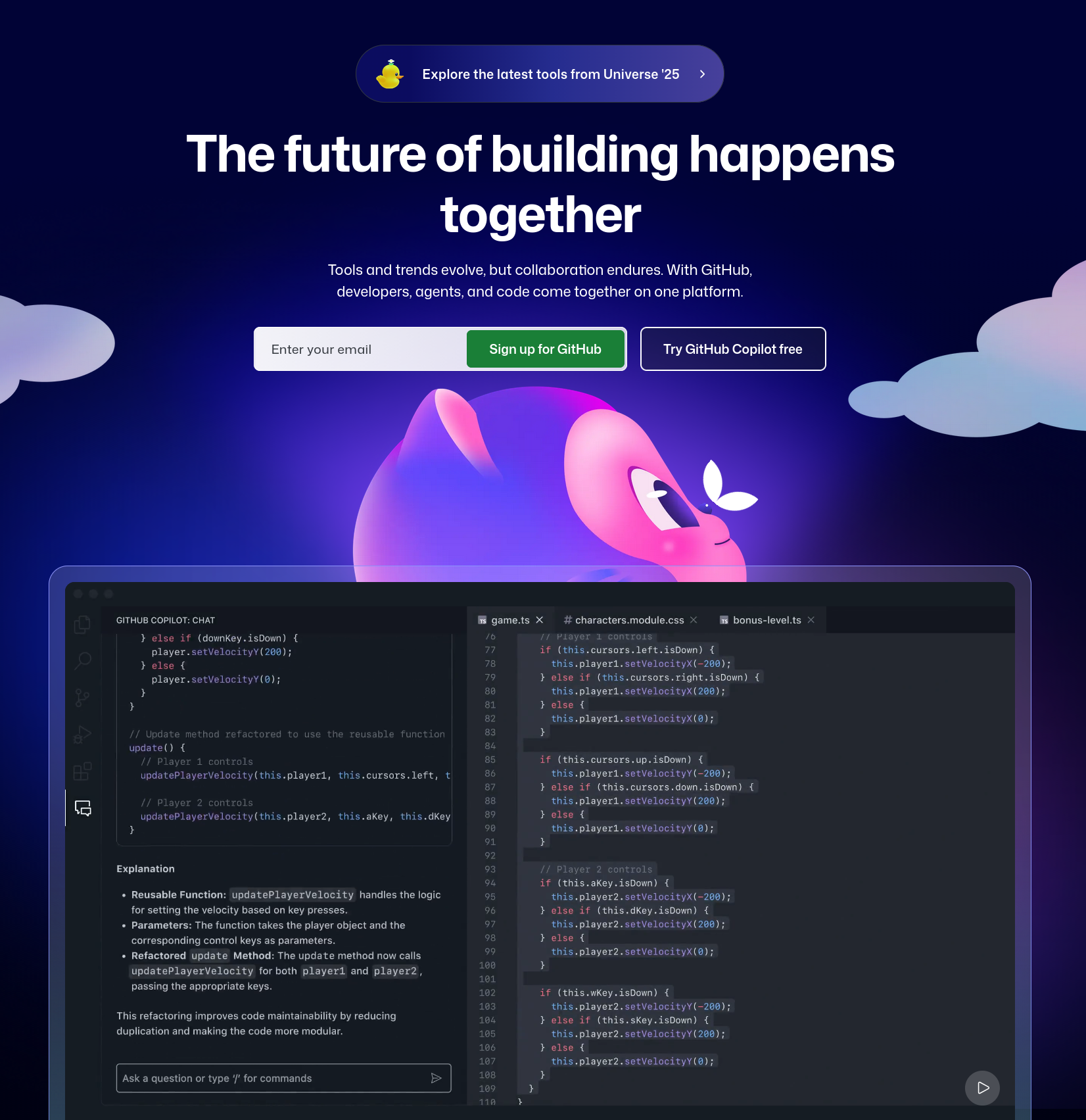
# 1. Checklist of Everything Needed to Install for AI-Assisted-Coding-In-R

* [] Install GitHub
* [] Sign up for GitHub Copilot
* [] Install R
* [] Install Visual Studio (VS) Code
* [] Adjust VSCode to work with R
* [] Download Zip file

When these are completed, you are ready for your AI-Assisted-Coding-In-R workshop.

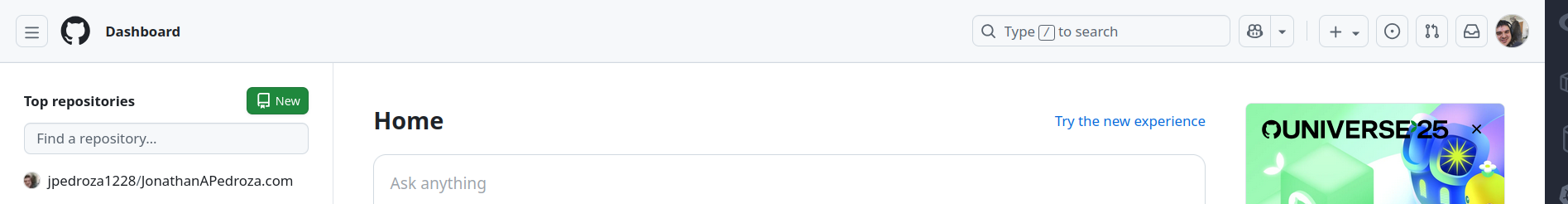
# 2. Installing GitHub

Go to [GitHub](https://github.com/) and sign up for a GitHub account.



At this page, you will sign up for GitHub with your @berkeley.edu email account. Follow the directions to verify your account. Below are some recommendations for creating a username (inspired by https://happygitwithr.com/github-acct).

* Use part of your real name so it is easier for people to know who you are
* Try and keep it short, you may have to type it a lot
* Keep everything lowercase. If you really want to separate words, use a hyphen (-) or an underscore (\_)

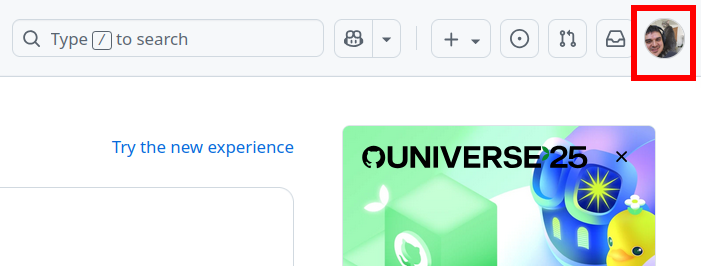


Once you sign in, you will be at your dashboard. You have now downloaded GitHub! Congrats!

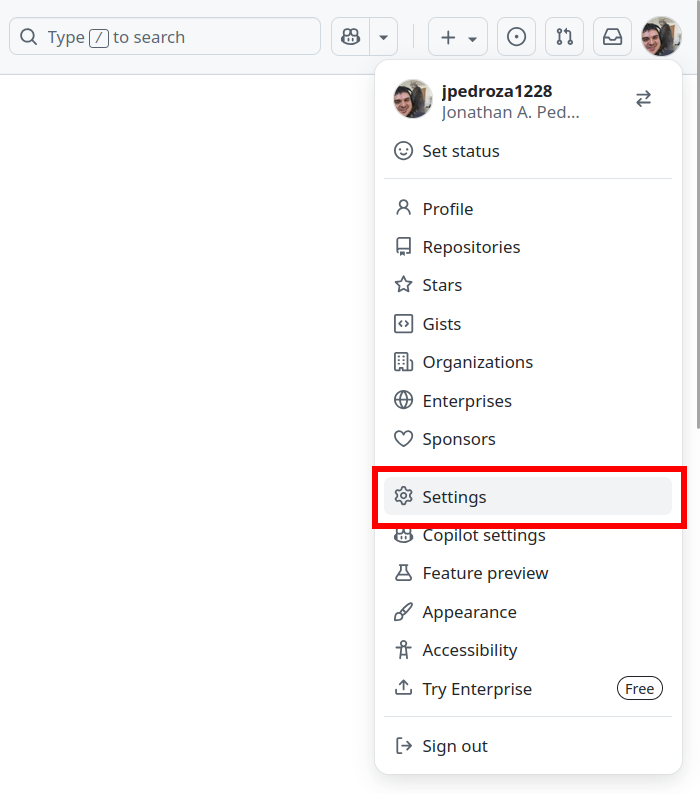
## 2.1 Optional: Two-Factor Authentication (2FA)

While not part of the tutorial, GitHub can house public and private data. If you are planning on continuing to use GitHub, please think about creating additional safeguards by setting up your Two-Factor Authentication (2FA). Below are some resources and a quick look into 2FA.

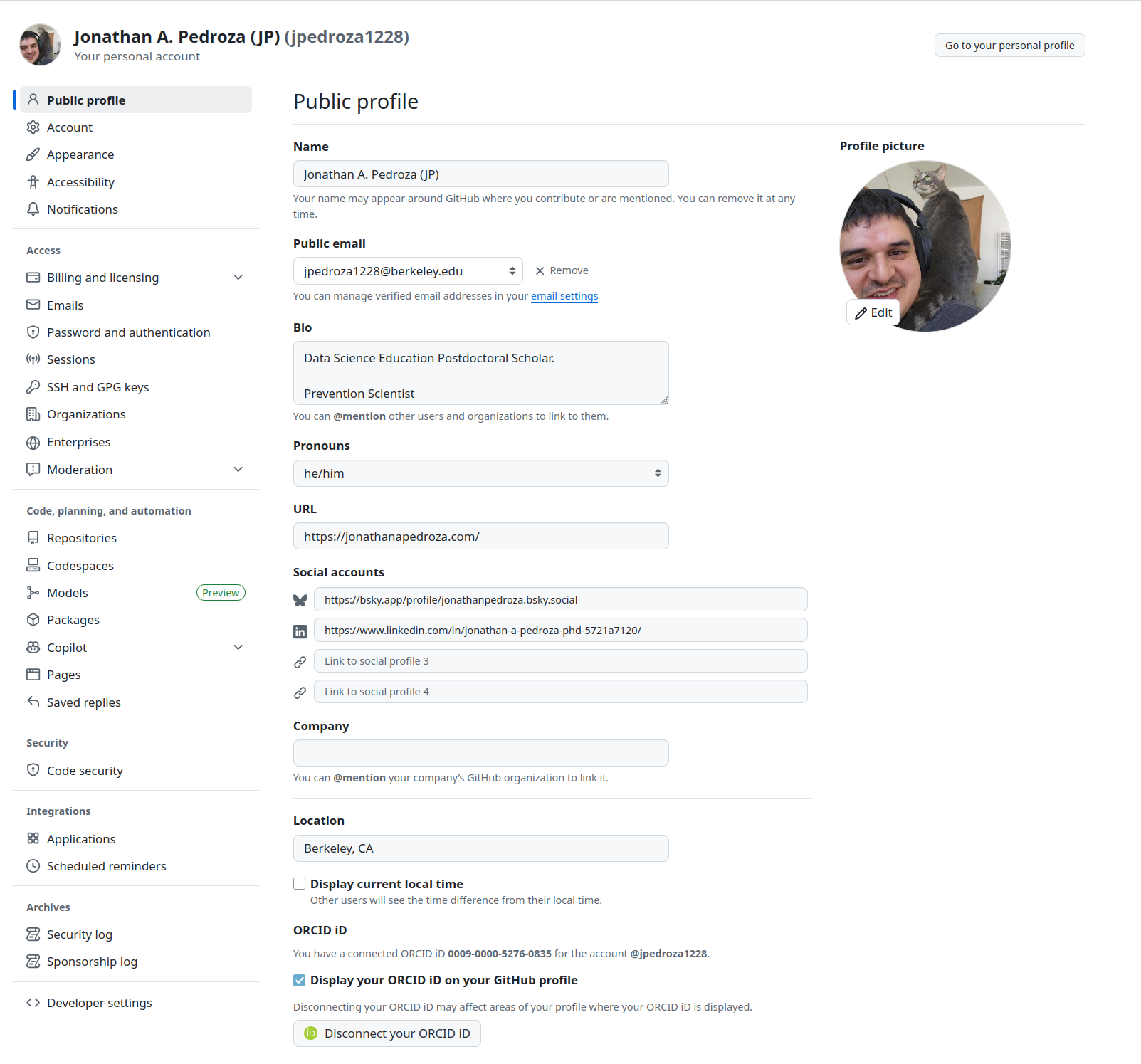
For more information on 2FA, you can find resources [here (About 2FA)](https://docs.github.com/en/authentication/securing-your-account-with-two-factor-authentication-2fa/about-two-factor-authentication) or [here (Securing account with 2FA)](https://docs.github.com/en/authentication/securing-your-account-with-two-factor-authentication-2fa).



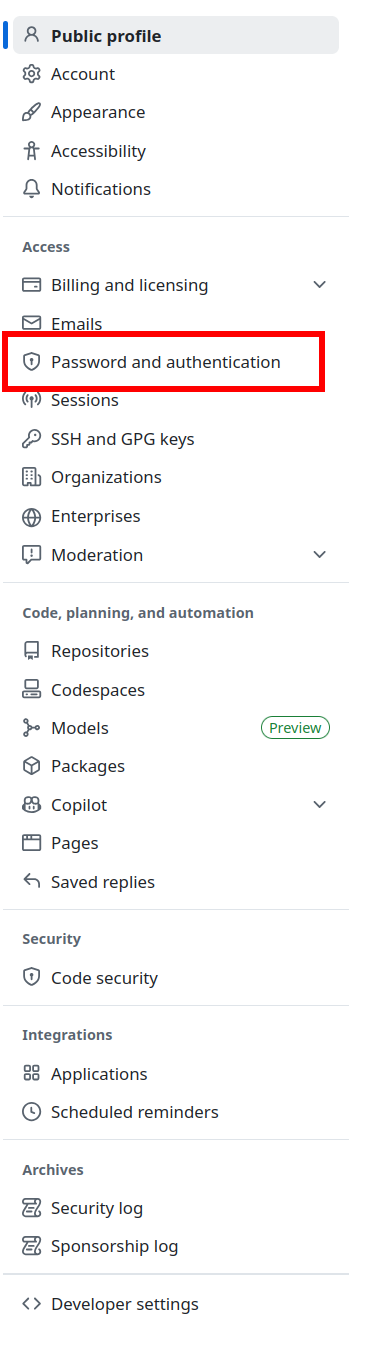
From your dashboard, you will want to to to your profile. If you just created your account, you will not have a profile picture. You’ll then click on your profile circle to show a dropdown menu.



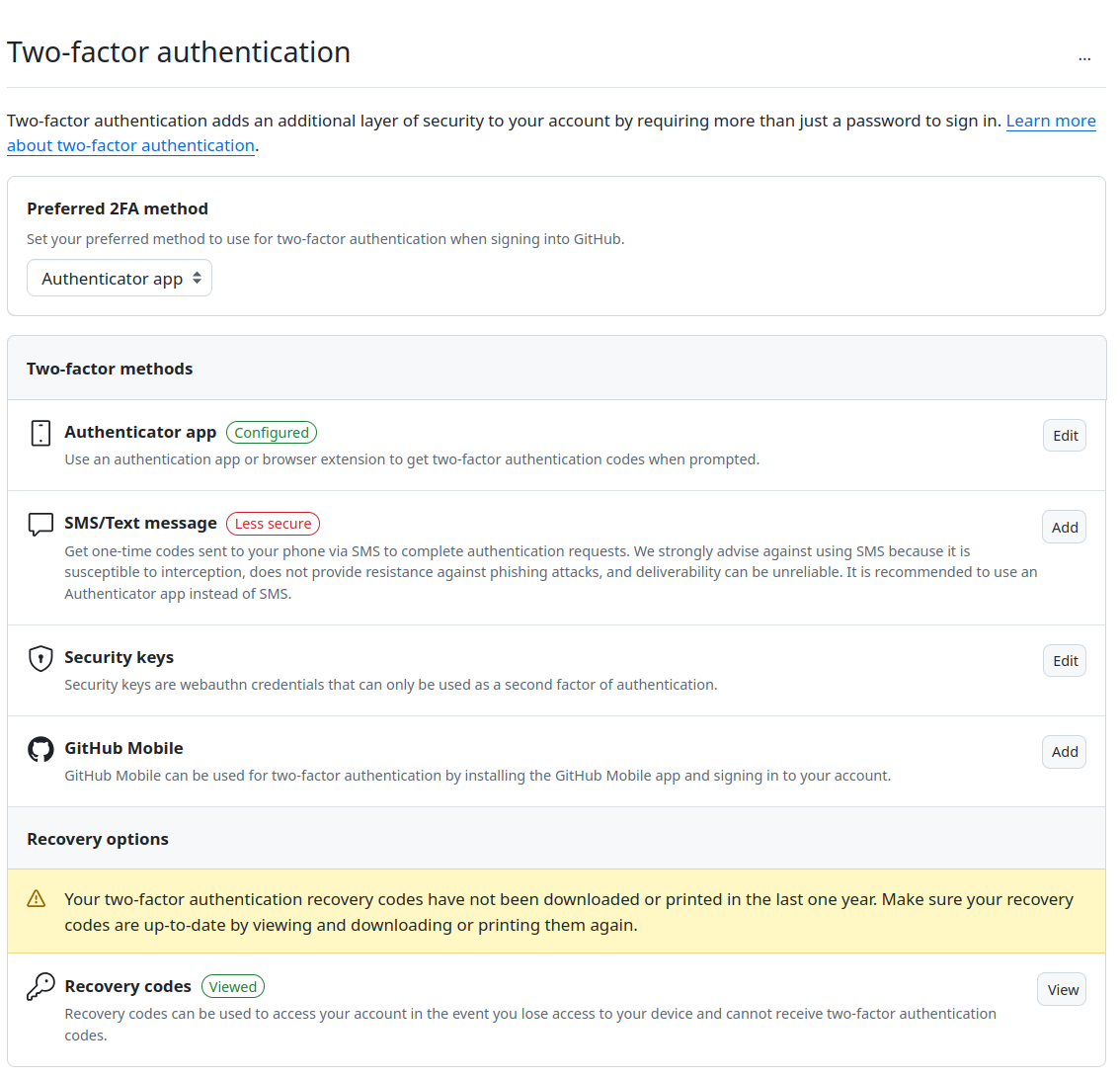
At this dropdown menu, you will go to your Settings.



At your settings, there will be a lot of information.



On the left sidebar, you will want to go to Password and authentication. Here you can customize how you would like to sign into GitHub.



While there are options for 2FA, I would recommend using an authenticator app. So every time you sign in (among other actions on GitHub), you will sign in with your username and password then verify it with a code from your authenticator app.

* ☒ Install GitHub
* [] Sign up for GitHub Copilot
* [] Install R
* [] Install Visual Studio (VS) Code
* [] Adjust VSCode to work with R
* [] Download Zip file

# 3. Sign Up for GitHub Copilot

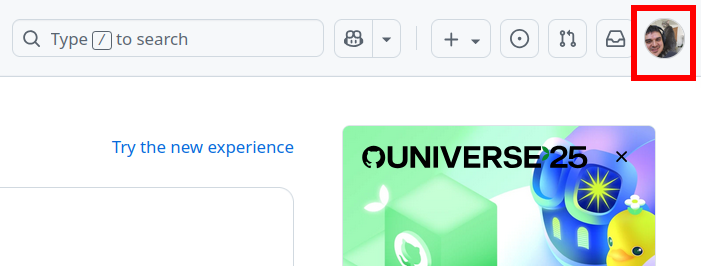
Signing up for GitHub Copilot will depend on whether you plan to sign up for the free version, CoPilot *Pro* using verified information, or pay for a Pro plan ([see pricing information here](https://github.com/features/copilot/plans)).

## 3.1 Free Version

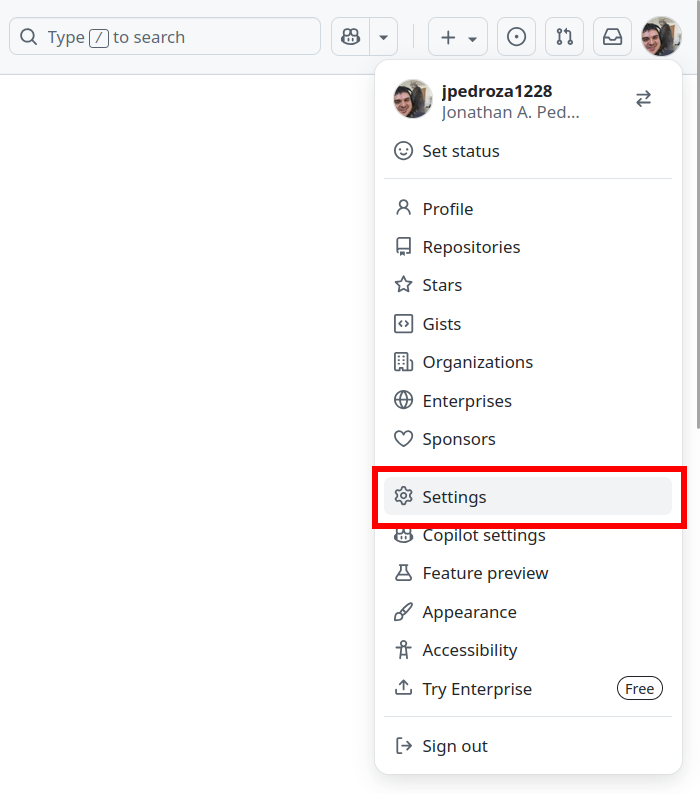
The free version of GitHub Copilot comes with VSCode. You can install the necessary extensions in the following section. You can also try out a [30-day trial here](https://github.com/github-copilot/pro).

## 3.2 Copilot Pro (Instructions for Verification)

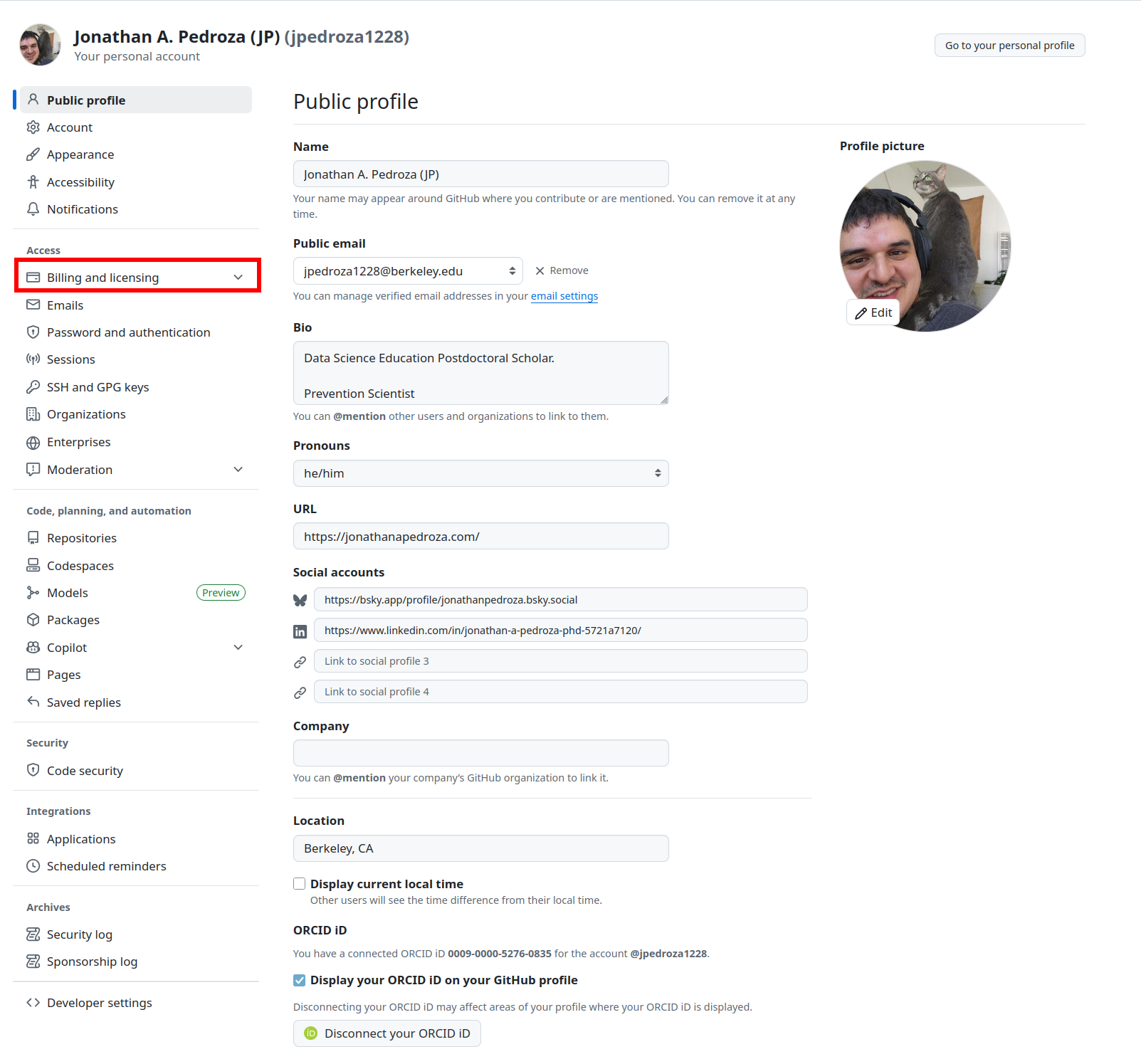
To get Copilot *Pro* for teachers and students (for free), you will need to follow the following steps. Below are some steps from the optional 2FA section above.



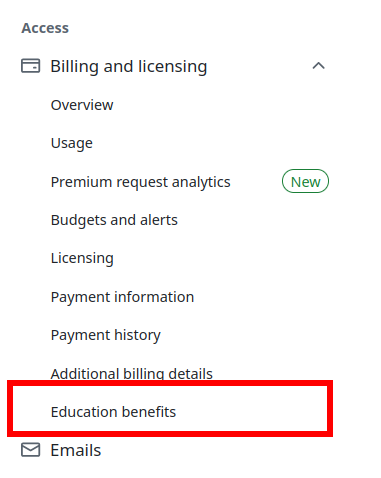
From your dashboard, you will want to to to your profile. If you just created your account, you will not have a profile picture. You’ll then click on your profile circle to show a dropdown menu.



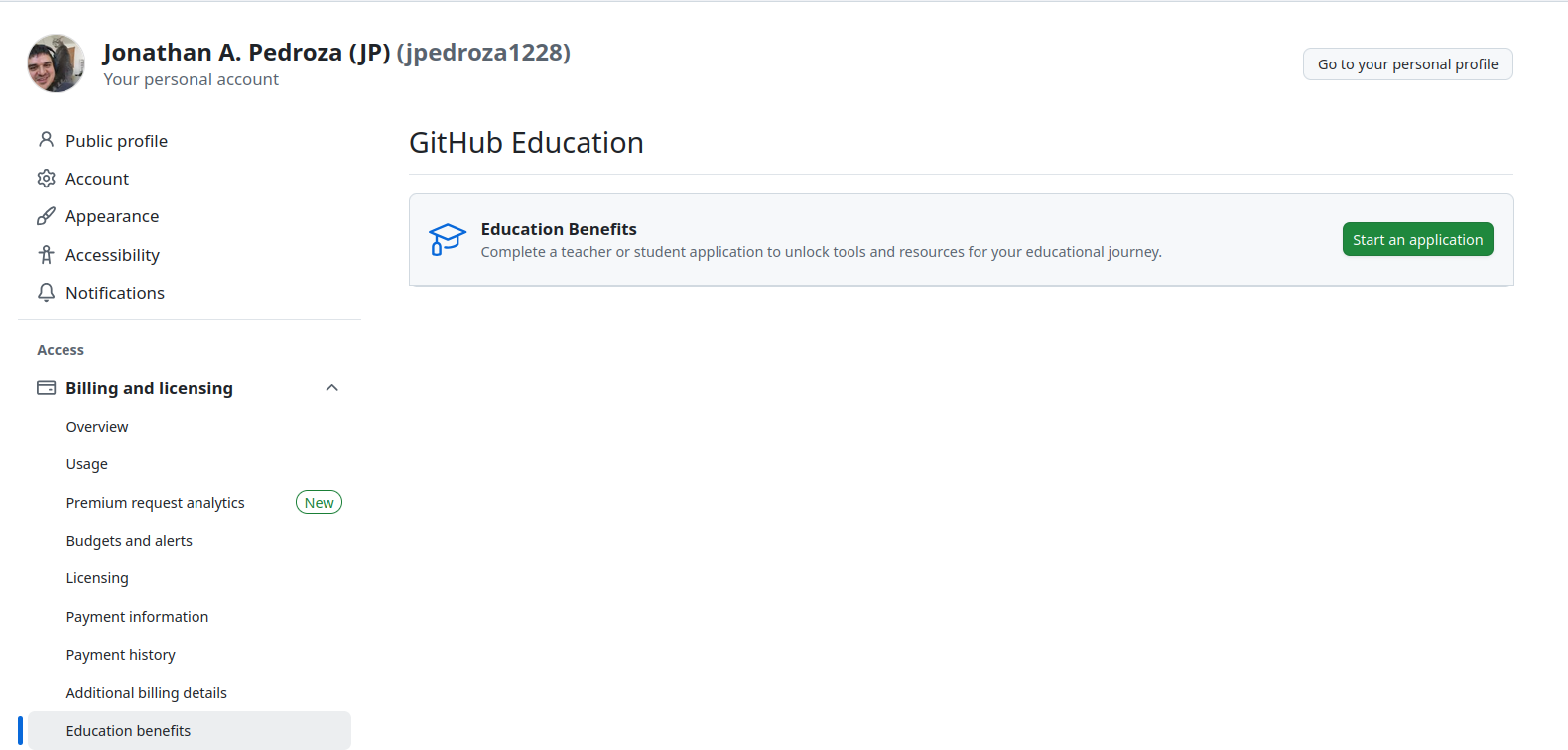
At this dropdown menu, you will go to your Settings.



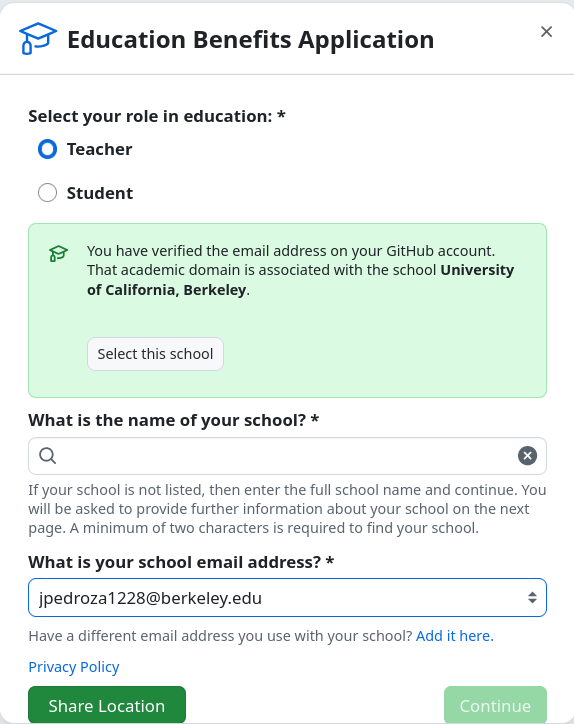
At your settings, there will be a lot of information. This time, you will click on the dropdown menu for Billing and licensing.



From the dropdown menu, you will click on Education benefits.

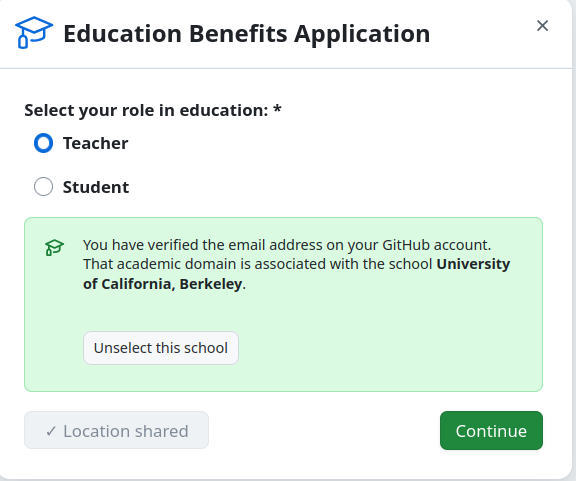


Clicking on Education benefits will take you to GitHub Education. There you will start your application to get additional benefits, including GitHub Copilot Pro.



Once you start your application, you will have the option of choosing your role At UC Berkeley. Below are general instructions for teachers and students; however, the instructions below will start to shift toward specific instructions for teachers.

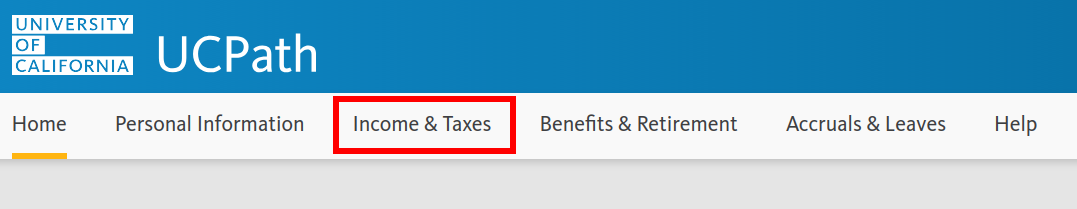
On the application, you can include the full name of the university (University of California, Berkeley) and when given an option to Select a school, click Select this school. You will also include your school email address in the dropdown menu. Make sure you are including your @berkeley.edu email address since verification relies on that email address.



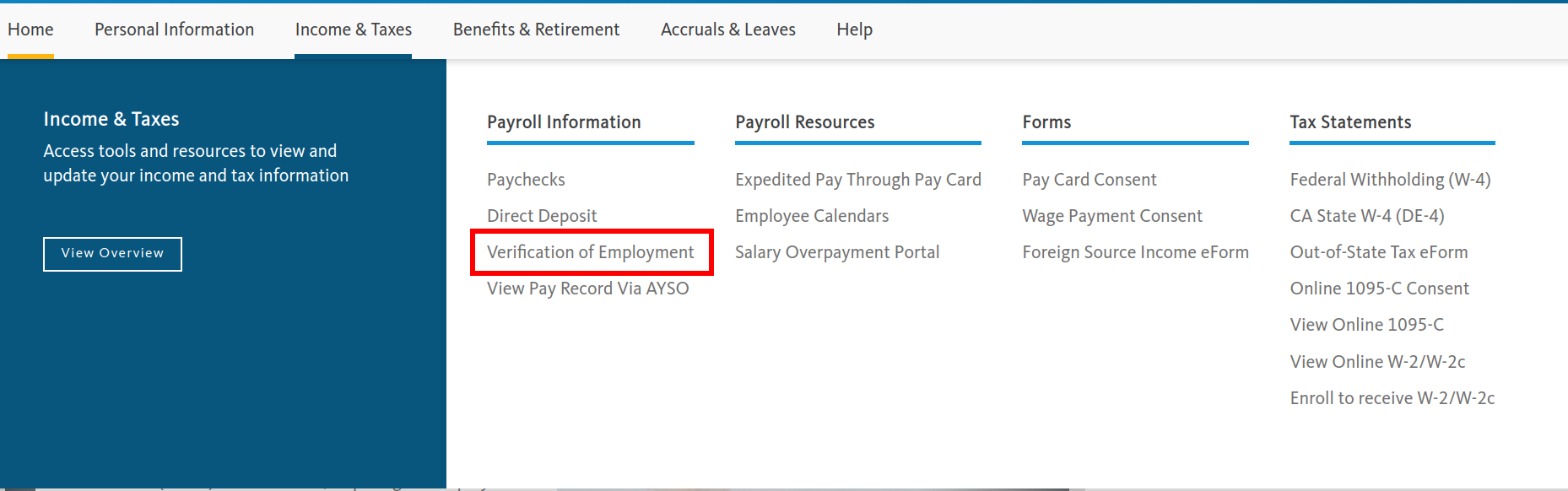
You will also be asked to share your location, which will then allow you to click on Continue and finish this section of the application. You will then be asked for additional information in the form of proof of you being associated with UC Berkeley. The following information will be for teacher roles.

You will need to provide proof of your affiliation. **This proof must have your name, a current date, and the name of your institution on it**. If your application is rejected, it is most likely because your proof of affiliation was missing one of these three things (but, you can always apply again with new proof!). For example, some UC Berkeley student IDs do not have a date on them, so they will not be accepted. You may also have to follow some additional steps to verify your GitHub account, make sure to check your email for instructions. **Note** We have also heard of a bug that results in .png files not being accepted while .jpeg files are.

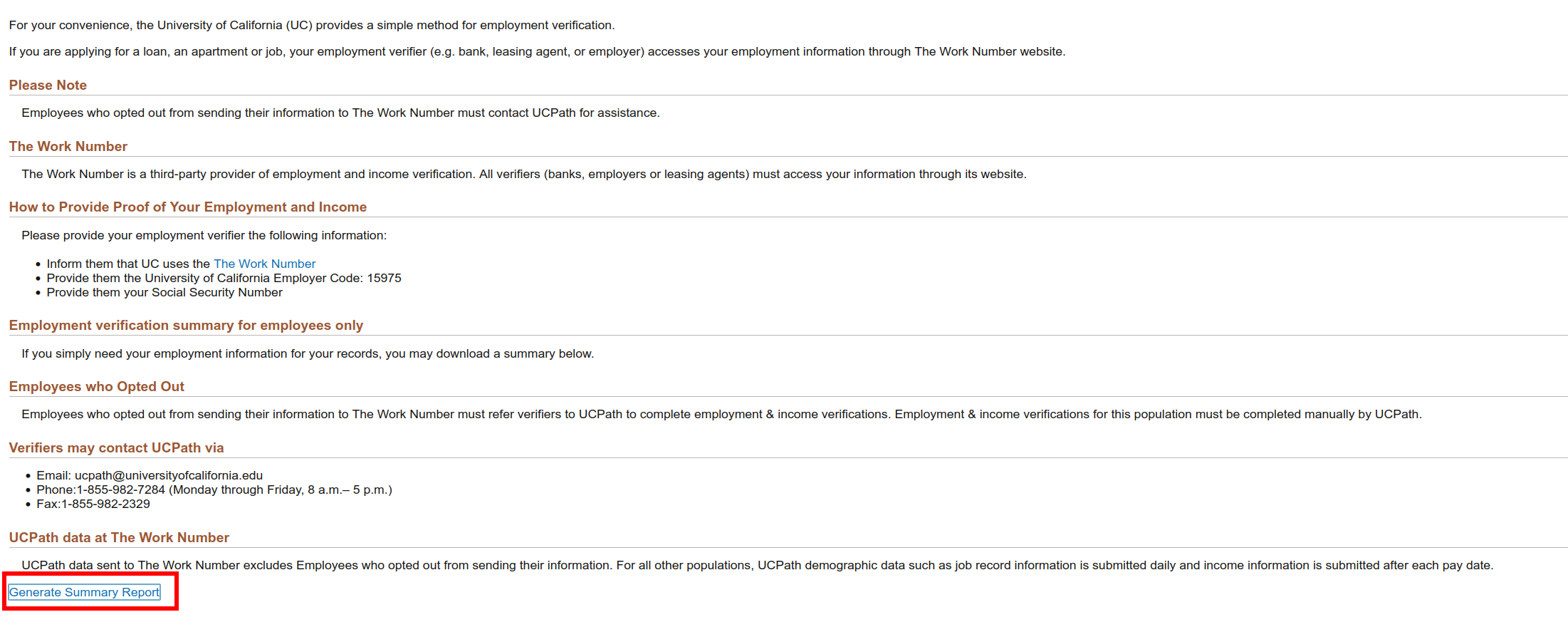
If you are a UC Berkeley student, the most straightforward way to get proof is to download a certificate of enrollment verification by going to CalCentral –> My Academics –> Enrollment Verification (under Academic Records) –> View or Print Enrollment via Self Service –> Obtain an enrollment certificate. This will give you a PDF enrollment certificate which you can screenshot and submit for proof of affiliation (you need to use a screenshot because the application does not accept PDFs).



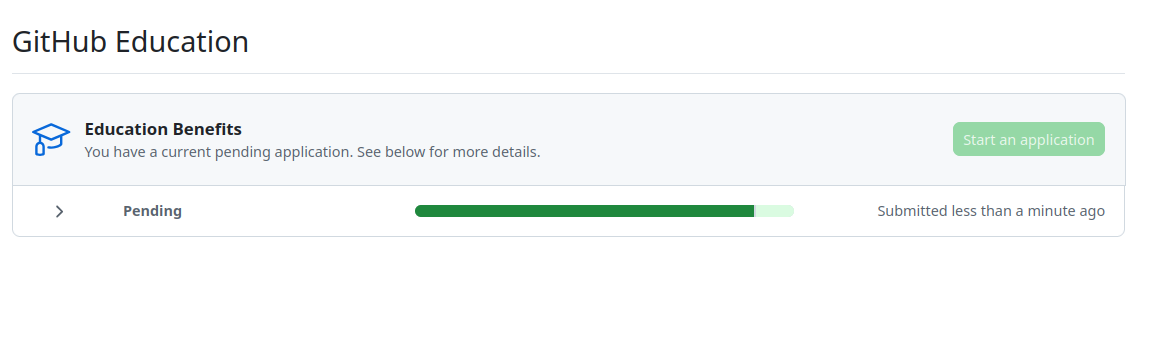
For teacher roles, you will need to go to UCPath.



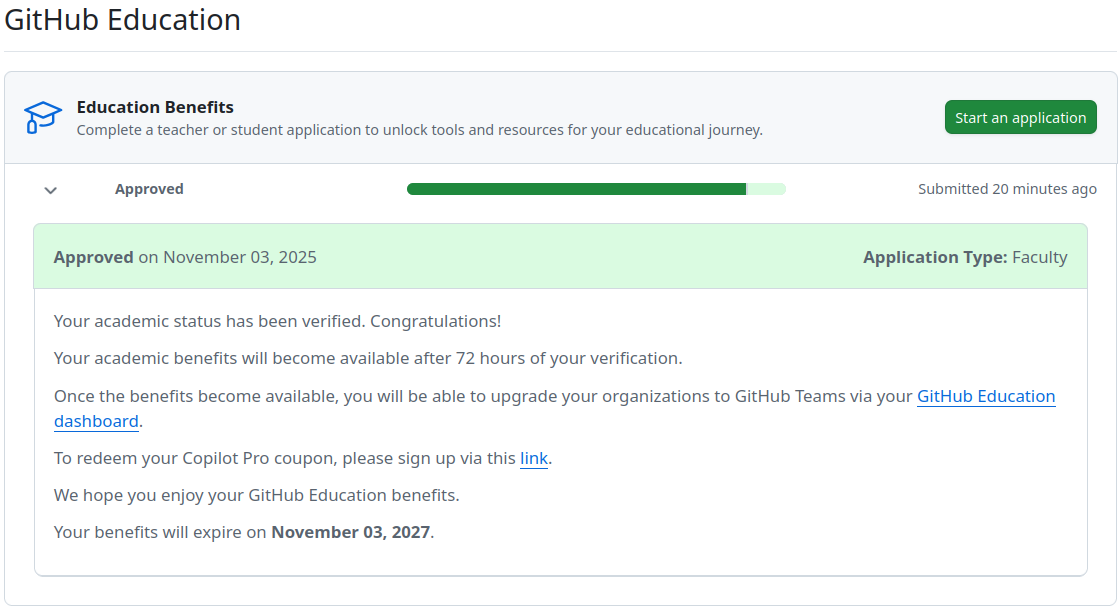
From the main page, you will want to click on Income & Taxes and go to Verification of Employment under Payroll Information.



To get verification of your employment, you will then go to the bottom of the page and click on Generate Summary Report. You will be showed a pdf with your title, the current date of when you generated report (today’s date), and other information. Since you cannot submit PDFs, make sure you get a screenshot of the report and save it as a **.jpg** file.



Once you have submitted your proof, your application will show that it is Pending.



After some time, it will state that you are approved. This is not a full approval, and full approval will take some time to gain access to Copilot Pro.

## 3.3 Section on Getting Copilot Set Up (NEEDS WORK –> WAITING ON ACCESS)

NEED TEXT HERE

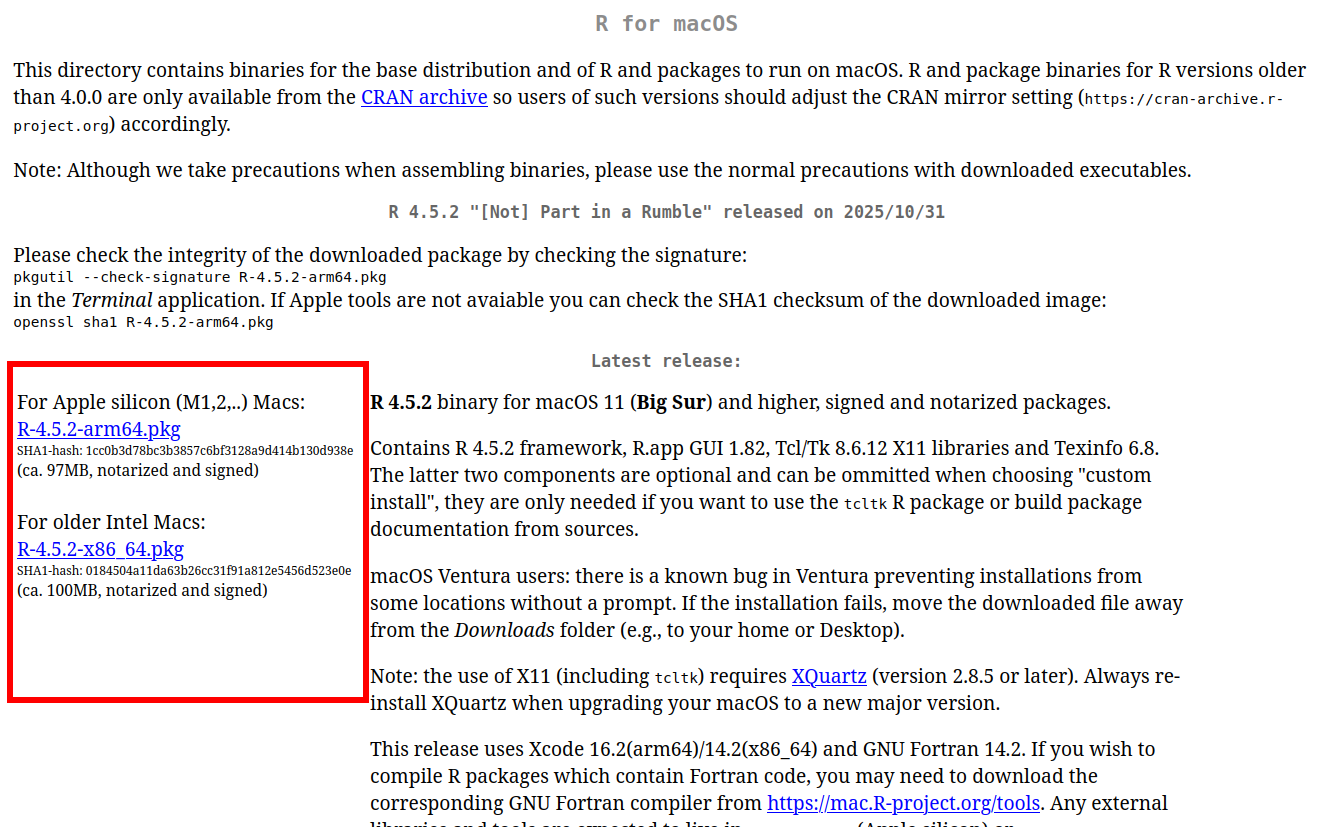
* ☒ Install GitHub
* ☒ Sign up for GitHub Copilot
* [] Install R
* [] Install Visual Studio (VS) Code
* [] Adjust VSCode to work with R
* [] Download Zip file

# 4. Installing R

To install R, you will go to [The Comprhensive R Archive Network (CRAN) website](https://cloud.r-project.org/). There you can download R for your operating system.

## 4.1 Mac

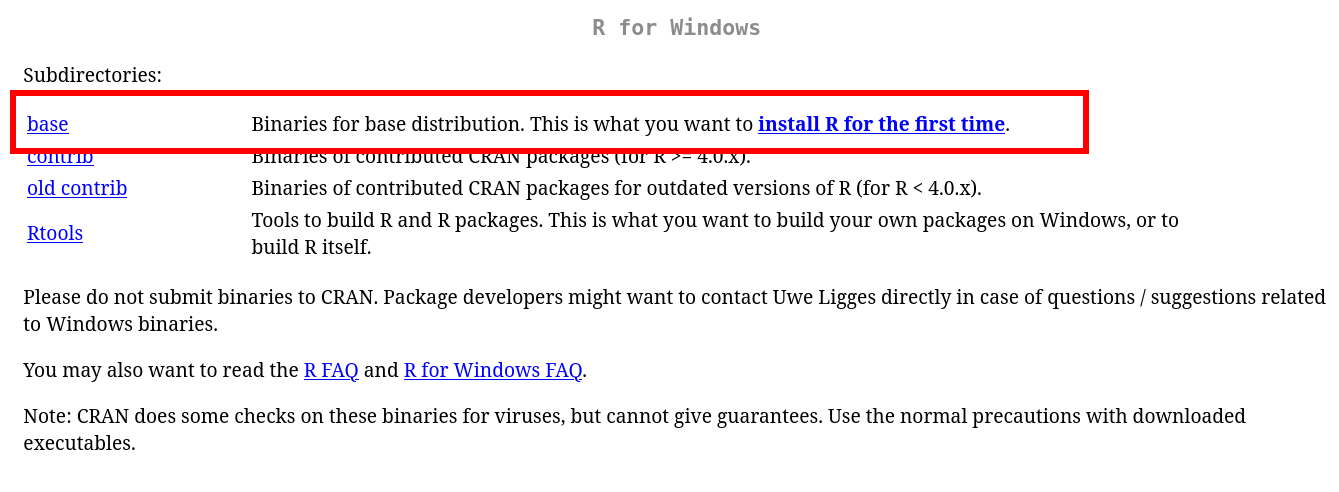
Follow the directions for installing R.



Depending on the type of mac you have, you will choose one of the latest releases below. When going through the installation process, you can keep the defaults for everything. When installed, you should see a message stating The installation was successful.

## 4.2 Windows

Follow the directions for installing R.



When installing for windows, you will use click on the base option and then click Download R-## for Windows to choose the latest R option for Windows. You can keep the defaults for everything. When installed, you should see a message stating The installation was successful.

## 4.3 Linux

There are some really good instructions on how to [install R here](https://cloud.r-project.org/bin/linux/ubuntu/#user) for Linxu distributions. One issue I have faced when installing R from these instructions is that I can install R, but common R packages do not install and lead to a non-zero status error. To prevent that, you will run the following code in your terminal. This will install necessary components that work under the hood for R. If you are having any issues with installing R, think about [submitting a consulting request](https://dlab.berkeley.edu/consulting/submit-consulting-request) where D-Lab consultants can help you get started with R on your Linux machine.

sudo apt update  
sudo apt install -y build-essential libcurl4-openssl-dev libssl-dev libxml2-dev  
  
# optionally  
# sudo apt install -y libfontconfig1-dev libharfbuzz-dev libfribidi-dev libfreetype6-dev libpng-dev libtiff5-dev libjpeg-dev

* ☒ Install GitHub
* ☒ Sign up for GitHub Copilot
* ☒ Install R
* [] Install Visual Studio (VS) Code
* [] Adjust VSCode to work with R
* [] Download Zip file

# 5. Install VSCode

Let’s move forward with installing VSCode. You can install [VSCode here](https://code.visualstudio.com/download) for your operating system. Below are detailed instructions on how to install VSCode.

### 5.0.1 Mac

[Link for Mac install](https://code.visualstudio.com/docs/setup/mac)

Follow the directions to install VSCode. Below, I will include some helpful tips for VSCode extensions that may help when using R.

### 5.0.2 Windows

[Link for Windows install](https://code.visualstudio.com/docs/setup/windows)

Follow the directions to install VSCode. Below, I will include some helpful tips for VSCode extensions that may help when using R. **For Windows users, it is recommended to check “Save version number in registry” during installation so that the R extension can find your R installation automatically. If you have not done this you may need to add the location of your R to your PATH manually (see FAQ 3.1 I am using windows and my VS Code can’t find R!).**

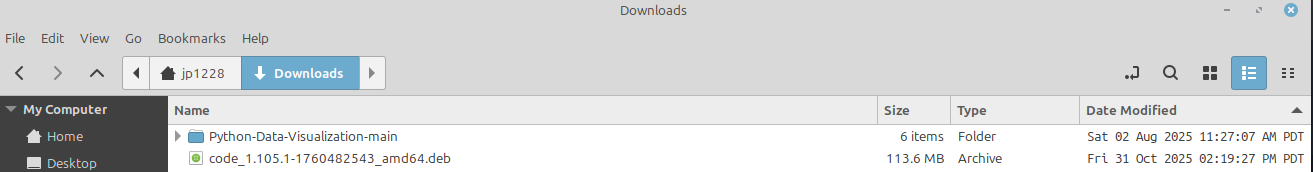
### 5.0.3 Linux

**Note**: Everything below shows the installation using Linux Mint.

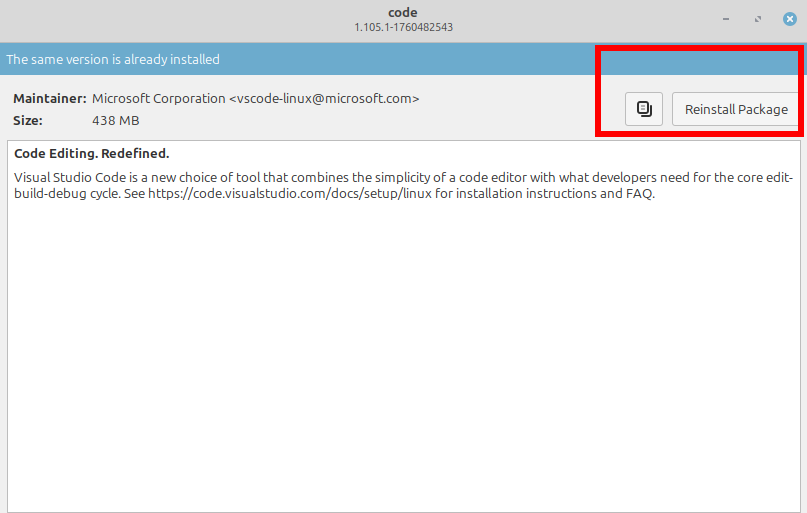
System:  
 Host: jp1228-Swift-SF314-52 Kernel: 6.8.0-87-generic arch: x86\_64 bits: 64  
 Desktop: Cinnamon v: 6.4.8 Distro: Linux Mint 22.1 Xia

[Link for Linux](https://code.visualstudio.com/docs/setup/linux)

**Option 1: Use the Link**



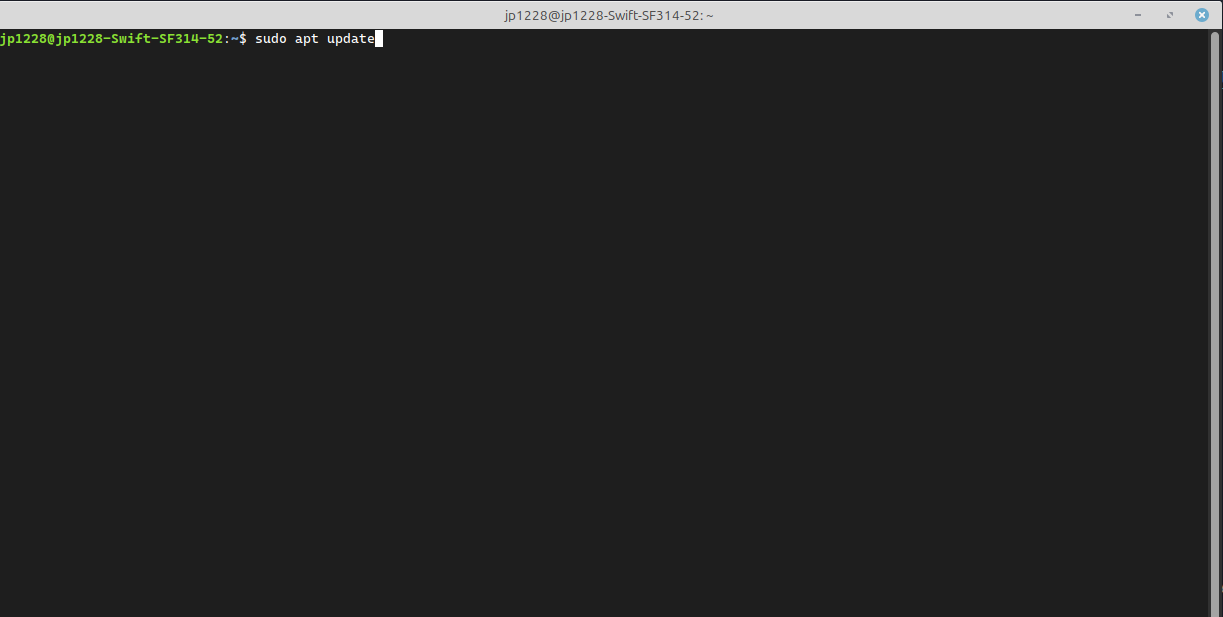
1. Click on the link above and open the download file.



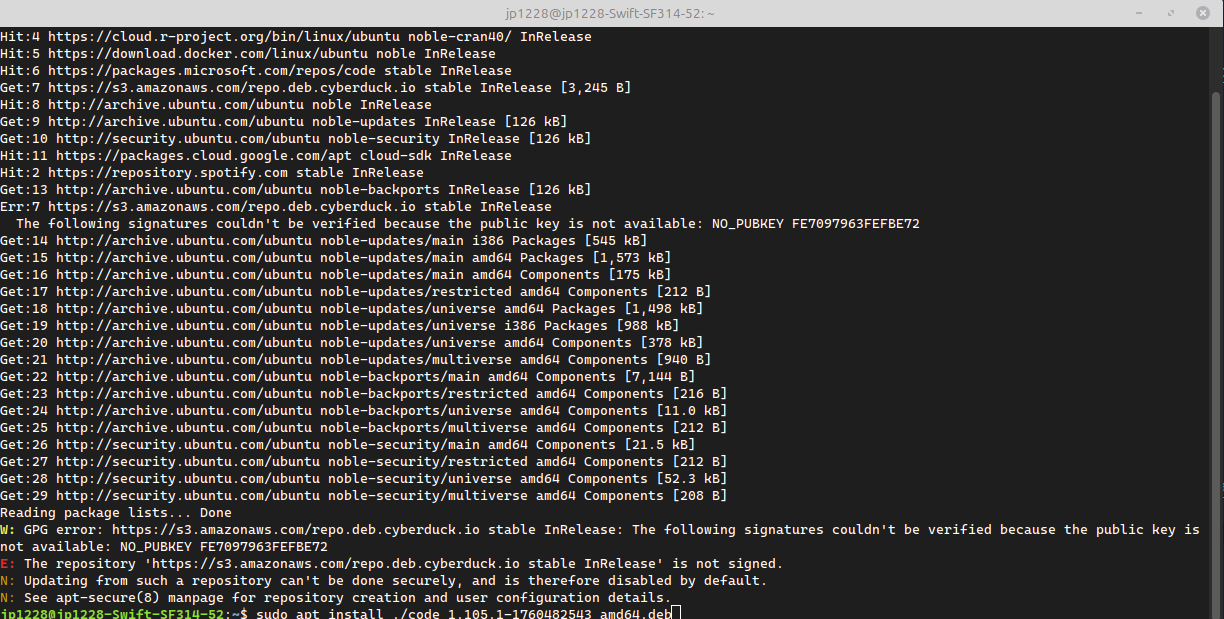
1. Click Install Package to start the install.

You should be set to use VSCode.

**Option 2: Download Using Terminal**



1. Update your programs.



1. Then you will install the file that you downloaded from [this page](https://code.visualstudio.com/download) as shown below. Your file will look different, depending on the version and differences in your linux distribution, but it should start downloading after running the code below.

sudo apt install ./<file\_name>.deb  
# include the name of your file and change <file\_name> to the name of your file

You should now be set up to use VSCode. If you are having difficulties, it may be easier to use option #1 above.

## 5.1 VSCode Documentation

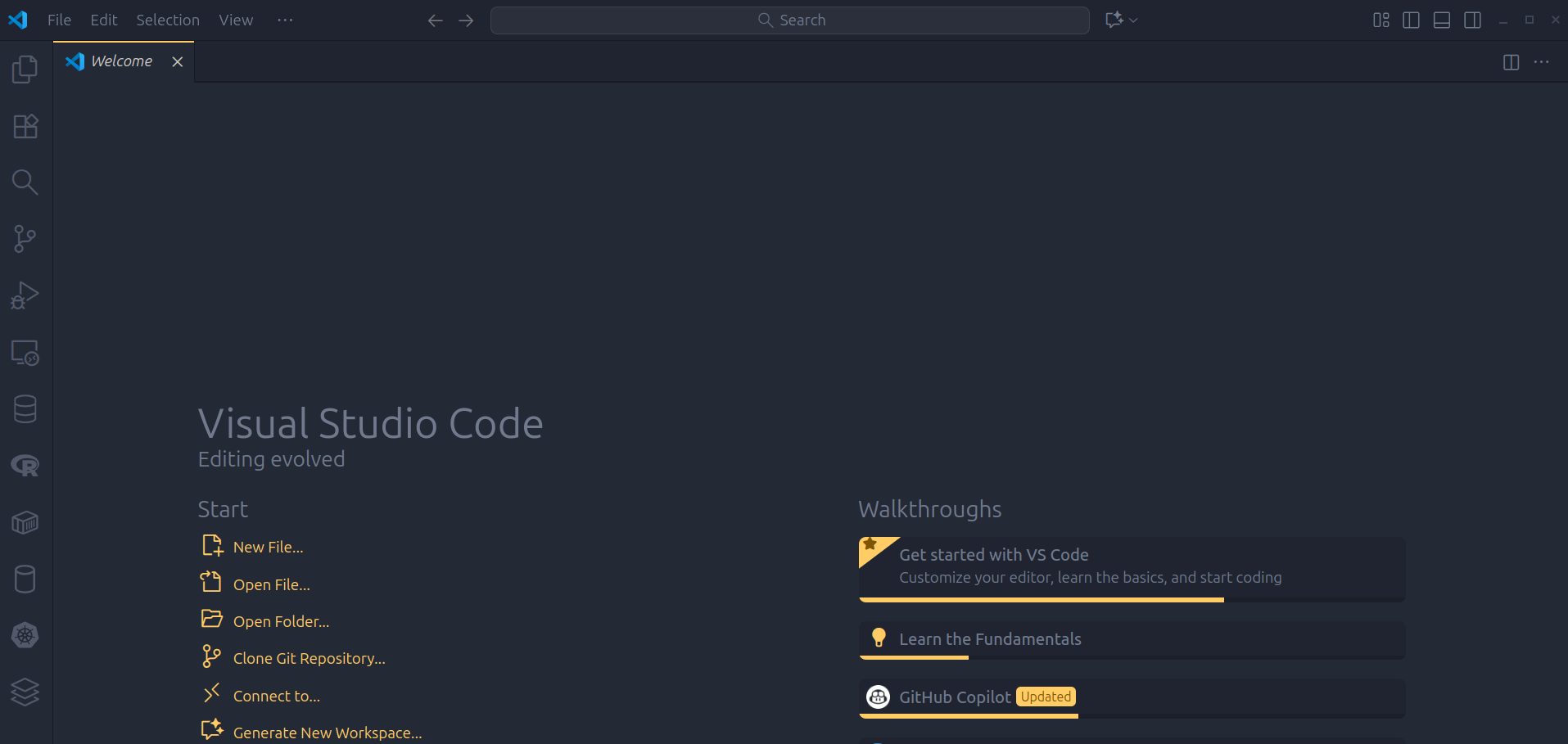
When you installed VSCode, it should have brought you to the documentation page. If not, you can find all the [documentation here](https://code.visualstudio.com/docs). [This tutorial](https://code.visualstudio.com/docs/getstarted/getting-started) also provides an in-depth tutorial on getting started with VSCode.

* ☒ Install GitHub
* ☒ Sign up for GitHub Copilot
* ☒ Install R
* ☒ Install Visual Studio (VS) Code
* [] Adjust VSCode to work with R
* [] Download Zip file

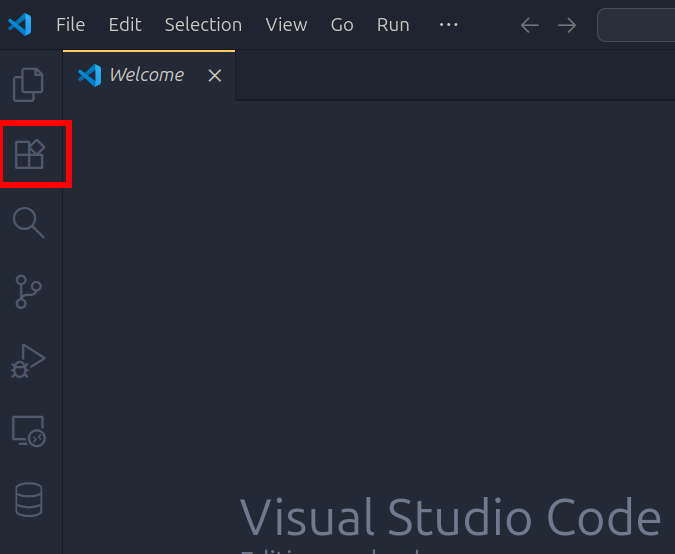
# 6. Setting up R in VSCode

**Note** Your VSCode will look slightly different, as the screenshots are from a custom VSCode theme. Additionally, the ordering and number of tabs on the sidebar may be different from the VSCode default.

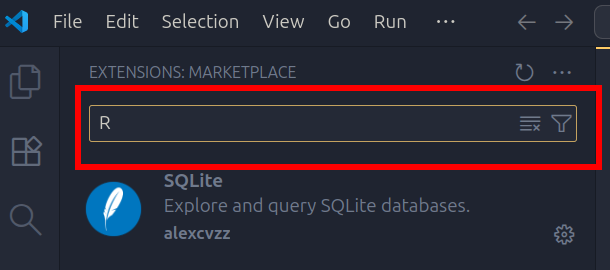
Depending on the version of R you installed, there may be some issues in incorporating some of the packages. I will include alternatives to try and make sure everything works. If you have not downloaded R, then you will want to do that first. Once you have R, then you can move forward with installing extensions for R.



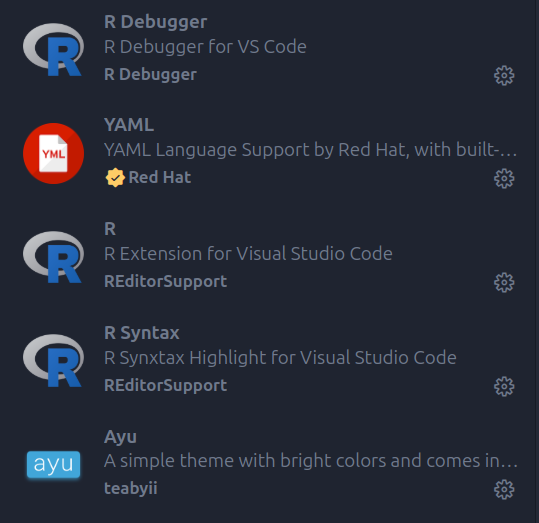
Once you open VScode, you will be on the Welcome page.



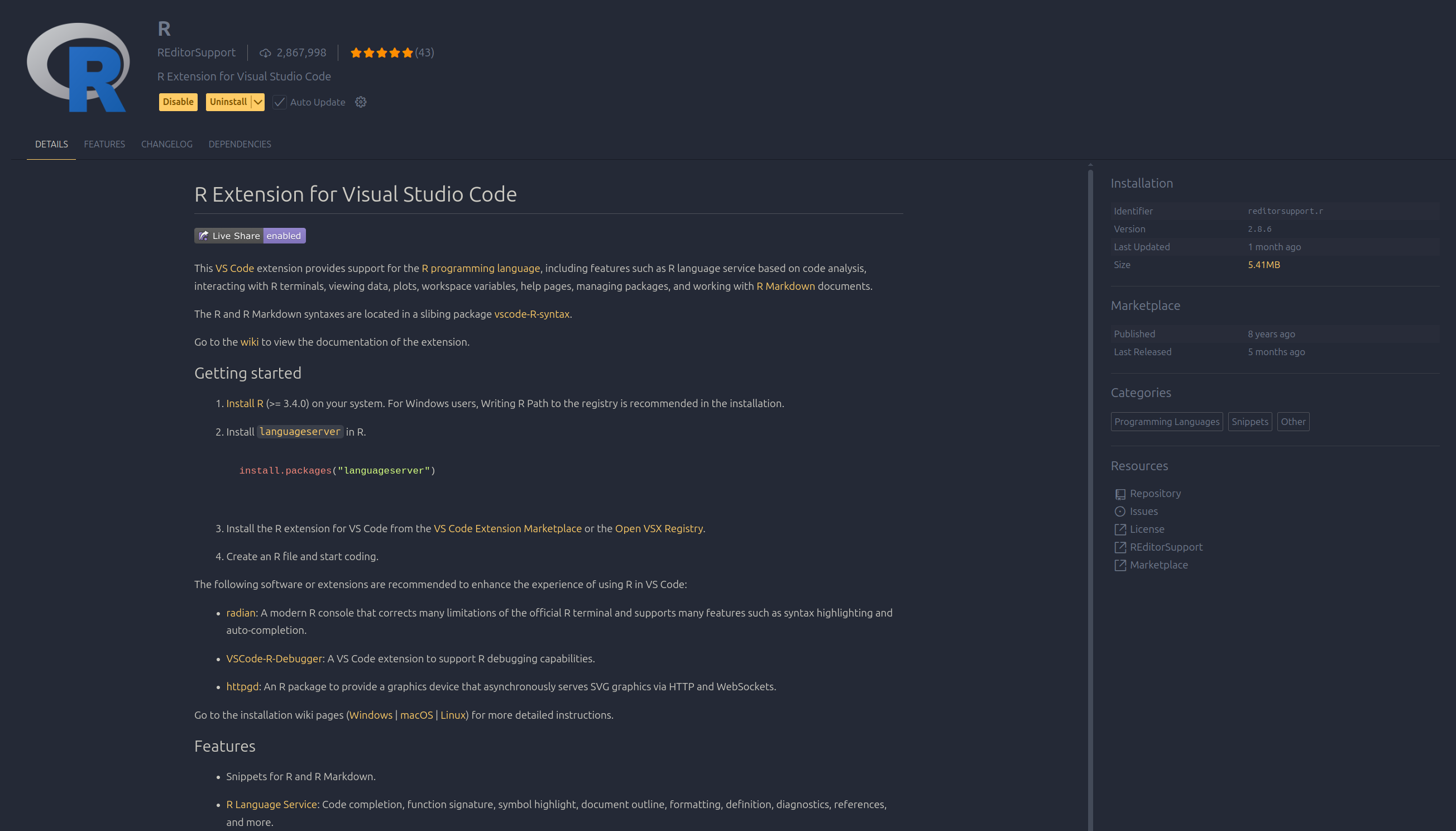
From there, the first place you will want to go is to the left sidebar. There you will see several tabs. You will click on the Extensions tab shown above.



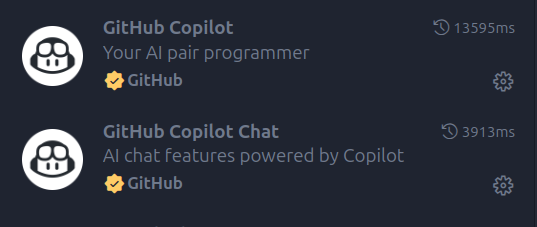
The easiest way of finding all the extensions you will need is to use the search bar at the top. \*You can also use this to find a theme for your VScode. [See several examples of themes here](https://vscodethemes.com/) that you can then search for by name.



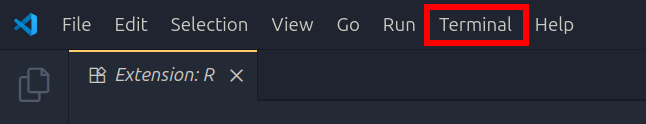
First, you will want to install the R extension. In addition to installing the R extension, you may also want to install the **R Debugger** and the **R Syntax** extensions. These add some additional tools when using R.



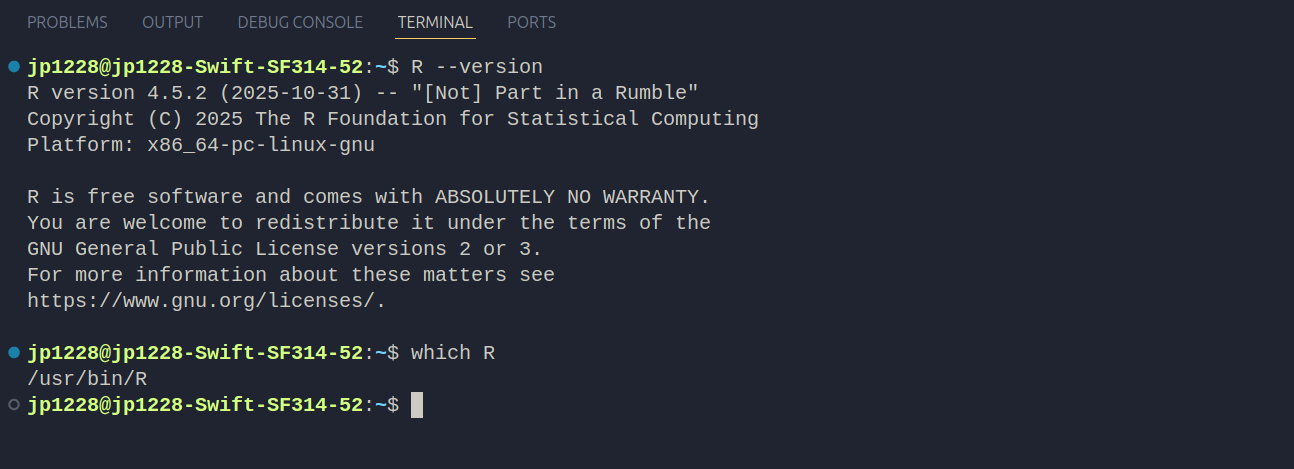
Once you click on the extension on the sidebar, a tab will open up on the main window section of VSCode, there you can click Install and read the documentation behind each extension. For the R extension, there are summarized instructions below.



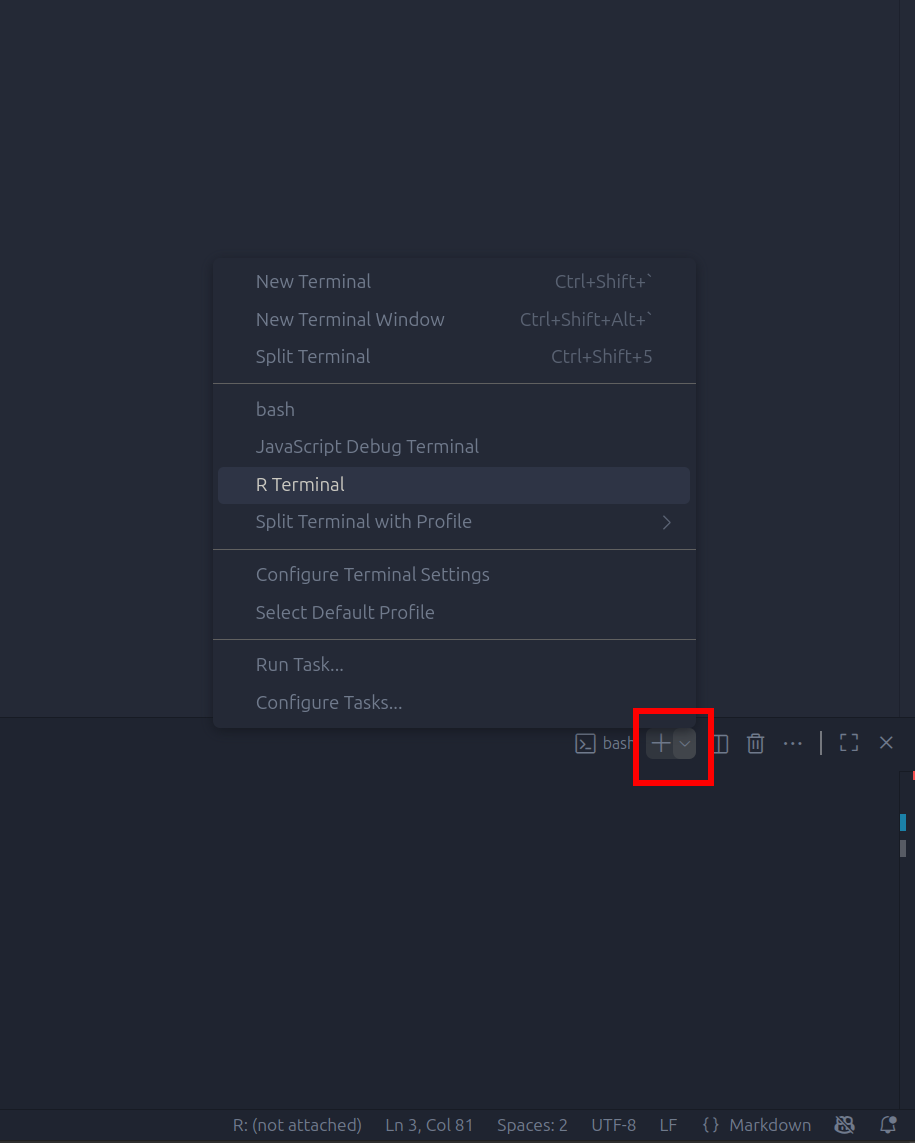
Right now would be a great time to search and install GitHub Copilot. There are two extensions that should be installed, GitHub Copilot and the GitHub Copilot Chat extension. GitHub Copilot Chat should be installed once you install GitHub Copilot.



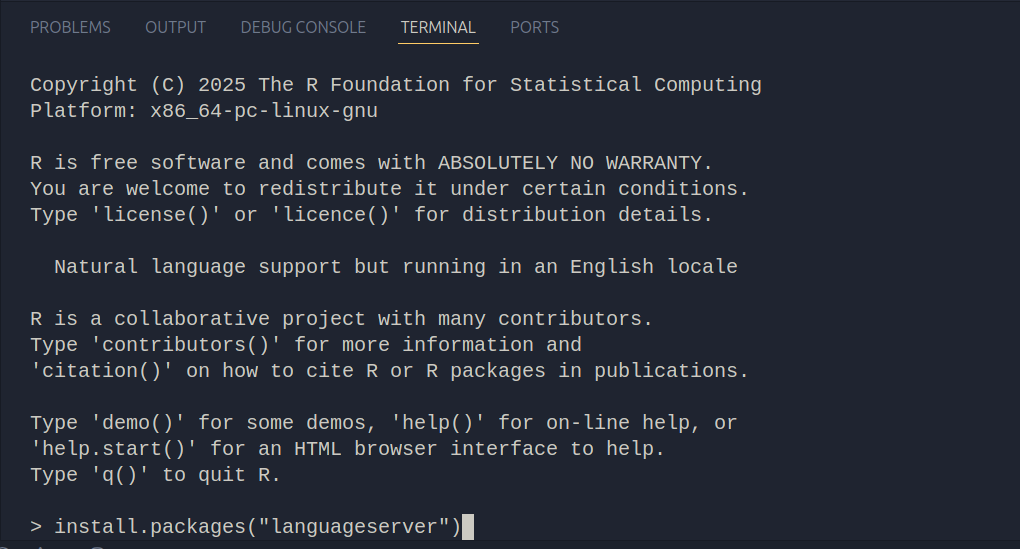
The easiest way to install the necessary packages is to go to the top left corner on VSCode and click on Terminal. There you will then click on New Terminal. A new terminal will open at the bottom of your VSCode. It should be listed as a bash terminal.



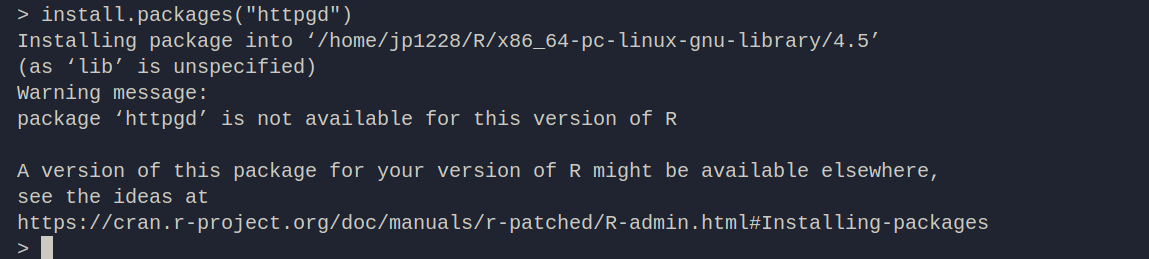
If you want to double check the status of your R installation, you can type R --version into your bash terminal to see if R is installed correctly. You can also see where R is located by typing which R. This will show you the location of R that VSCode is using.



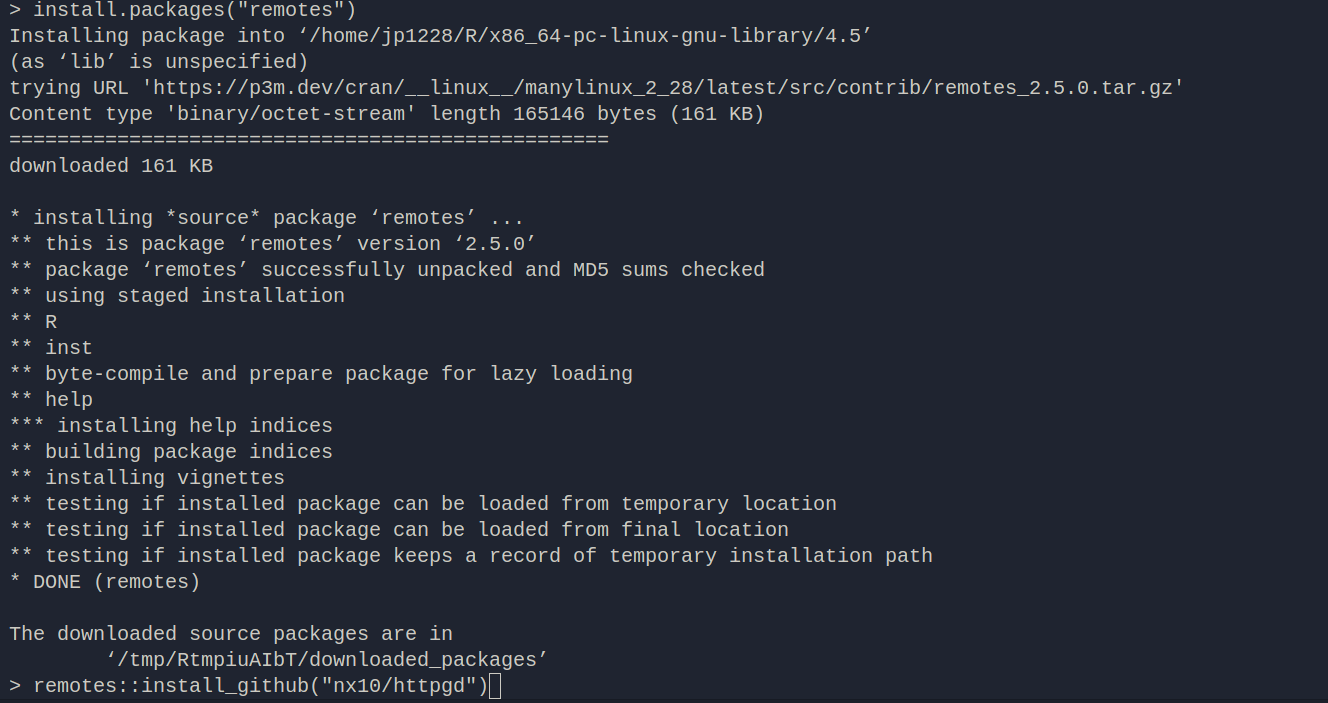
The easiest way to install the languageserver and httpgd packages is by going to the right side of your terminal and clicking on the dropdown menu and choosing a R Terminal. As mentioned in the documentation for the R extension, you can install the radian package and make the appropriate changes in the settings.



As this is now a R terminal, you can install the languageserver package by using install.packages("languageserver").



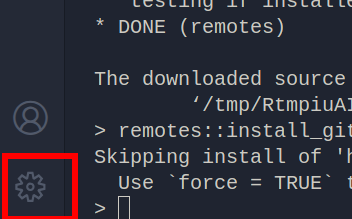
When installing the httpgd package, I have found some issues when trying to install it from newer versions of R. If you experience these issues, you should install the development version of the package.



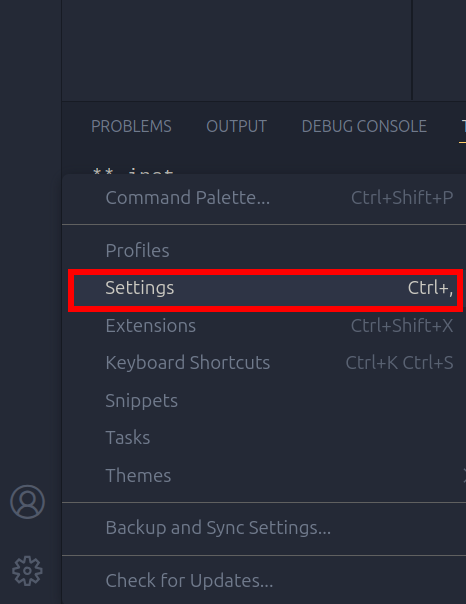
You can get the httpgd package by first installing the remotes package (using the typical method of install.packages("remotes")). This will then allow you to install the development version from GitHub. If you want to follow along with the instructions from the developer(s) of the httpgd package, you can [follow the installation instructions here](https://github.com/nx10/httpgd).

Lastly, I will show some extra customizable settings to make VSCode similar to using RStudio. Below are some settings that you can change in your settings.json file. You can also change the settings by going to the gear (see directions below) to make changes to your VSCode.

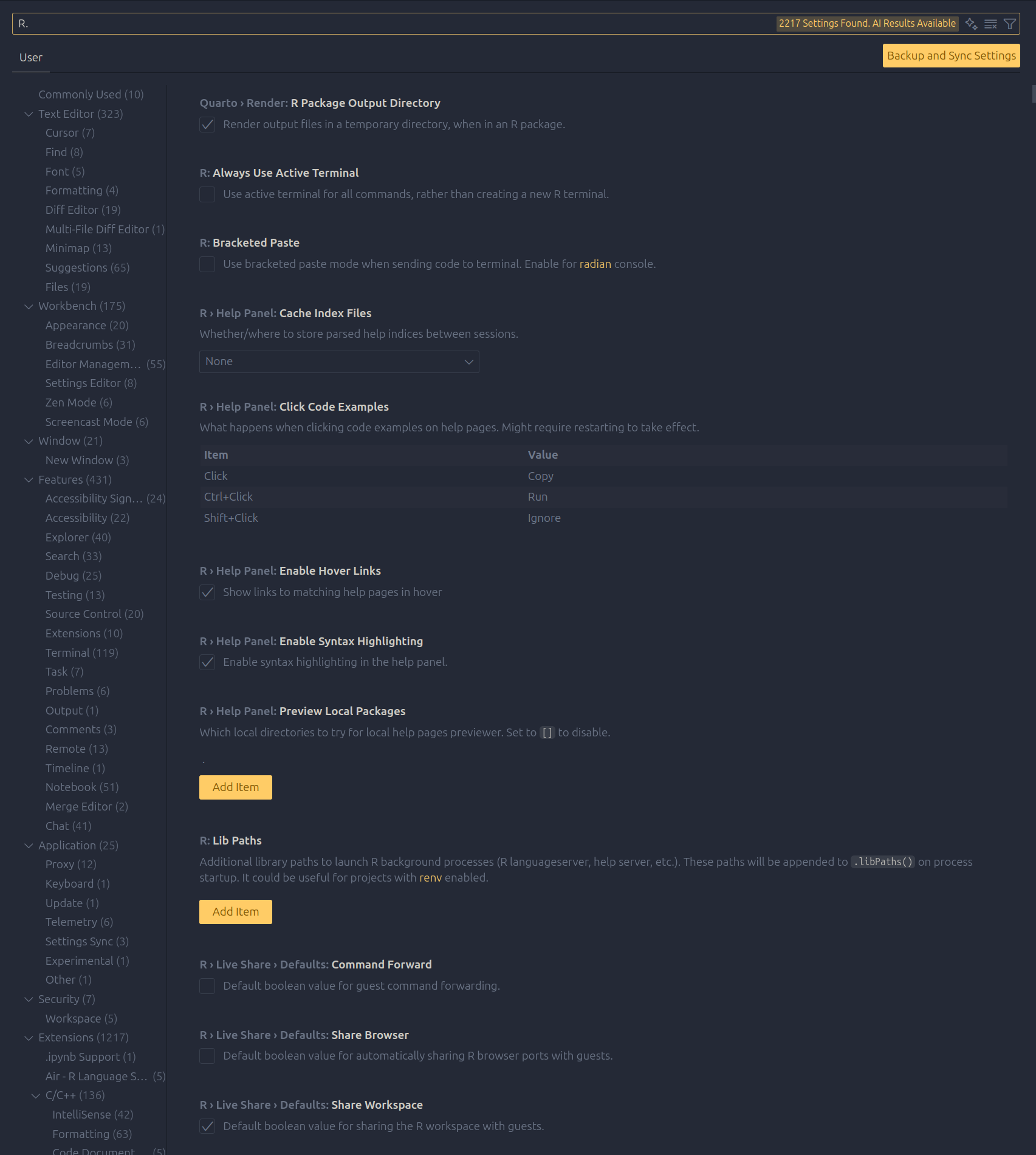
1. Making Changes by Settings Tab



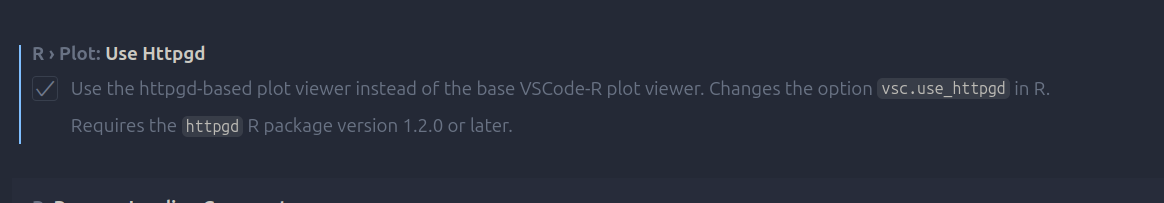
At the bottom of the sidebar you will see a gear. Clicking on the gear will open options for customizing your VSCode.



Clicking on settings will show you everything that you can alter in VSCode.



To make changes to R in VSCode, you will type out R. in the search bar.

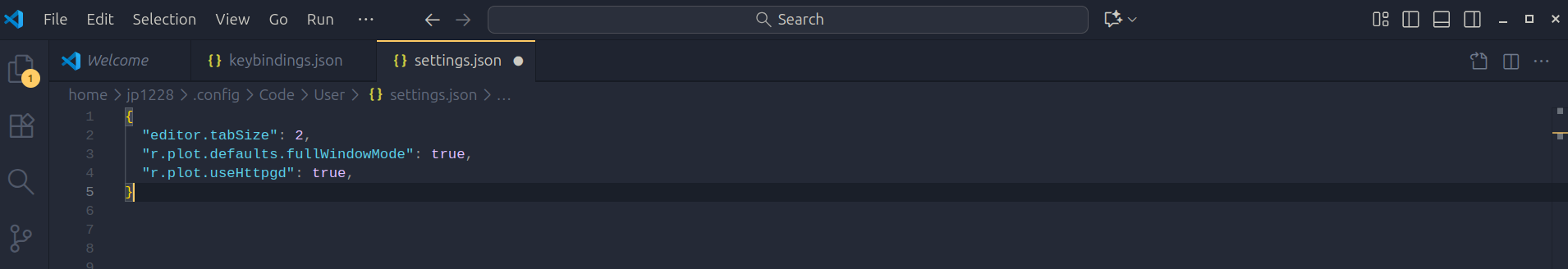


For your purposes, you can scroll through the R options until you find the Use Httpgd option and click on the check box.

1. Making Changes Using the settings.json File

The second option is to use the settings.json file. To get to the settings\_json file, you will use the following keybinding shortcut (Windows/Linux: Ctrl + Shift + P, Mac: Cmd + Shift + P). This will open the command palette where you can then search for Preferences: Open User Settings (JSON). Here you can then copy and paste the code below to use the httpgd package and tab 2 spaces rather than the default 4. If you change your theme, this will also show up here as well as preferences made for other extensions. Once you have made these changes, you can save your settings and close out of the settings.json file.

{  
 "editor.tabSize": 2,  
 "r.plot.defaults.fullWindowMode": true,  
 "r.plot.useHttpgd": true,  
}

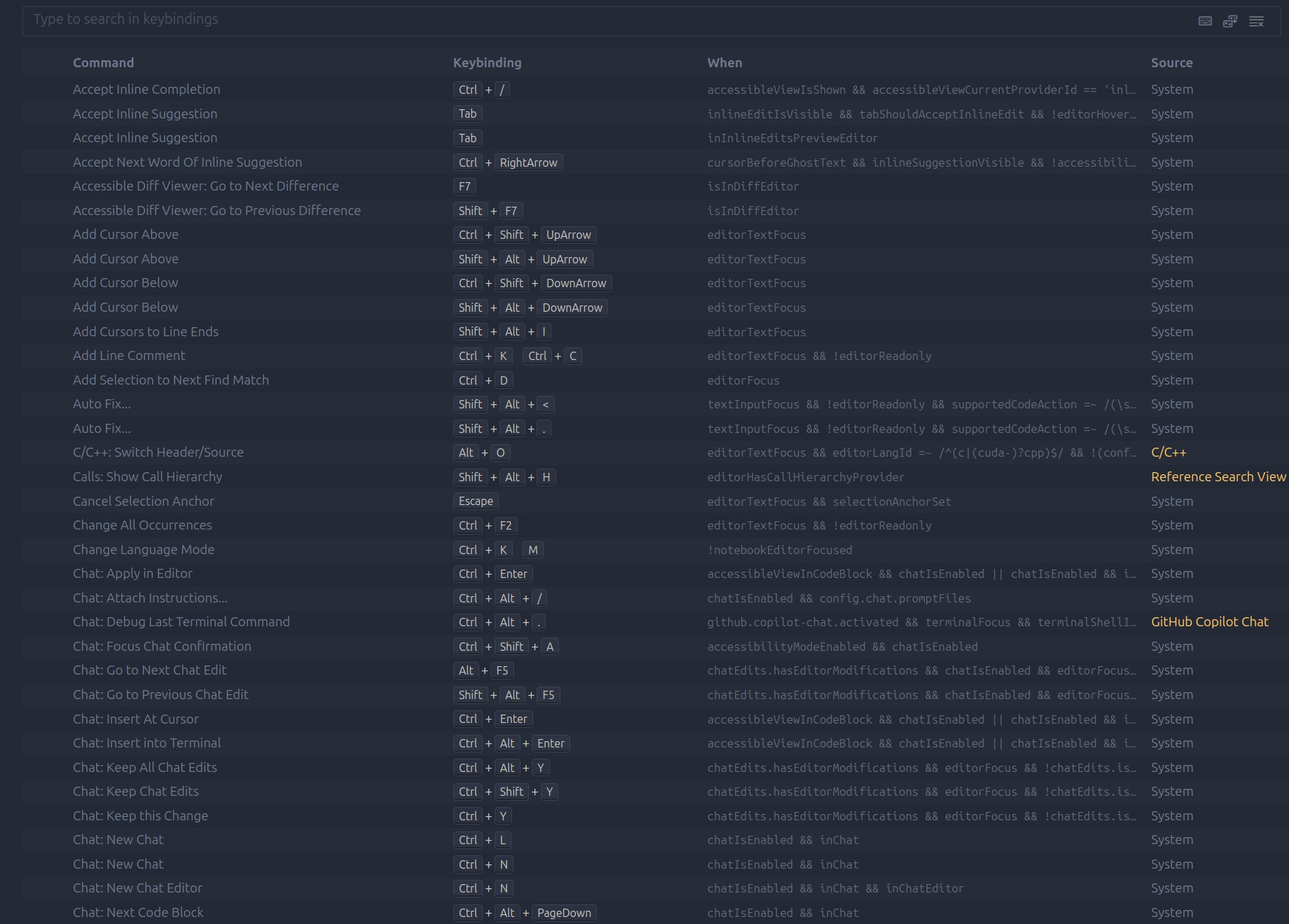


Another preference is to include shortcuts for some common RStudio shortcuts. The main two are being able to comment in/out code in your R scripts. Similar to RStudio, you can add the shortcuts below using the keybindings.json file.

You will use the keybinding shortcut (Windows/Linux: Ctrl + Shift + P, Mac: Cmd + Shift + P) to get to the command palette again. Here you can search for the Preferences: Open Keyboard Shortcuts (JSON) and copy and paste the code below. **Note** Be aware that if you do not have R version 4.1.0 or greater, you will need to change the { "text": "|>" } to { "text": "%>%" }. You can then save the file and close out and you should be able to use these shortcuts.

[  
 {  
 "key": "ctrl+shift+c",  
 "command": "editor.action.commentLine",  
 "when": "editorTextFocus && !editorReadonly"  
 },  
 {  
 "key": "ctrl+shift+m",  
 "command": "type",  
 "args": { "text": "|>" },  
 "when": "editorTextFocus && editorLangId == 'r'"  
 }  
]

To make changes to any other shortcuts, you can use the command palette to search Keyboard Shortcuts to change any other shortcuts. **Note** Be aware that you could possibly overwrite other important VSCode functions so be cautious when making changes. You can type in the shortcut you want to create in the search bar at the top to see what current functions use that shortcut.

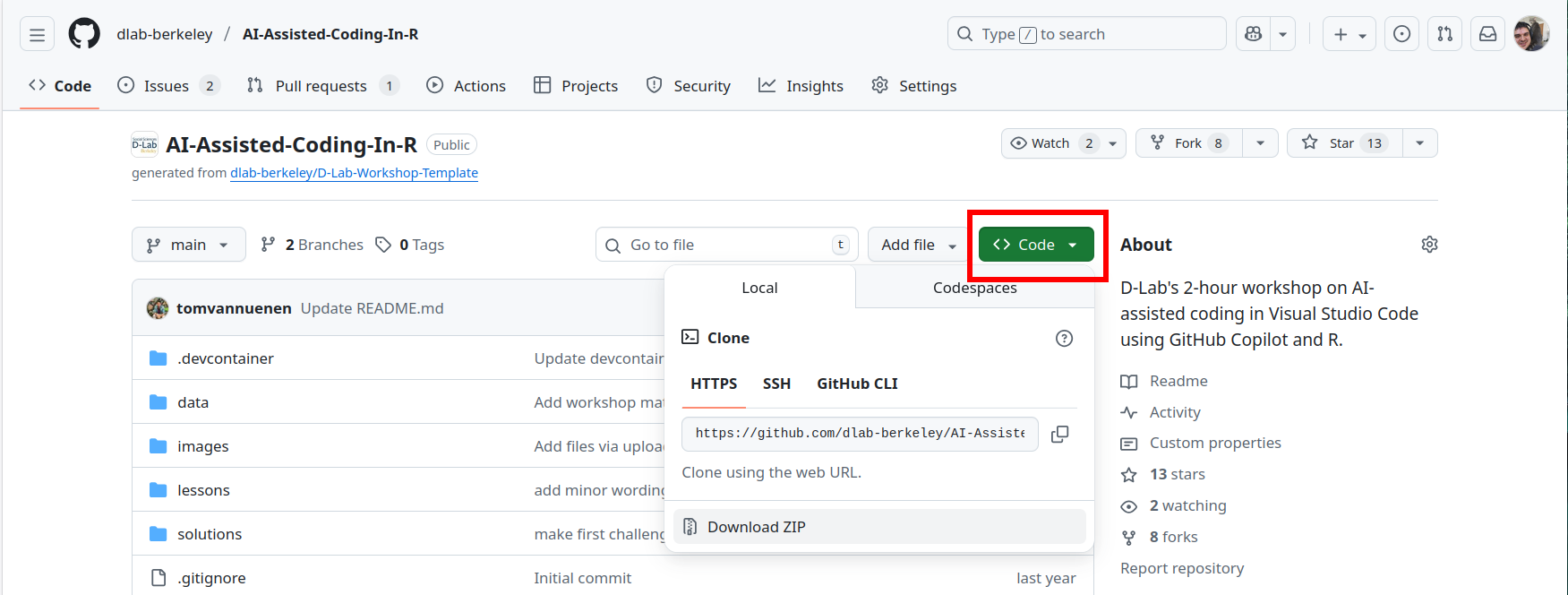


* ☒ Install GitHub
* ☒ Sign up for GitHub Copilot
* ☒ Install R
* ☒ Install Visual Studio (VS) Code
* ☒ Adjust VSCode to work with R
* [] Download Zip file

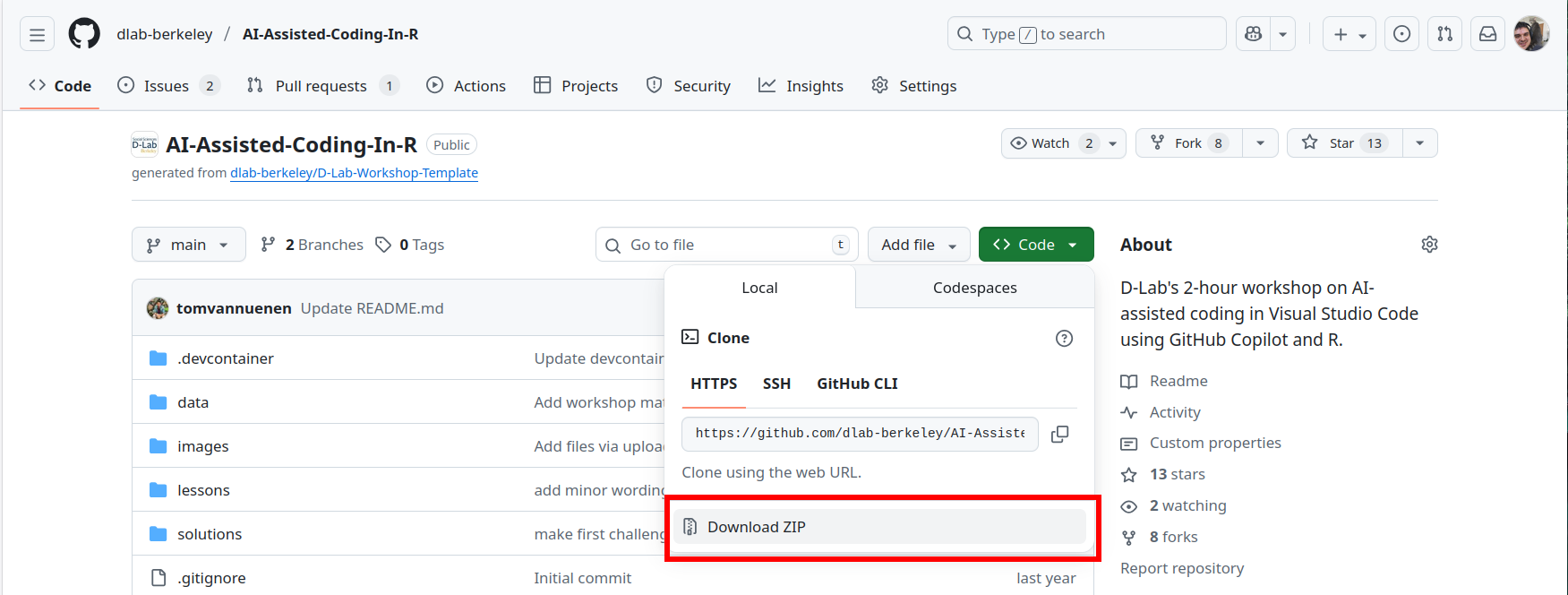
# 7. Download Zip File

To get all the materials for the AI-Assisted-Coding-In-R workshop, you will want to download the [Zip file here](https://github.com/dlab-berkeley/AI-Assisted-Coding-In-R/tree/main).

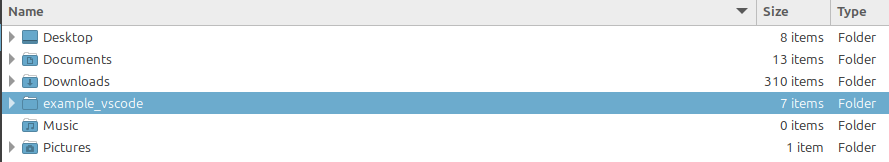
**Note** You may want to wait until the day of your workshop to make sure you download the most recent version of the contents.



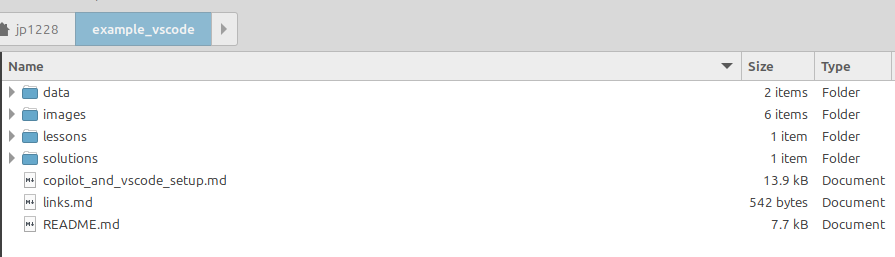
Once you are at the D-Lab GitHub repository for the AI-Assisted-Coding-In-R workshop, you will go to the green button that says <Code> with a dropdown menu. Clicking on the button will give you options for how to put the repository’s contents on your local computer.



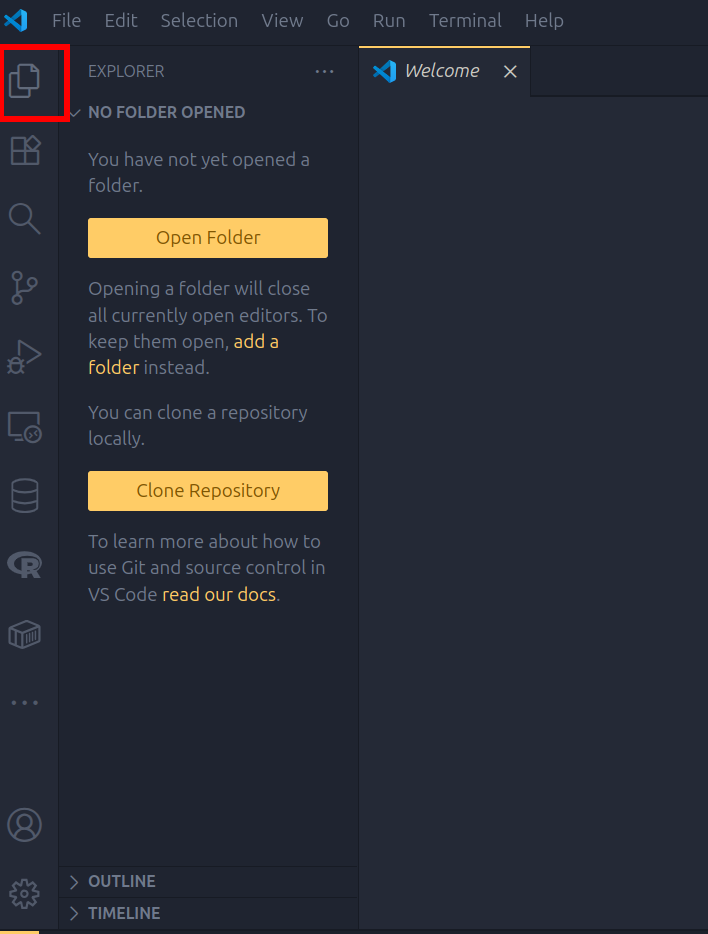
For now, you can Click on the Download ZIP to download a ZIP file. The ZIP file should then be in your Downloads folder. There you can extract the contents of the ZIP file.



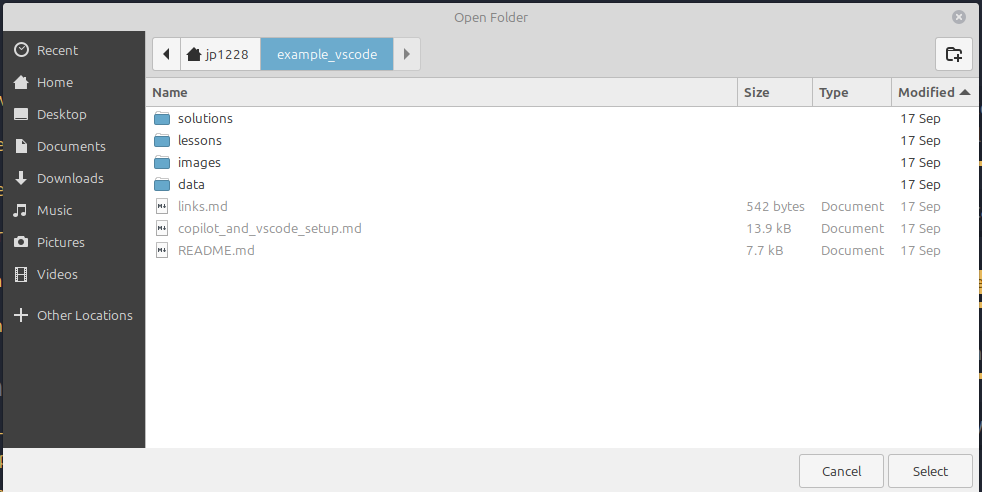
Once you have extracted the contents of your ZIP file. I put all of the contents in a new folder. You can put this folder wherever it makes sense to you. I have my folder (which I named example\_vscode) under my username folder. You can create the folder on your Desktop for easy access or within your Documents folder if you would like.



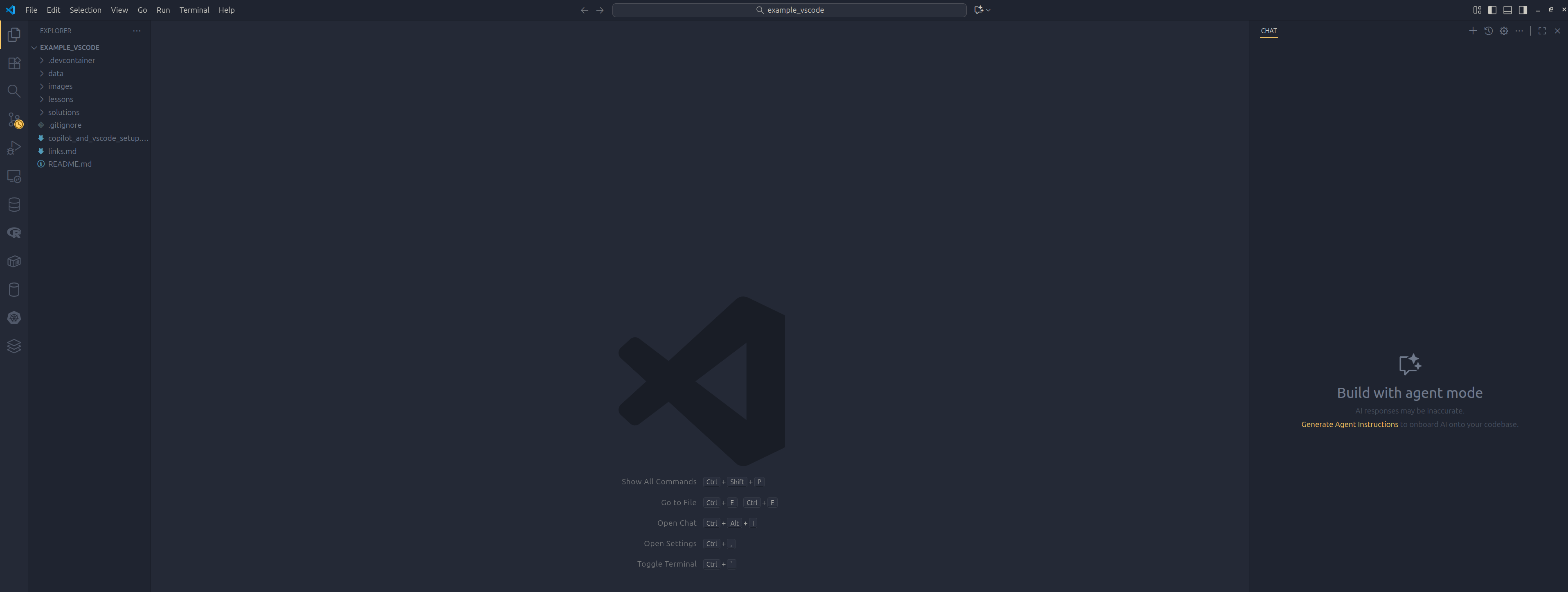
Within the folder, I copied and pasted everything from the ZIP file into this folder. The files should look similar to the screenshot above.



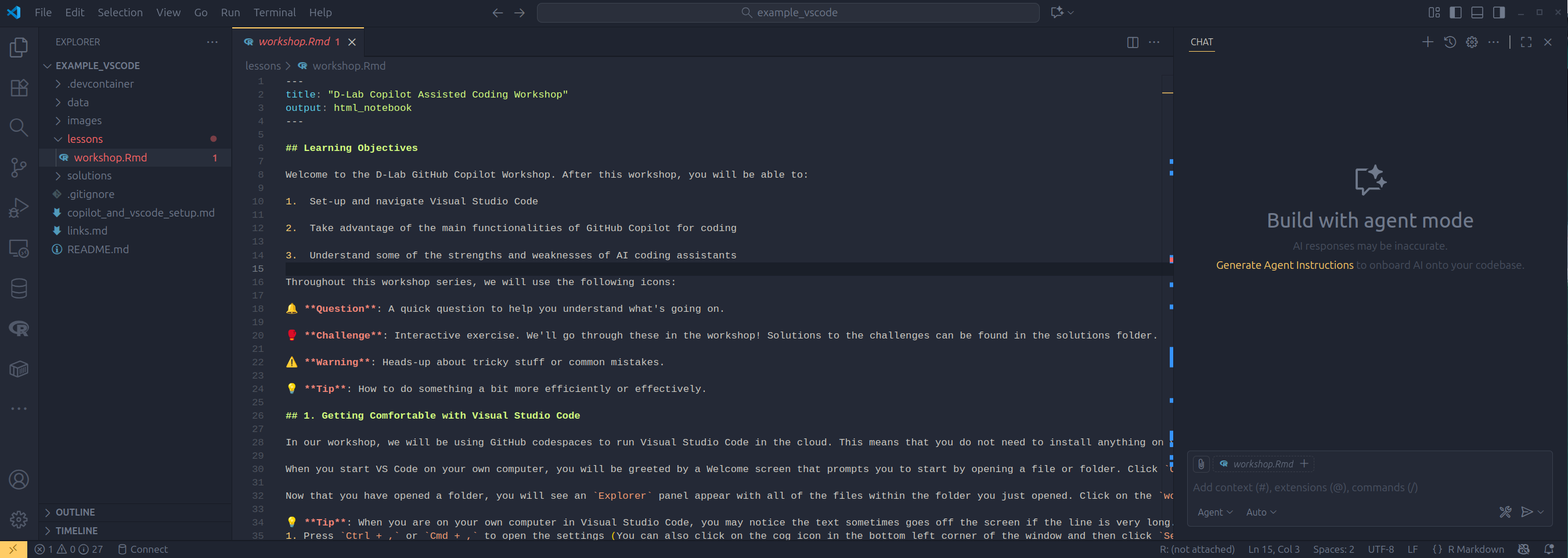
Now you can open VSCode and click on the folder tab on the left sidebar. There you can click on the option Open Folder to open the folder with the workshop contents.



In my case, I will look for my example\_vscode folder and click select at the bottom to start VSCode from this folder.



Once you select your folder, your VSCode will populate with your workshop files on the left and a tab on the right for your prompts with GitHub Copilot.



Finally, you can click on the lessons folder on the left and click on your notebook.

Congrats! You are now ready for your AI-Assisted-Coding-In-R workshop.

* Install GitHub
* Sign up for GitHub Copilot
* Install R
* Install Visual Studio (VS) Code
* Adjust VSCode to work with R
* Download Zip file