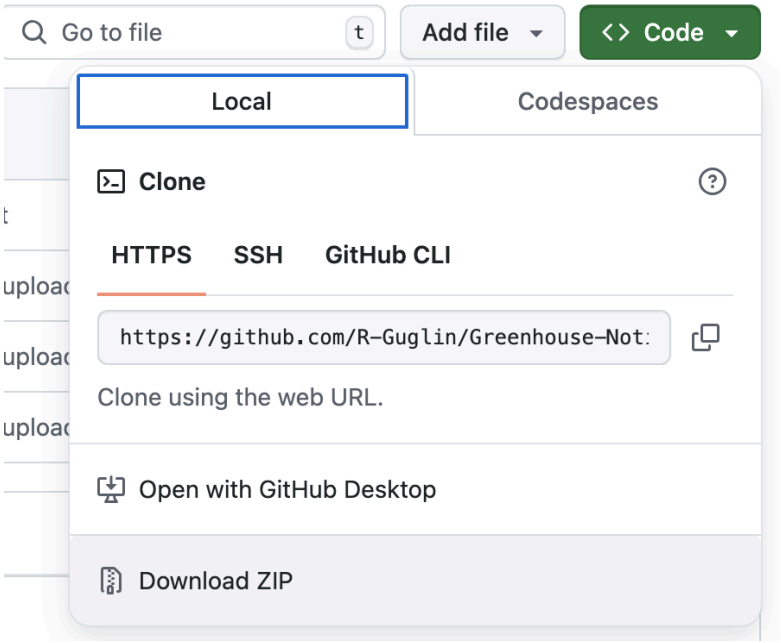
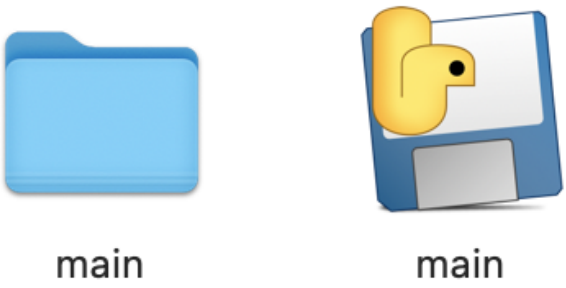
Installation

1. Go to https://github.com/R-Guglin/Greenhouse-Notion.
2. Click the green “Code” button, then select “Download ZIP.”



3. You should now have a ZIP file in your designated downloads folder. Double click this file to expand its contents.
4. You should see three files: a README, a textfile called cohort.txt, and another ZIP called dist.zip — expand dist.zip into its contents as well.
5. Finally, you should end up with a folder called **dist** which contains two items: a folder called **main**, and an *executable* called **main**. The executable is our integration app.



6. If you want, you can drag the executable into your Applications folder, desktop, or wherever else you choose. It doesn't need to be in the same location as the Main folder. But make sure not to delete that folder, since it contains all the libraries and other dependencies that our code runs on.
7. For extra convenience, try making a folder called that contains all of the things you'll need to run the program:
    a. The executable
    b. The .txt file
    c. Any .csv files downloaded from Google forms which will be fed into the program.

Getting the CSV

1. Go to forms.google.com and edit the form you are hoping to pull information from.
2. In the "responses" tab, navigate to the form's linked Google Sheet.
3. Make a copy of this Sheet, and delete any rows corresponding to people you don't want to add to the directory. You do not need to delete any columns. You also don't need to delete duplicates, as the program will check the directory for existing people before adding their information again.
4. **Important: Updating the .txt** (if needed)
    a. Every time the Google Form's structure (i.e., the questions and their order) is modified, you will need to check **cohort.txt** and make sure everything is up to date. You can edit this file using TextEdit or Notepad or any other text editor.
    b. The way my program works is to scan through a CSV for columns with certain titles, like "Location" or "Name." Google Forms automatically names Sheet columns after the corresponding question. Therefore, some of the columns will need to be renamed. This is what cohort.txt is for. In *integration.py*, a pair of functions parse the file and assign these column names to the .csv file automatically.
    c. **Each line in cohort.txt** corresponds to a column in the Sheet, in order. Think of it as the columns being flipped on their side to form rows instead.

```
Timestamp
Email
First Name
Last Name
LinkedIn
Pass1
Pass2
Pass3
Pass4
Give
Problem
Pass5
Ask
Pass6
Pass7
Pass8
Pass9
Pass10
Pass11
Location
Time Zone
Website
Pass12
```

Timestamp	Email	What's your name? (First	Last name	Please provide a link to y	How would you descrit
6/29/2023 23:35:52	sara.zeidan26@gmail.co	Sara	Zeidan		I've already started -- I
6/30/2023 14:09:50	mitadishant@gmail.com	Mita	Shah Bhagat	www.linkedin.com/in/mita	I've already started -- I
6/30/2023 14:55:36	emmayu0220@gmail.cor	Emma	Yu	https://www.linkedin.com	I'm sure I want to start
7/1/2023 9:23:55	tejasdutt.06@gmail.com	Tejas		https://www.linkedin.com	I'm sure I want to start
7/3/2023 3:44:47	vandanajain1601@gmail	VANDANA	JAIN	https://www.linkedin.com	I've already started -- I
7/4/2023 15:04:01	sophiadevine1969@gma	Sophia	Devine	https://www.linkedin.com	I've already started -- I
7/4/2023 17:51:24	ania.korsunska@gmail.c	Ania	Korsunska	https://www.linkedin.com	I've already started -- I
7/5/2023 6:49:29	lauren.vanoss@gmail.co	Lauren	van Oss	https://www.linkedin.com	I'd really like to start sc
7/5/2023 13:28:28	akhila@whatifdesign.in	Akhila	Kosaraju	https://www.linkedin.com	I've already started -- I
7/5/2023 15:08:17	op@sustaio.com	Olivia	Pedersen	din.com/in/olivia-pederse	I've already started -- I

Each row in the .txt corresponds to one column in the spreadsheet.

Each line in cohort.txt corresponds to a column in the Sheet, in order. This is the biggest pain point of the process: if the columns are not aligned with the lines in the .txt, then everything will be skewed, so **make sure it is up-to-date**! For relevant columns, assign a name from the list below; otherwise, you can name it anything. I use Pass1, Pass2, etc. to indicate columns that aren't relevant; but you can use anything, as long as you don't leave it blank.

d. The full list of attributes that are currently supported:
    i. First Name
    ii. Last Name
    iii. Email
    iv. LinkedIn
    v. Ask – what they are hoping to get out of the program/community.
    vi. Give – what they contribute to the program/community.
    vii. Problem – the climate problem(s) they're working on.
    viii. Website — a URL to their website if they have one.

5. Go to File → Download → Comma-Separated Values (CSV) and save the file.
6. (Optional) Drag the CSV file into the folder you created with your executable.

Running

1. (If you're on a Mac) The first time you try to run the executable, Apple will block it because it can't be scanned.
    a. Open System Settings → Privacy and Security
    b. Scroll down to near the bottom, and there should be a note saying that the system blocked an app called "main" from opening. Click "Open Anyway" and enter your password if necessary.
    c. The app should open.
2. Upload the .txt and .csv files you're using and type in the name of this cohort. Once all the necessary information is filled in, click "Add to Database".

Notes/Possible Issues

1. When the program checks the directory for duplicates, it is looking for *exact* copies of someone's email address. If someone didn't provide an email, this won't work.