Download the Prerecs:
DO THIS NOW!!!!

https://ln.cobular.com/intro

Your First Webapp

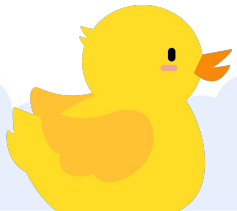
w/ HTML, CSS, JS, Flask

– Julie Cover

Attendance!

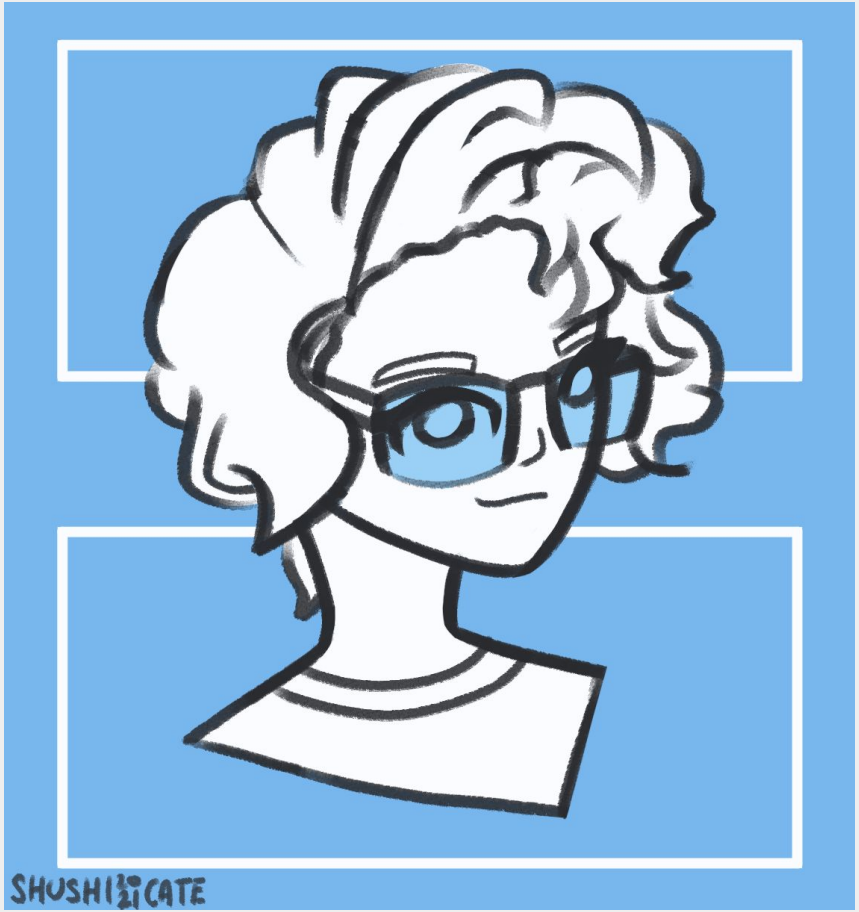
Best attendance will receive a prize during the closing ceremony!

https://tinyurl.com/lahacks23attendance



Me!
Julie Cover

- she/her
- CSE @ UCLA, Class of 2024
- Hackathon Organizer, Database Deleter, Kinda Fruity, Curly Hair Haver



What You're Gonna Do

1. Make your own website (that uses ChatGPT!)
2. Connect it to your own backend

And if you come to my later workshop…

3. Host it on the cloud!

Questions - Raise hands, ask blue shirts around you!

Step -1: VS Code and Starter Code

Hopefully, you did this before you came!

We don’t have time to wait more than like 5 minutes for this, but in case you missed a step or two:

1. Download VS Code
2. Download the Starter Code
3. Install Python 3.10

https://ln.cobular.com/intro

(both available at the QR Code up there!)

HTML

HyperText Markup Language

The thing with `<div></div>`

CSS

Cascading Style Sheets

The thing with:

```
.class {
    margin: auto;
}
```

JavaScript

JavaScript

Coding on the web! Looks kinda like C++ but without types

Flask

Python web framework

The part that gets data and sends your site to users!

First Steps - Hello Web!

1. Open the folder in VS Code
2. Open a terminal with **CTRL + `** at the same time
   a. ` is the key to the left of 1
3. Run **`python ./server.py`**

Go to **http://localhost:8000/** in your browser and click on **`part_1_html`**

HTML - The Skeleton 💀

The shape of your site

What features do you want?

HTML - Things We're Gonna Do

Show The Ideas

Show The Cost

Get More Ideas

HTML - Headings

We can show a **BIG HEADING** with the **<h1>** tag!

1. Find the first FIXME in the part 1 **index.html** file and turn the empty tags (<> </>) into *heading 1 tags*:
   a. **<h1>(your big title here)</h1>**
2. Reload the page to see the change!

HTML - Lists

A bit further down, we have a List!

HTML Lists are made of an outer *unordered list* **`<ul>`** and an inner *list item* **`<li>`**

Add 4 more list items by copying the **`<li>(text)<\li>`** tags

HTML - Learn More

Head over to **https://ln.cobular.com/html** to learn more about all the different HTML tags!

CSS - The Makeup 💅

Making things look pretty

How do you want the pieces from before to look and fit together?

CSS - Types of Styles

Open `part_2_css/styles.css`!

Element Styles

`body {}`

Applies to all elements with the same name

ID Styles

`#thing {}`

Applies to the single element with that ID

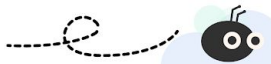
IDs are unique!

Class Styles

`.eee {}`

Applies to all elements with the class

Elements can have multiple classes, multiple classes can be on the same element



CSS - Try it out!

Add the **`number`** class to the **`<span>`** element

```
<span class="number">
```

Reload the page and watch the styles defined in the css file be applied!

CSS - Try it Out!

The title should be bigger!

Add this line to the **`h1 {}`** section at the end, inside the braces:

```
font-size: 3rem;
```

To change the font size from 2 to 3 REM. Reload after saving!

JS - The Muscles 💪🏼

Making things move, change, and react to inputs on a page!

Make it easier to later load more data

JS Basics

Don’t worry if you don’t know this stuff! Just a quick overview

Syntax: C-like

Types: Not Really

Semicolons: Eh if you want

Compiled: Also No

Speed: Kinda mid

JS - JQuery

Makes stuff easier!

Superpower: the **`$()`** function

```
let button = $("css selector")
```

Gives you a reference to an HTML component, which you can do anything with!

JS - Try It Out

Open **`part_3_js/script.js`**

This is set up already to be automatically be loaded before the page shows up!

Add a CSS selector to the parenthesis to get a reference to the button by it's ID

Remember, the button looks like this:

```
<button id="fetch_data">
```

JS - Try It Out

To reload a page with code, you need to call the following function:

```
location.reload()
```

Flask - The Brains 🧠

Sends your files

Fills in info automatically!

Stores secrets, is fast, secure, etc.

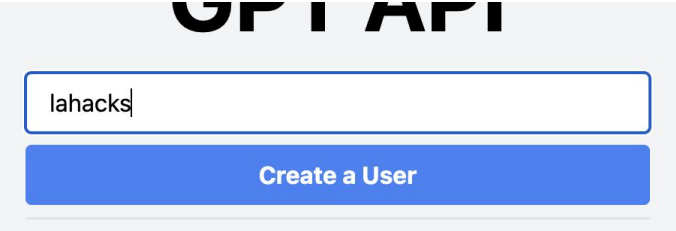
0. Using an API

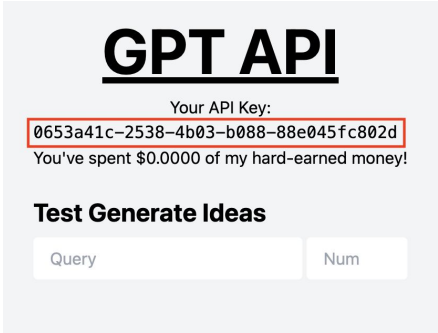
Keys and Requests

1. Get YOUR Api Key!

1. Go to https://gpt.cobular.com
2. Enter the secret key “lahacks” (lowercase, one word)
3. COPY YOUR KEY!!!

Give it a test if you want! Put in a prompt, like “funny dog names” and see what you get back!





Test the API

1. Open `test.py`
2. Put in your key at the top
3. Put in your prompt next
4. Run `python ./test.py` to get responses IN CODE!!

API Considerations

Security + Billing

Call APIs on the backend

Implement the API

1. Fill in **`part_4_flask/app.py`**
   a. API Key
   b. Prompt
2. Copy over CSS & JS files to **`part_4_flask/static`**
3. Copy contents of body from step 3 to the **`index.jinja`**
   a. Update the **`<ul>`** for the loop version in the comment!

Run Flask Server

1. Kill the old server
    a. Go to the terminal that was running it
    b. Type **CTRL + C** at the same time to kill the server
2. Move to the part 4 folder
    a. `cd ./part_4_flask`
3. Install the files
    a. `pip install -r requirements.txt`
4. Run the new server!
    a. `flask run --debug`
5. View the site
    a. Go to `localhost:5000`