Zoom Lecture Overview

You will enter with audio muted and video off

- Please keep them that way unless asking a question

To ask a question audibly

- "Raise your hand" and wait for us to acknowledge you
- Unmute yourself and ask your question
- Mute yourself again when done

To ask a question in text

- Type it in chat, the CA will read it out loud
- Please reserve chat only for questions

Notes

- Lectures are recorded, including your audio and text
- Aiming to post lecture slides ahead of time

Welcome to CS193X

Michael Chang

Spring 2020

A unique quarter

If you haven't seen it, please read the message we sent to enrolled students

- Reach out to us at any time

Plan for today

Course overview and logistics

- More than usual...

Brief context: the Internet

- Servers, web browsers, accessing a web page

Introduction to HTML

- File structure, tags
- Elements, attributes
- How pages are laid out

Introduction to styling pages

My experience with web dev

Work on software tools for the CS dept

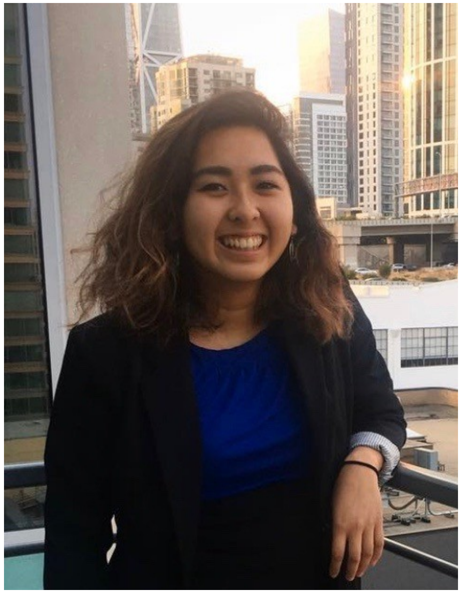
- Paperless, autograder infrastructure

Learned a bunch of web dev through tutorials, blogs, even reading the standard

- There's a lot of content. A lot of it is wrong/outdated

Hoping to point out some highlights

Our awesome CAs



Course website

https://cs193x.stanford.edu

Requires Stanford login, as it has links to lectures

Enrollment

Limited to 100 students

- ~90 enrolled, waiting to hear from ~15
- Waitlist of ~30

Use this tool (also on website) to check your status

- If you have a permission number, please enroll soon

Will try to settle the numbers by end of week

Auditing

Welcome to audit the course

- Use website for announcements and materials

You can join the forum

- But please be mindful of staff time

Can't accept assignment submissions or accommodate in office hours

Will send out a form to see who's auditing

CS142 vs. CS193X

CS142

Focus on systems principles

Uses web dev as tool to talk about larger systems

Uses frontend framework (React) to build larger apps

Well-established class, stable materials

Prereqs: 107 and 108

Counts toward undergrad major

CS142 vs. CS193X

CS193X

- "Practical skills" class
- Focus on core concepts in web dev
- No frontend framework, modern standardized tech
- Newer course, materials in flux
- Prereqs: 106S
- Does not count toward undergrad program sheet

Our Approach

Web dev is a huge space

- Constantly changing, often a bit slow to converge
- Many ways to solve the same problem

We will be "opinionated"

- Show you one way to do things
- Won't necessarily be best, but it will be one we've seen a lot, based on reliable sources

No frontend framework

- By the end, we hope you can pick up new frameworks and tech quickly

Coursework

Assignments

- 4 assignments + setup/survey assignment 0
- Roughly 1.5-2 weeks apart
- Specific tasks to complete
- Individual

Assignment 0

- Survey to get to know you
- Could have been a Google form
- But also a chance to get your software set up

Coursework

Assignments

- 4 assignments + setup/survey assignment 0
- Roughly 1.5-2 weeks apart
- Specific tasks to complete
- Individual

Project

- Your chance to be created, make something cool
- Milestones throughout the quarter, due roughly with assignments (except assign0)
- Individual

Grading

Bucket scale, like 106s

- Functionality and style, equal weight

Buckets

- +: met all requirements, great work
- ok: solid work, some improvements/issues
- -: more significant issues
- --: incomplete or multiple major issues
- 0: no submission

Grading

Late policy

- 4 "late days", 24 hour
- Intended for use for unexpected situations
- At most 2 days late for each assignment/milestone (with exceptions)
- Work not accepted after you're out of late days

We want to be flexible

- Please reach out to us if you're in a tight spot

Honor Code and Collaboration

- **Please read the collaboration polcy fully**
- **We know this is a stressful time**
  - But this is no time to short change yourself
- **Summary**
  - Write your own code
  - Do your own design and debugging
    - But solicit and offer tips
  - Seek clarification, discuss course topics
  - Look up background info, references
  - Cite your sources of inspiration, ideas, fixes

Getting help

Ed Discussion Forum

- Ask and answer questions
- Can make questions private, but if general question, please make public
- Can be anonymous to students, but we encourage an open, positive community

Office hours

- Calendar posted on website, with Zoom links
- Can help with assignment-specific issues, or just chat

Email

- For grading concerns, more personal'/sensitive issues

Topic outline

Web page layout in the browser (2 weeks)

- HTML, CSS

Client-side Javascript (3-4 weeks)

- Interactive pages, events
- APIs, dynamic content

Server-side Javascript (2 weeks)

- Simple web server, creating an API
- Data storage with MongoDB

Full stack topics (1.5-2 weeks)

- Mobile and accessible design
- Security considerations

Plan for today

Course overview and logistics

- More than usual...

Brief context: the Internet

- Servers, web browsers, accessing a web page

Introduction to HTML

- File structure, tags
- Elements, attributes
- How pages are laid out

Introduction to styling pages

The Internet

When you type an address (URL) in a browser

- Find the server that has the web page
- Ask the server for the page
- Display the page

Servers and Ports

Each server (computer) can offer many things

- Web, email, remote login (SSH)

How to know which program a request belongs to

- Each program listens on a "port" (a number)
- Client says which port it wants to talk to
- Standard ports for web, mail, etc.
- Can use custom ports for more unusual services, internal use only, etc.

Protocols

How do the client and server "talk?"

- Protocol: set of "messages" written in a standard
- Any program that "speaks" a protocol can communicate with any other
- E.g. different browsers, different web servers

Example (demo): HTTP

Why talk about this now

- **We could technically just double-click web pages to open them**
  - But we'll run into many problems later
- **Instead, we will run a web server on localhost**
  - Provided in the starter code
  - localhost: always refers to the local machine
  - Others can't access your server (for security)
  - Custom port (also security)

Plan for today

Course overview and logistics

More than usual...

Brief context: the Internet

Servers, web browsers, accessing a web page

Introduction to HTML

File structure, tags

Elements, attributes

How pages are laid out

Introduction to styling pages

Demo: Intro HTML

See website for link to demo code

Takeaways

HTML is a markup language

- Not a program, doesn't define logic
- Structured as a tree

element: a node in the tree

- Has a parent, most can have children
- Can contain text

tag: marks the start or end of a node

- E.g. `<p> ... </p>`

Takeaways

Page laid out in order

- Some elements laid out top to bottom, some left to right

Element define semantics

- <p> (paragraph), <a> (link), `<section>`

Elements have default presentation

- <p>: Space above and below
- <a>: Underlined with a color
- `<section>`: Nothing!

These styles can be overridden

- But not the semantics (unless you work hard, but don't)

Summary

For next time

- Follow up about enrollment
- Join the forum
- Look at assignment 0

Next time: styling and CSS

- Ways to define styles
- Overriding defaults
- Colors, borders, spacing, fonts