

Coding Survey



<https://shorturl.at/3UaNt>

6/11.C35, 6/11.C85

Interactive Data Visualization & Society

Programming Labs

TAs



Dana Hua



Sophia Zheng



Simone Peter



Riccardo Fiorista



Ane Zuniga



Zoe De Simone

Logistics

- Undergraduates (6/11.C35):
 - Mandatory
- Graduates (6/11.C85):
 - Optional for course 6 students (EECS)
 - Mandatory for course 11 students (DUSP)
 - Enroll in 11.S942 to get extra 3 credits for the lab work.
- Attendance to in-person lab sessions is optional but highly recommended.

Lab structure

- Lab structure:
 - Part 1: lecture + activities (15-30mins)
 - Part 2: hands on work (30-45mins)
 - Part 3: async work at home (~2hours)
- Lab checkoffs: show your work before the next lab
 - Come to any TA Office Hours: <https://vis-society.github.io/logistics/staff.html#office-hours>

Lab 1:

Introduction to the Web Platform

02/06/2025

HOW DOES THE WEB WORK? What is a URL?

A **URL** (Uniform Resource Locator) is the address of a unique resource on the internet. It allows users and systems to access specific resources on the web.

HOW DOES THE WEB WORK? What is a URL?

A **URL** (Uniform Resource Locator) is the address of a unique resource on the internet. It allows users and systems to access specific resources on the web.

For example:

`https://vis-society.github.io/labs/1/hello.html`

HOW DOES THE WEB WORK? Breaking down a URL

A **URL** (Uniform Resource Locator) is the address of a unique resource on the internet. It allows users and systems to access specific resources on the web.

For example:

https://vis-society.github.io/labs/1/hello.html



Protocol: defines the rules for communication

HOW DOES THE WEB WORK? Breaking down a URL

A **URL** (Uniform Resource Locator) is the address of a unique resource on the internet. It allows users and systems to access specific resources on the web.

For example:

`https://vis-society.github.io/labs/1/hello.html`



Host: identifies the web server

HOW DOES THE WEB WORK? Breaking down a URL

A **URL** (Uniform Resource Locator) is the address of a unique resource on the internet. It allows users and systems to access specific resources on the web.

For example:

`https://vis-society.github.io/labs/1/hello.html`



Path: specifies the location of the file on the server.

HOW DOES THE WEB WORK? Relative URLs

Relative URLs are useful when working within a website to reduce redundancy (no need to repeatedly specify *protocol* and *domain*).

HOW DOES THE WEB WORK? Relative URLs

Relative URLs are useful when working within a website to reduce redundancy (no need to repeatedly specify *protocol* and *domain*).

For example:

- Relative URL: `hello.html`
- Base URL: `https://vis-society.github.io/labs/1/index.html`
- Absolute URL: `https://vis-society.github.io/labs/1/hello.html`

HOW DOES THE WEB WORK? Relative URLs

Shortcut symbols for navigation:

- `/` → Relative to the root (starts from the website's root directory)
- `..` → Navigate up (moves one level up)
- `.` → Current directory

[\[More info\]](#)

HOW DOES THE WEB WORK? Relative URLs

Shortcut symbols for navigation:

- `/` → Relative to the root (starts from the website's root directory)
- `..` → Navigate up (moves one level up)
- `.` → Current directory

For example:

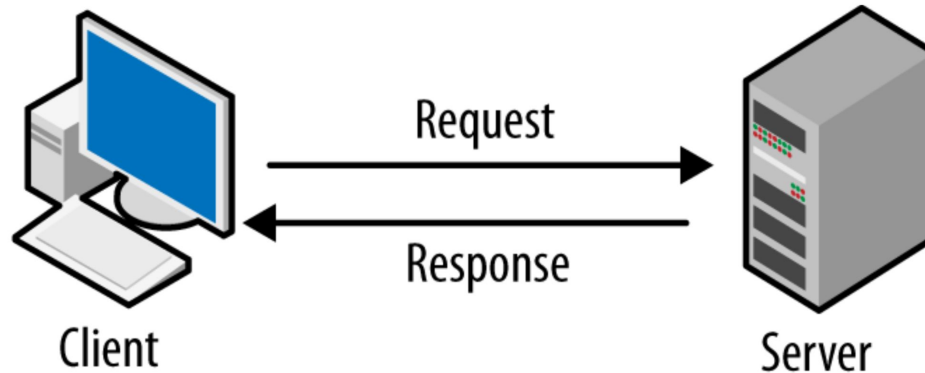
- `/images/photo.jpg` → Loads `photo.jpg` from the root `/images/` folder.
- `../about.html` → Moves up one level and accesses `about.html`.
- `./style.css` → Loads `style.css` from the current directory.

[\[More info\]](#)

CLIENTS & SERVERS

The Web consists of clients (browsers, apps) and servers (web hosts).

- **Clients** make requests for resources
- **Servers** listen for such requests and respond with the resources (or an error)



THE WEB: OPEN BY DESIGN

The Web is designed to be **transparent**. You can view and modify the webpage code using **DevTools**.

Vis & Society
2025

Home

Schedule

Theme: Housing

Affordability

Assignments

A1: Data, Visualization,
and Housing

A2: Exploratory Visual
Analysis

A3: Visualization Design

A4: Persuasive/Deceptive
Visualization

Final Project

Programming Labs

Reading Group

Course Logistics

Course Staff & Office
Hours

Grading Policies

Inclusion and Respect

Resources

This site uses [Just the Docs](#), a documentation theme for Jekyll.

Search Vis & Society 2025

CanvasFeedback Form

MIT Interactive Visualization & Society
(Spring 2025)

The world is awash with increasing amounts of data, and we must keep afloat with our relatively constant perceptual and cognitive abilities. Visualization provides one means of combating information overload, as a well-designed visual encoding can supplant cognitive calculations with simpler perceptual inferences and improve comprehension, memory, and decision making.

Moreover, visual representations may help engage more diverse audiences in the process of analytic thinking.

By the end of this course, you should expect to be able to:

- 1 Understand key visualization techniques and theory.
- 2 Design, evaluate, and critique visualization designs.
- 3 Wrangle and explore datasets through visualization using Tableau.
- 4 Tell stories with data, and use visualization for policy change.
- 5 Implement interactive data visualizations using D3.js.
- 6 Develop a substantial visualization project.
- 7 Read and discuss visualization research papers (graduate students only).

Theme: Housing Affordability in Metro Boston

Each year, the class works with a "client" focusing on a theme. This year, our partner is the [Metropolitan Area Planning Council \(MAPC\)](#), a public planning agency whose goals are to promote regional collaboration around the following goals: sound municipal management, sustainable land use, protection of natural resources, efficient and affordable transportation, a diverse housing stock,

ElementsConsoleSourcesNetwork

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <body>
      <a class="skip-to-main" href="#main-content">Skip to main content</a>
      <svg xmlns="http://www.w3.org/2000/svg" class="d-none">
        <symbol id="svg-link" viewBox="0 0 24 24">
          <title>Link</title>
          <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentcolor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-link">
            <path d="M18 13a5 5 0 0 7.54 5.43" stroke="currentcolor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-link">
            <path d="M14 11a5 5 0 0 7.54 5.43" stroke="currentcolor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-link">
          </svg>
        </symbol>
        <symbol id="svg-menu" viewBox="0 0 24 24">
          <title>Menu</title>
          <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentcolor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-menu">
            <line x1="3" y1="12" x2="21" y2="12">
            <line x1="3" y1="6" x2="21" y2="6">
            <line x1="3" y1="18" x2="21" y2="18">
          </svg>
        </symbol>
      </svg>
    </body>
  </html>
```

htmlbodysvg.d-nonesymbol#svg-menusvg.feather.feather-menu

StylesComputedLayoutEvent ListenersDOM BreakpointsProperties

Filter

element.style {

{

box-sizing: border-box;

fill: none;

stroke: currentcolor;

stroke-width: 2;

stroke-linecap: round;

stroke-linejoin: round;

svg:not(:root) {

overflow: clip-margin; content-box;

overflow: hidden;

:not(:foreignObject) > svg {

THE WEB PLATFORM

The collection of technologies that we use to create websites and web applications is called **The Web Platform**.

THE WEB PLATFORM

The collection of technologies that we use to create websites and web applications is called **The Web Platform**.

It consists of three main technologies: **HTML**, **CSS**, and **JavaScript**.

THE WEB PLATFORM

- **HTML:**

- **Structure** of the webpage (e.g., headings, paragraphs, lists).

- **CSS:**

- **Presentation**: defines styles like colors, fonts, and layouts.

- **JavaScript:**

- **Behavior**: adds interactivity (clicks, animations, dynamic content).

THE WEB PLATFORM

- **HTML:**

- **Structure** of the webpage (e.g., headings, paragraphs, lists).

- **CSS:**

- **Presentation**: defines styles like colors, fonts, and layouts.



HTML



HTML + CSS



HTML+CSS+JavaScript

- **JavaScript:**

- **Behavior**: adds interactivity (clicks, animations, dynamic content).

Image source: <https://medium.com/@rashmipaboda2/basic-concepts-of-html-e9ce7c7e4e82>

HTML ELEMENTS

Syntactically, HTML elements are made up of **tags**, which are sequences of characters enclosed in angle brackets.

HTML ELEMENTS

Syntactically, HTML elements are made up of **tags**, which are sequences of characters enclosed in angle brackets.

- Structural elements: `<h1>`, `<p>`, `<section>`, `<article>`, `<header>`, `<footer>`.
- Lists and tables: ``, ``, `<table>`, `<tr>`, `<td>`.
- Forms and user input: `<form>`, `<input>`, `<button>`, `<textarea>`.
- Multimedia: ``, `<video>`, `<audio>`.
- Interactive elements: `<details>`, `<summary>`, `<dialog>`.

HTML ELEMENTS

Syntactically, HTML elements are made up of **tags**, which are sequences of characters enclosed in angle brackets.

- Structural elements: `<h1>`, `<p>`, `<section>`
- Lists and tables: ``, ``, `<table>`
- Forms and user input: `<form>`, `<input>`
- Multimedia: ``, `<video>`
- Interactive elements: `<details>`, `<summary>`

The HTML between the starting and ending tag is called the **content** of the element, and may contain raw text or other HTML elements.

HTML ELEMENTS

The HTML between the starting and ending tag is called the **content** of the element, and may contain raw text or other HTML elements.

- `<h1>` element (top-level heading)

`<h1>` Hello world! `</h1>`

Starting/opening tag Element content Ending/closing tag

HTML ELEMENTS

The HTML between the starting and ending tag is called the **content** of the element, and may contain raw text or other HTML elements.

- `<h1>` element (top-level heading)

```
<h1>Hello world!</h1>
```

Starting/opening tag	Element content	Ending/closing tag
----------------------	-----------------	--------------------

NOTE: Many HTML elements do come with some default styling, for example a heading element will be rendered in bold and a larger font size. However, it is an antipattern to use HTML elements to style content, that's what CSS is for!

Emmet's HTML skeleton



Emmet's HTML skeleton



```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Document</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

Emmet's HTML skeleton



```
<!DOCTYPE html>
```

Document type declaration

```
<html lang="en">
```

```
<head>
```

Contains metadata about the page

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Document</title>
```

Title of the page, which is shown in the browser tab

```
</head>
```

```
<body>
```

Contains the visible content of the page

```
</body>
```

```
</html>
```

Part 2: hands-on work!

Coding Survey



<https://shorturl.at/3UaNt>