Moving Forward with Coding

Final Lecture

Today’s Outline

- Dictionary Exercise!
- Let’s talk Github
- Review considerations when coding.
- Briefly touch on:
  - Python at home: finding out if you have it!
  - Various Editors / IDLE.
  - Some resources to look at for more excerises.
  - Finding a Python community.

Dictionary Exercise

- Create a historgram (shows the frequency of data) to count how many letters appear in a word.
- The output should be a dictionary. The method `dict()` creates an empty dictionary.
- So if you make a variable like `d = dict()` then anything you put into `d` will be output in a dictionary.
- `d['key'] = 'value'` will add to a dictionary:
- We are counting key letters, so values should be an integer that number of times that letter appears.

Dictionary Exercise

- Use the following structure:

```
Define function called histogram that (takes a string for an argument):
    create an empty dictionary
    for a letter in a word:
        if that letter is not in the empty dictionary:
            create a key/value in that dictionary. Value should be 1 to count how many times a letter appears.
        or else:
            if that letter is in the dictionary, increment value of the key/value pair by 1. (hint += 1)
    return the final dictionary
```

Exercise Answer

- My solution

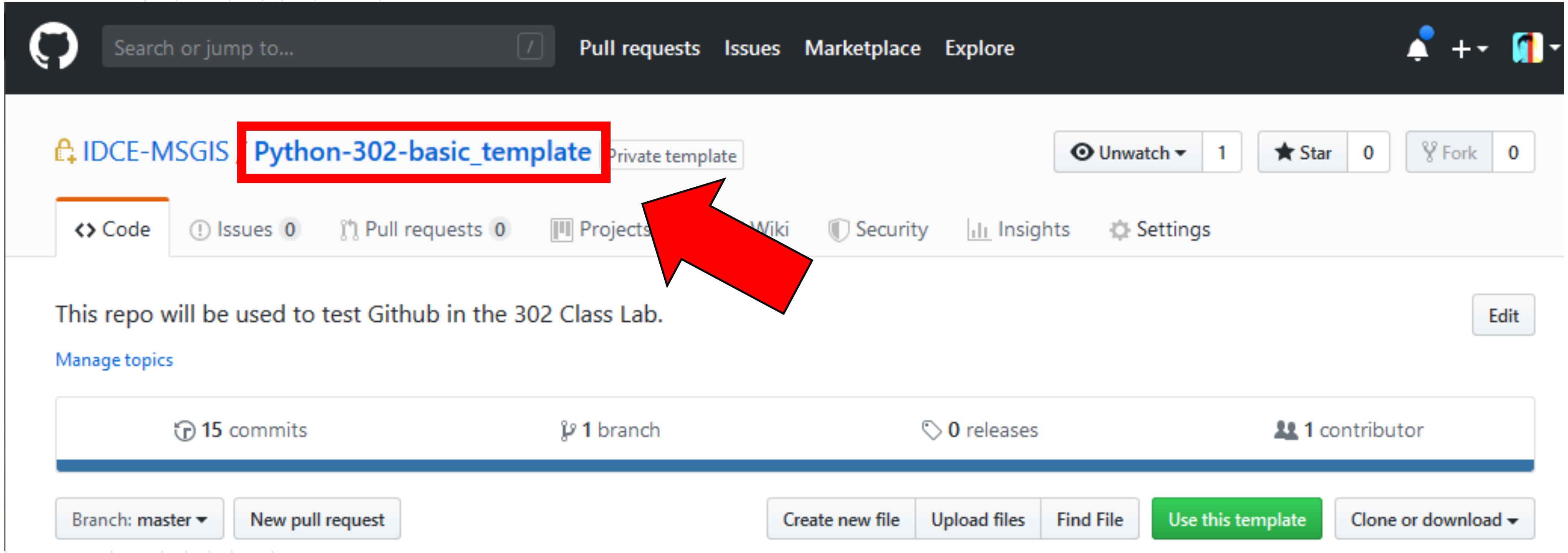
```
def histogram(word):
     d = dict()
     for letter in word:
          if letter not in d:
               d[letter] = 1
          else:
               d[letter] += 1
     return d

print(histogram('Massachusetts'))
```

Your Github Repos!

- We're going to give you your repos!
- We'll do the first lab today. After we're done grading, we'll send a notice via Moodle for you to transfer all of your labs to your personal Github account.
- They will appear as repositories.
- They will no longer be available to me or Priyanka.
- They will be public (if you have the free account)
- The steps look like this…

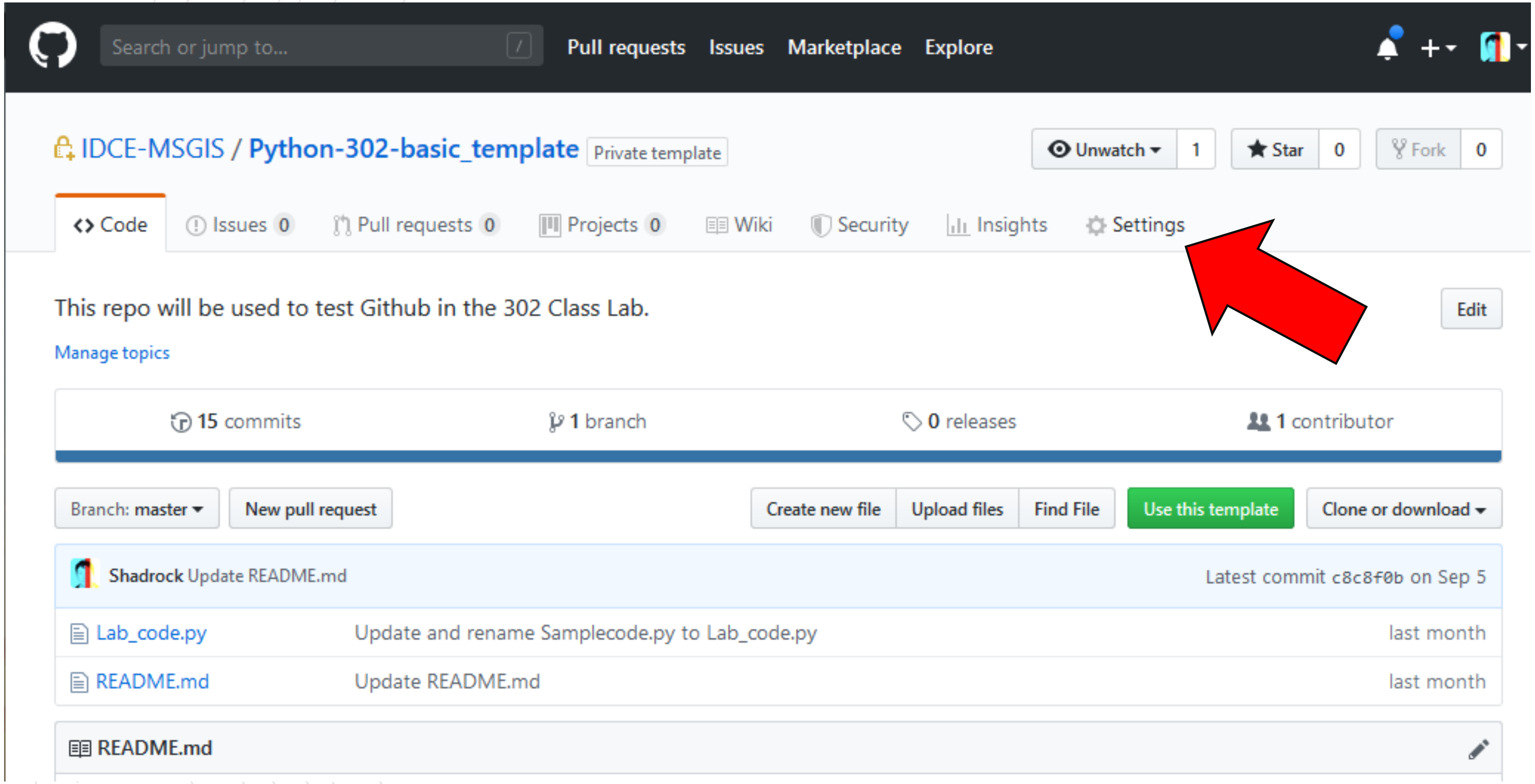
Copy Repo Name



This repo will be used to test Github in the 302 Class Lab.

- Do NOT copy the “IDCE-MSGIS” part.

Go to Settings



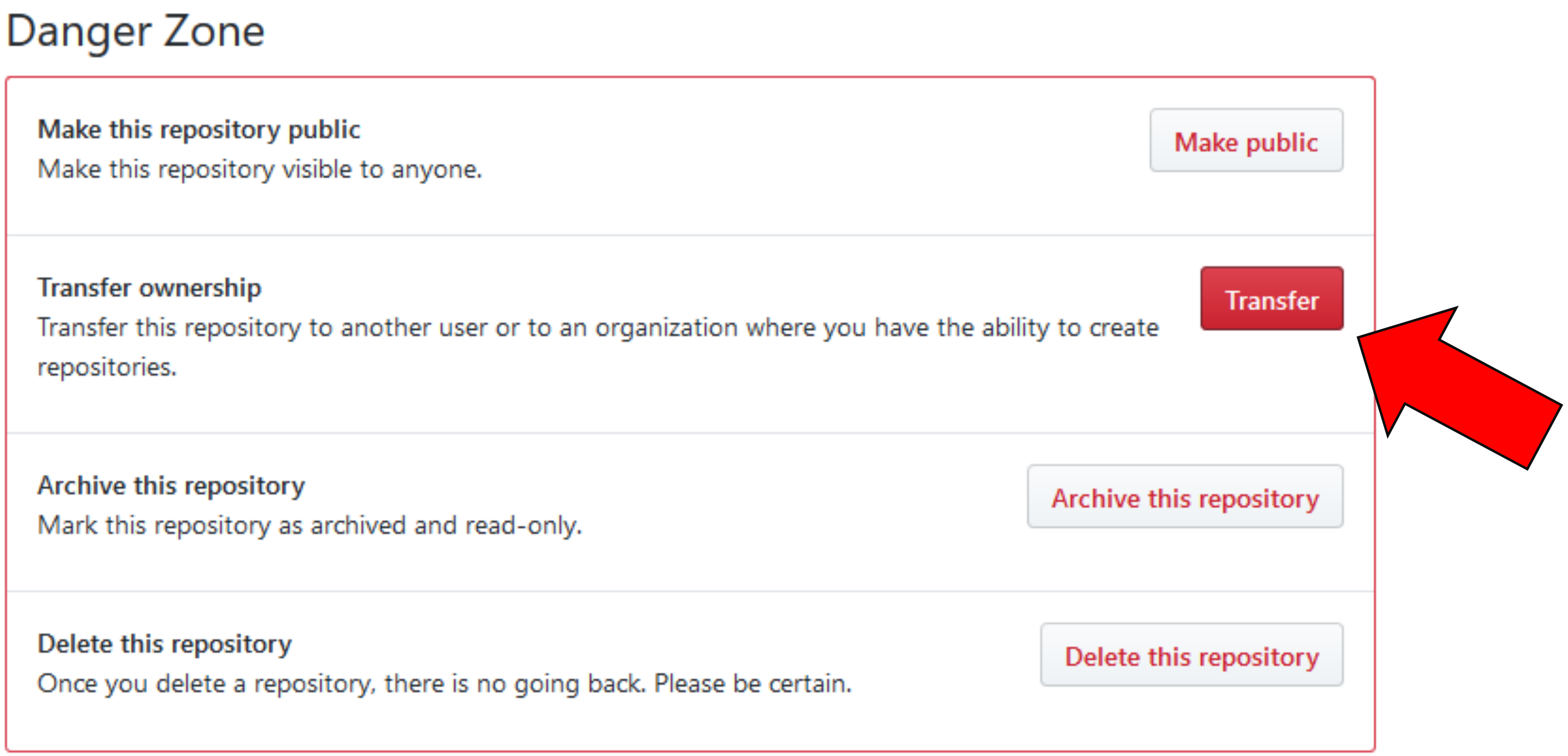
This repo will be used to test Github in the 302 Class Lab.

Shadrock Update README.md — Latest commit c8c8f0b on Sep 5

Lab_code.py	Update and rename Samplecode.py to Lab_code.py	last month
README.md	Update README.md	last month

README.md

Scroll down to Danger Zone & click “Transfer”



Danger Zone

Make this repository public
Make this repository visible to anyone.

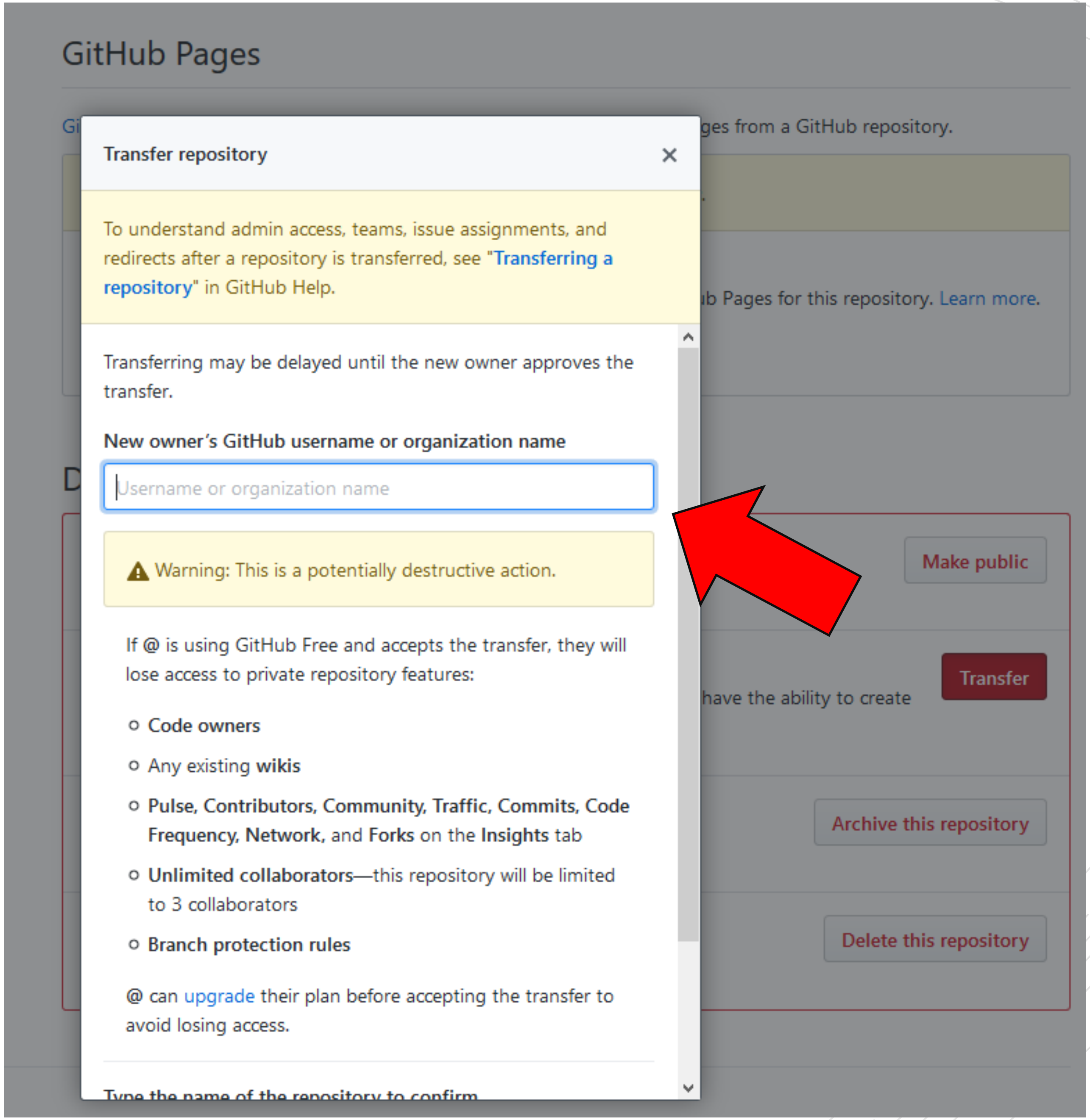
Transfer ownership
Transfer this repository to another user or to an organization where you have the ability to create repositories.

Archive this repository
Mark this repository as archived and read-only.

Delete this repository
Once you delete a repository, there is no going back. Please be certain.

Enter Github account.

- Scroll down.
- Enter / paste the name of the repo then click the “I understand” button.
- Check your personal account.



GitHub Pages

Transfer repository

To understand admin access, teams, issue assignments, and redirects after a repository is transferred, see "Transferring a repository" in GitHub Help.

Transferring may be delayed until the new owner approves the transfer.

New owner's GitHub username or organization name

Username or organization name

⚠ Warning: This is a potentially destructive action.

If @ is using GitHub Free and accepts the transfer, they will lose access to private repository features:

- Code owners
- Any existing **wikis**
- **Pulse, Contributors, Community, Traffic, Commits, Code Frequency, Network,** and **Forks** on the **Insights** tab
- **Unlimited collaborators**—this repository will be limited to 3 collaborators
- **Branch protection rules**

@ can upgrade their plan before accepting the transfer to avoid losing access.

Type the name of the repository to confirm

×
ges from a GitHub repository.
ub Pages for this repository. Learn more.
have the ability to create
Make public
Transfer
Archive this repository
Delete this repository

Github Repo Set up

- There's no right or wrong way to set up your repos.
- You may want to have one repo with all your labs as separate bits of code: you can explain them in your README.
- You may want each lab to be in its own repo.
- You may mix and match: most labs in a "lab repo" then your final project in a more polished repo.

Github Repo Set up

- There’s no right or wrong way to set up your repos.
- You may want to have one repo with all your labs as separate bits of code: you can explain them in your README.
- You may want each lab to be in its own repo.
- You may mix and match: most labs in a “lab repo” then your final project in a more polished repo.

Github Repo Set up

- Example of nested folders by class: https://github.com/talkpython/100daysofcode-with-python-course
- Example multiple scripts in one repo (lousy readme!) https://github.com/appium-boneyard/sample-code/tree/master/sample-code/examples/python
- Extremely detailed Readme for installing: https://github.com/evhart/crees also uses Github pages: https://evhart.github.io/crees/.
- Host a web page from your repo: https://pages.github.com

Always think about

- What you want to accomplish!
  - It helps me to write this out.
  - In my work, goals for code are spelled out in a “scope of work” or “Terms of reference” at the project level.
  - At the code level, well written issues or comments in Github.
- Documentation – Why?
  - Helps you remember what you’re doing!
  - Allows others to engage with your code or understand it!
  - Should always be a requirement on project work.

Always think about

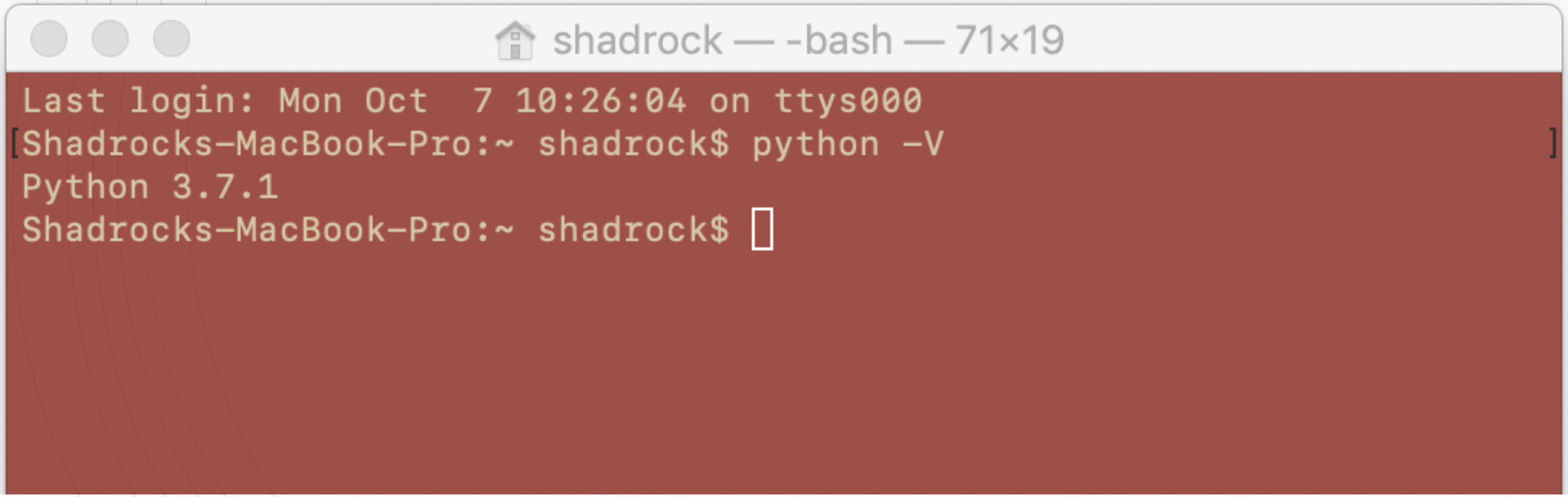
- Maintenance – why?
  - ‘Stale’ code: modules and dependencies change. Example?
  - What might a developer need to update somebody else’s code?
  - DOCUMENTATION!
- Accessibility – what are examples of different group that might want to access your code?

Always think about

- Open Source?
- Not everything must be open source. Ex. Ushahidi
- Just because something is publicly accessible doesn't mean it's open source. You must have the right license.
- Example: https://creativecommons.org/licenses

Python at Home

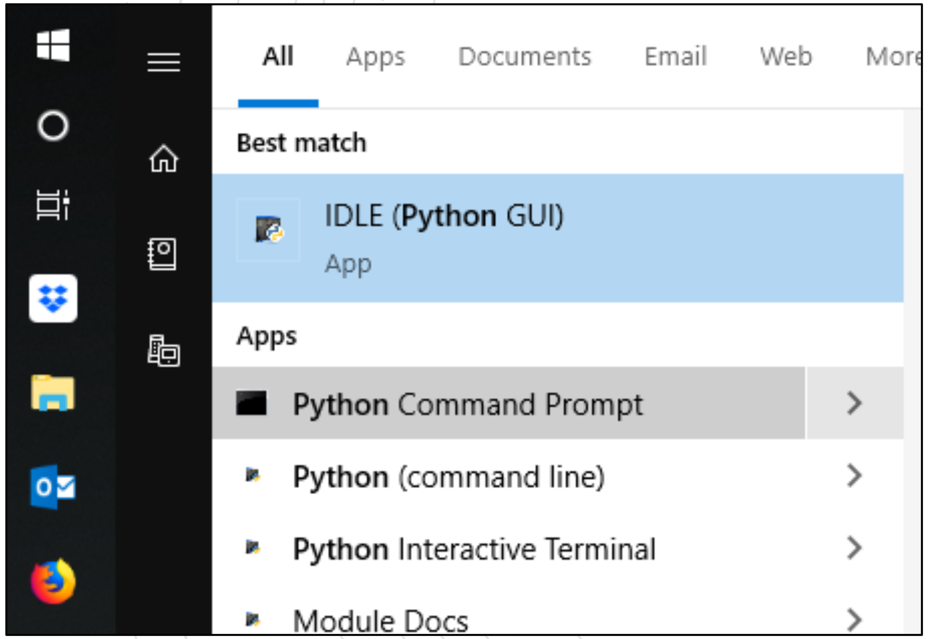
- Do you have Python already?
  - Mac – open terminal and type **python –V**

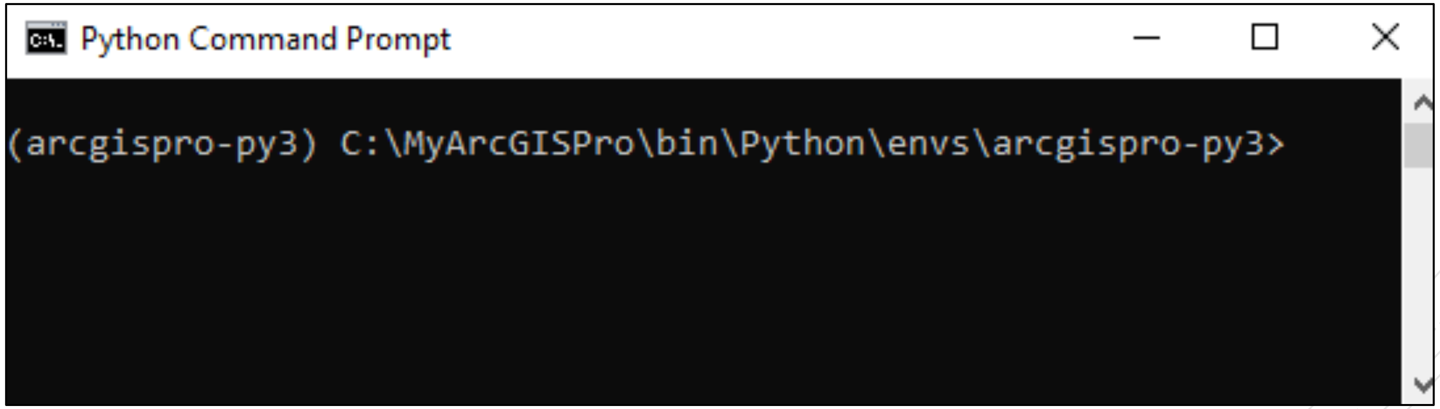


```
Last login: Mon Oct  7 10:26:04 on ttys000
Shadrocks-MacBook-Pro:~ shadrock$ python -V
Python 3.7.1
Shadrocks-MacBook-Pro:~ shadrock$ 
```

Python at Home

- Do you have Python already?
  - Windows – search from taskbar and (if you have it) choose open in Command line.





```
(arcgispro-py3) C:\MyArcGISPro\bin\Python\envs\arcgispro-py3>
```

Python at Home

- If you *don't* have Python already you'll need to download it, which can be done via lots of different place. A few to recommend:
  - https://realpython.com/installing-python/ (Windows and Mac)
  - Good Windows walkthrough from UC, Irvine: https://www.ics.uci.edu/~pattis/common/handouts/pythoneclipsejava/python.html
  - Mac is a bit more complicated… good walkthrough here: https://docs.python-guide.org/starting/install3/osx/
  - Good Mac walkthrough from PyLadies: http://www.pyladies.com/blog/Get-Your-Mac-Ready-for-Python-Programming/

Python Editors / IDLE

- Atom Editor: https://atom.io
  - Available for PC and Mac.
  - Active Slack group! Great place to ask for help.
- PyCharm: https://www.jetbrains.com
  - Available for PC and Mac
  - Has an active community forum where people ask for help
- Jupyter Notebooks: https://jupyter.org
  - open-source web application that allows you to create and share documents that contain live code,

Resources

- Open source text full of great exercises with practical applications: https://automatetheboringstuff.com
- Also available as an online course through "Udemy."
- Towards Data Science blog on Medium: https://towardsdatascience.com
    - Eg. This web scraping tutorial: https://towardsdatascience.com/how-to-web-scrape-with-python-in-4-minutes-bc49186a8460 (spoiler alert, takes a bit longer than 4 minutes!)

Community

- Meetups!
  - https://www.meetup.com/bostonpython
  - Presentations, project working sessions, etc.
- Hackathons / Code a thons.
  - Technotopia in Worcester? https://www.meetup.com/Technocopia-Open-Hack-and-Craft-Nights/
  - Worcester Code Camp: https://www.meetup.com/Lets-Learn-Programming-in-Worcester-MA/ Python night Jan 8!
- Become an open source contributor (e.g. Ushahidi)

Summary

- Created a histogram function. Makes a great Christmas present.
- You have a repo! More to come! Also a great present.
- Things to consider before coding: the goal, the user, etc.
- There are TONS of resources out there. Explore them and have fun. Hope you'll find some of the resources useful.