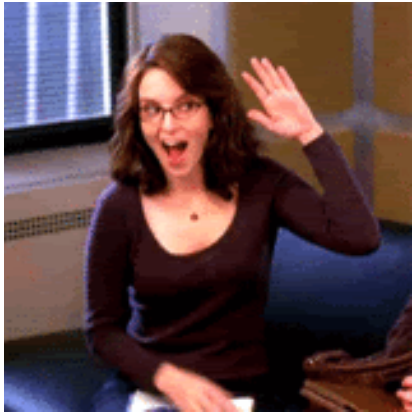




Project Time!

You've Made It!

Congratulations on making it this far!



You Do: Project

For the final project, you're making a new app of your choosing from scratch.

If you're having trouble coming up with a topic, consider:

- Find a fun API and base it on that.
- Make an “About Me” site that's all about you.
- Make a portfolio website for one of your hobbies.
- Make a blog about a topic you find interesting.
- Make something you think would improve your life in some way.
- Feel free to share resources and inspiration with your classmates!

Deliverables

You must have a Flask site running locally on your own machine.

You will work individually on this project, but feel free to share inspiration, resources, or cool datasets that you find with your classmates!

Requirements

Your assignment **must** include:

1. At least 3 working routes with associated views
 - Including at least one `GET` and one `POST` route
2. Data pulled from at least one API
 - Get creative! Tons of free APIs exist! Ask your instructor or classmates for ideas!
3. Have semantically clean HTML and CSS applied to a template - at least a small amount of styling.
 - Try to spend a little time styling your pages and making them look nice!
4. Core Python topics. At minimum:
 - Dictionaries *or* sets, *or* tuples.
 - `**args` *or* `kwargs` *or* `*kwargs`.
 - Basic debugging, such as a `try/except` block, (*only if necessary*).
 - A class.
 - User input *or* reading from a file.
5. Comments, so another developer can easily see what your app does.

Bonus: Have extra time? Ask how to deploy a Flask site to a cloud service like [Heroku](#)! Or, try to follow the directions on this [medium article](#).

Suggested Ways to Get Started

- **Begin with the end in mind.** Know where you want to go by planning ahead, so you don't waste time building things you don't need.
- **Read the docs** for whatever technologies or APIs you use. Most of the time, there is a tutorial that you can follow! This isn't always the case, though, learning to read documentation is crucial to your success as a developer.
- **Write pseudocode before you write actual code.** Thinking through the logic of something helps.

Useful Resources

- [A list of free APIs](#)
- An [extremely helpful debugging flowchart](#)
- The [Python Docs](#)
- [Keyword Args](#)
- [Args and Kwargs](#)
- [Chain and Other Itertools](#)
- [Sets, on a Python tutorial website](#))
- [Tuples](#)
- Writing Good [User Stories](#) (for a few user story tips)
- [Presenting Information Architecture](#) (for more insight into wireframing)
- [Heroku](#) (for hosting your back-end)

Evaluation

Your project will be evaluated based on the rubric below.

Score	Expectations
0	No credit - Incomplete
1	Half credit - Working app without all requirements
2	Full credit - Working app with all requirements
3	Extra credit - Working app with bonus enhancements

Let's Get Started!

Open the Final Project Prompt { `final-project-prompt` } to see the project and its instructions. Follow the instructions there.

Remember:

- You're making any app you'd like! Find something that interests you.
- You must have a website with at least 3 working routes with associated views.
- Don't be afraid to ask for help!