Course: Web Application Development
Lab Instructor: Msc. Nguyen Trung Nghia

Email: ntnghia@hcmiu.edu.vn

Lab 1: HTML & CSS

Date: Sep 22th, 2025

Tutorial:

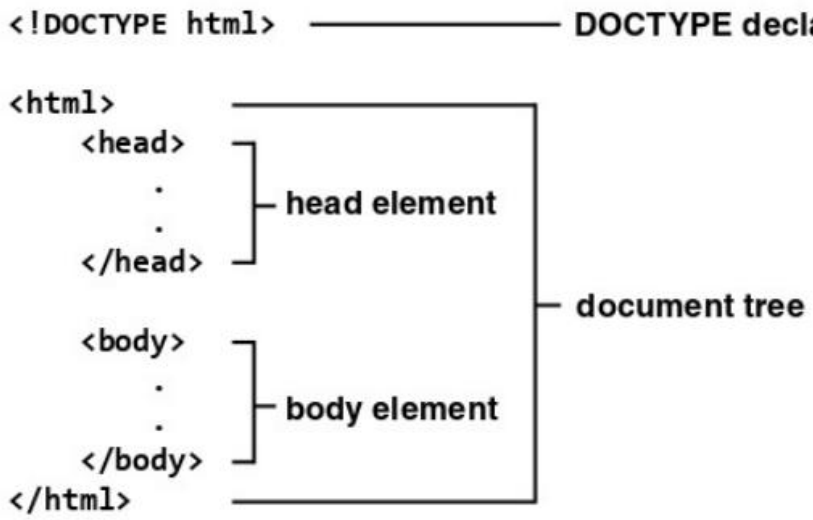
- HTML Learn HTML in 1 hour
- CSS (Basic) Learn CSS in 20 Minutes (youtube.com)
- CSS (Full) Learn CSS in 1 hour (youtube.com)

Duration: 3 hours

Part 1: Basic HTML

`

The basic structure of an HTML document

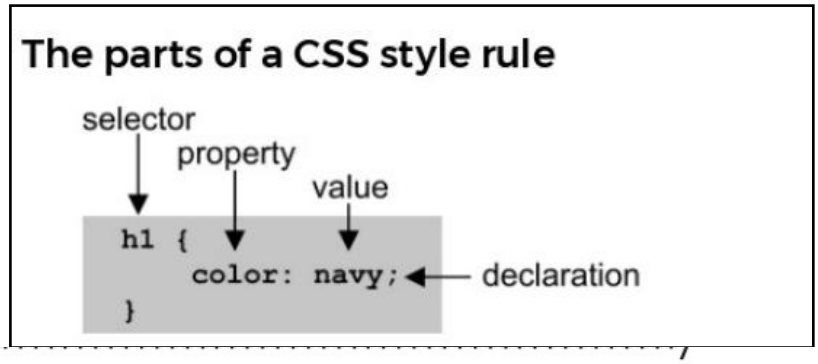


A simple HTML document

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>San Joaquin Valley Town Hall</title>
    </head>
    <body>
        <h1>San Joaquin Valley Town Hall</h1>
        <p>Welcome to San Joaquin Valley Town Hall.</p>
        <p>We have some amazing speakers in store for you this season!</p>
        <p><a href="speakers.html">Speaker information</a></p>
    </body>
</html>
```

Part 2: Basic CSS

Run and try to understand the example that you are given on BB




```
/* Adjust the styles for the body */
body {
    background-color: #FACD8A;          /* This is a shade of orange. */
}

/* Adjust the styles for the headings */
h1 {
    color: #363636;
}
h2 {
    font-style: italic;
    border-bottom: 3px solid #EF9C00;  /* Adds a line below h2 headings */
}

/* Adjust the styles for the unordered list */
ul {
    list-style-type: square;           /* Changes the bullets to squares */
}
```

Example: How to code basic selectors

basic_selector.html

```
<head>
    <meta charset="utf-8">
    <title>Basic Selectors</title>
    <link rel="stylesheet" href="basic_selectors.css">
</head>
<body>
    <h1 class="base_color">Student Materials</h1>
    <p>Here are the links for the downloads:</p>
    <ul id="links">
        <li><a href="exercises.html">Exercises</a></li>
        <li><a href="solutions.html">Solutions</a></li>
    </ul>
    <p id="copyright" class="base_color">Copyright 2022</p>
</body>
```

basic_selectors.css

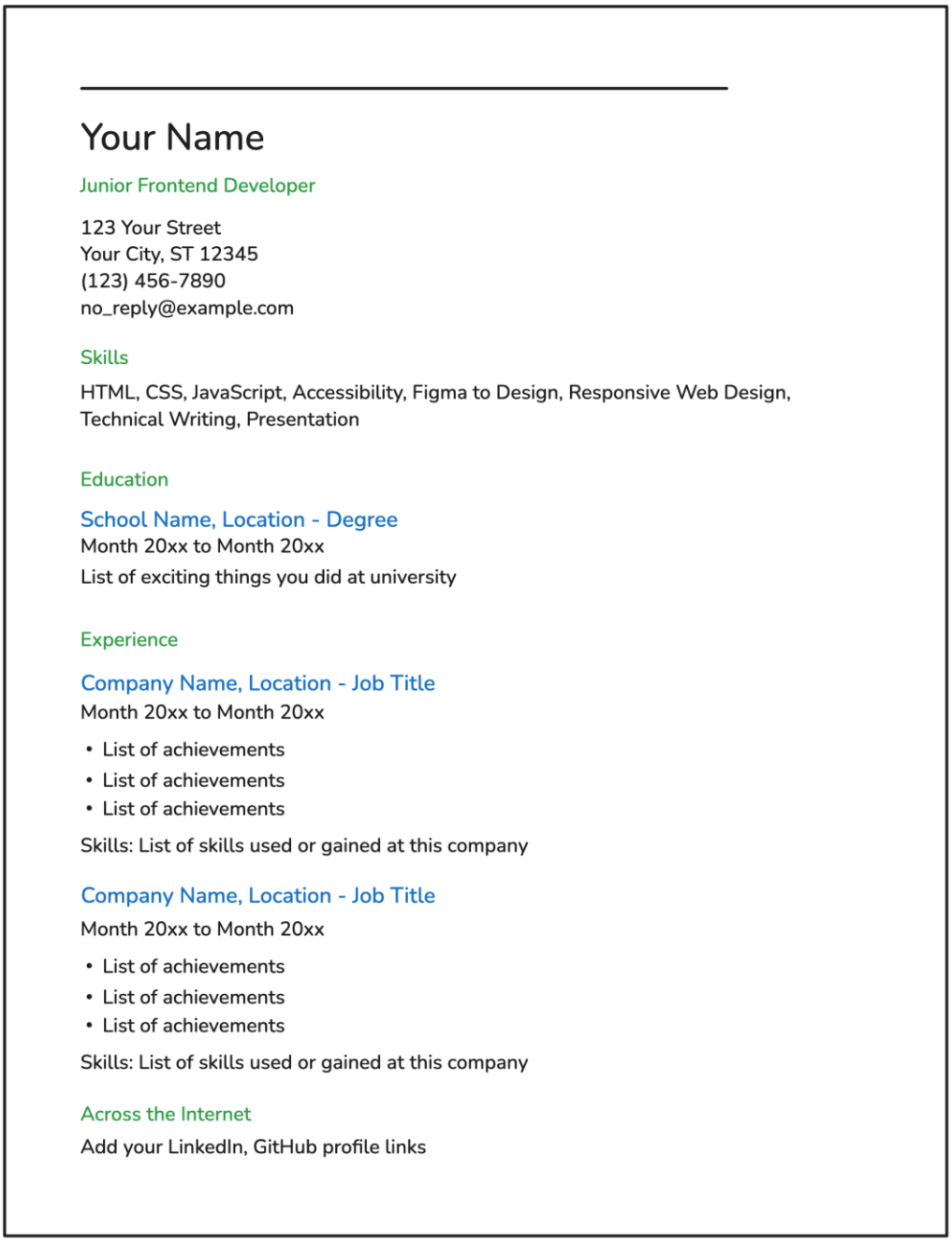
```
body {
    font-family: Arial, sans-serif;
    font-size: 100%;
    width: 300px;
    padding: 1em;
}

h1 {
    font-size: 180%;
}

/* ID */
#copyright {
    font-size: 75%;
    text-align: right;
}

/* Class */
.base_color {
    color: blue;
}
```

Part 2: In class practice

Exercise 1:

Design your personal website using HTML

Todo this project you may need to read: Welcome to Learn HTML! | web.dev
In this project, you are required to create a single-page CV (Curriculum Vitae) using only HTML. Your webpage should look like the following image:



Your Name

Junior Frontend Developer

123 Your Street
Your City, ST 12345
(123) 456-7890
no_reply@example.com

Skills

HTML, CSS, JavaScript, Accessibility, Figma to Design, Responsive Web Design, Technical Writing, Presentation

Education

School Name, Location - Degree
Month 20xx to Month 20xx
List of exciting things you did at university

Experience

Company Name, Location - Job Title
Month 20xx to Month 20xx

- List of achievements
- List of achievements
- List of achievements

Skills: List of skills used or gained at this company

Company Name, Location - Job Title
Month 20xx to Month 20xx

- List of achievements
- List of achievements
- List of achievements

Skills: List of skills used or gained at this company

Across the Internet

Add your LinkedIn, GitHub profile links

Key requirements for this project:

- **Semantic HTML**: Use appropriate HTML tags to structure your CV. Reading: Semantic HTML | web.dev
- **SEO Meta Tags**: Include essential meta tags for SEO. (optional)

- **Open Graph (OG) Tags**: Add OG tags for better social media sharing: https://web.dev/learn/html/metadata#open_graph
- **Favicon**: Add a favicon for your CV page.

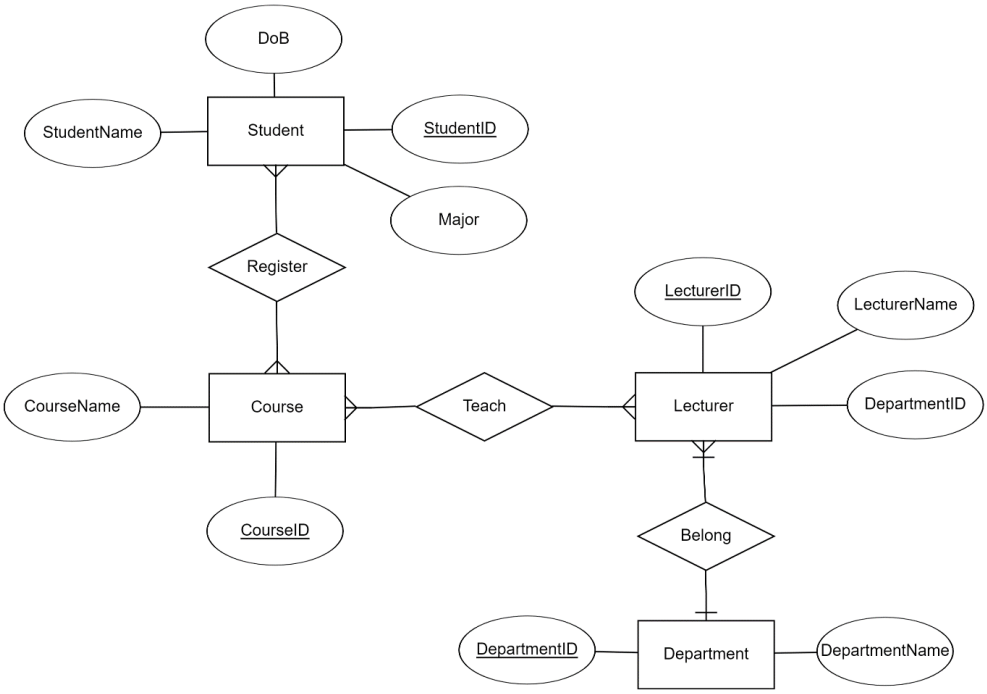
The structure of your CV should be easily understandable and ready for styling in a future project.

Submission Checklist:

- Semantically correct HTML structure.
- Single-page layout with sections for education, skills, and career history.
- SEO meta tags in the head section.
- OG tags for better social media sharing.
- A favicon linked in the head section.

Submit your GitHub link & source code to BB on time

Exercise 2:



Exercise 3: Check BB

How to submit your work.

- Create a database (with some random table and data) using MySQL and then export the data into one SQL file with format: **fullname_id.sql (Ex: ntnghia_ITITIU13047.sql)**
- Capture all your pages and place them in the pdf file.
- Compress all the HTML, CSS, PDF files, and SQL file with the format **fullname_id.zip** (Ex: ntnghia_ITITIU13047.zip) and submit to the Blackboard.
- Upload your work to GitHub and submit the Github link to BB

Part 3: Assignment-1 (2 weeks)

1) Project Objectives

- Build a **7–10 page personal website** with clear navigation.
- Correctly use **HTML5 semantics, iframe, media (images/video), links, full-featured form, and external CSS**.
- Ensure **usability (UX), responsiveness, accessibility, clean code**, and **deploy online** (GitHub Pages/Netlify).

---

2) Suggested Topics (choose one)

1. **Personal Portfolio**: About me, skills, projects, blog, contact.
2. **Club/Research Group**: Activities, events, gallery, membership form, contact.
3. **Coffee Shop/Small Business**: Menu, pricing, promotions, booking/order form, location (map iframe).
4. **Travel Site**: Destinations, itineraries, reviews, photo gallery, feedback form.
5. **Pharmacy/Science-themed site**: Introduce a topic, blog posts, search with datalist, booking form, infographic gallery.

---

3) Example Site Map (7–10 pages)

- /index.html (Home)
- /about.html (About)
- /projects.html or /services.html (Main content)
- /gallery.html (Gallery)
- /blog.html + /post-1.html (Blog list + sample post)
- /contact.html (Contact – **form**)
- /map.html (Map/video via **iframe**)
- /policy.html (Policies/Terms)

(Students choose 5–10 pages; must meet technical requirements below.)

---

4) Technical Requirements (for students to tick)

4.1 HTML & Structure

- **5–10 separate HTML files**.
- Use **semantic tags**: header, nav, main, section, article, footer.
- Consistent **navigation bar** across pages.
- Internal + external links.
- ≥5 images with alt; at least one figure/figcaption.
- At least one **iframe** (Google Maps, YouTube, etc.).
- Full **form** with multiple components (see 4.2).

4.2 Form Requirements

- Use **label** for all inputs; group with fieldset + legend.
- Include: text, email, tel, password, number, date, range, color, file, radio, checkbox, select (optgroup), textarea, datalist, hidden, submit/reset buttons.
- Use **required, pattern, min/max/step**.
- Show friendly error messages.
- (Optional bonus): progress, meter, output with small JS.

4.3 CSS & Styling

- External CSS file.
- Use **Flexbox/Grid**; responsive (at least 1 breakpoint ~768px).
- CSS variables for colors.
- Hover/focus states.
- Consistent color scheme & typography.

4.4 Accessibility

- Proper contrast, font ≥16px.
- Keyboard navigation OK, focus ring visible.
- Correct heading order, alt text, aria-label if needed.

4.5 Optimization & Quality

- Optimized images (WebP/JPG).
- Favicon, meta viewport, meta description, unique title tags.
- Clean folder structure.
- Pass **W3C Validator**.
- Credit image/icon sources.

4.6 Deployment

- Hosted on **GitHub Pages** or **Netlify**.
- Source code in GitHub repo.
- README with site map, features, credits.

5) Deliverables

1. Live URL (GitHub Pages/Netlify).
2. GitHub repo + zip file.
3. README file (site map, features, how to run, credits).
4. Wireframe/design sketch.
5. Short testing report.
6. **AI usage log** (if applicable).

---

7) Grading Rubric (100 points)

- Structure & number of pages: 10
- Navigation & links: 10
- Media & iframe: 10
- Full-featured form: 20
- CSS & responsive design: 20
- Accessibility: 10
- Code quality & optimization: 10
- Deployment & documentation: 10
- Creativity/design: 5
- Bonus features: +5 (not exceeding 100)

⚠️ **Fail condition**: Missing **form**, **iframe**, or **deployment** → max 60.

---

8) Student Self-Checklist

- 5–10 HTML pages.
- Consistent menu + working links.
- ≥5 images with alt; at least 1 iframe.
- Full form with all input types.
- External CSS + responsive layout.
- Validation + accessibility OK.
- Passed W3C validator.
- Deployment works (URL public).
- README file + credits.
- AI usage log included.

---

9) AI Usage Policy

Allowed: idea generation, grammar/style help, CSS/HTML snippets.
Not allowed: copying full template without understanding.
Required: submit **AI usage log** (tool used, prompts, how you modified).

---

11) Bonus (Optional)

- Dark/light mode.
- Print-friendly CSS.
- Theme switcher.
- Lazy-loading images.
- SEO meta tags.

---

12) Folder Structure (Sample)

```
/ (root)
  index.html
  about.html
  projects.html
  gallery.html
  blog.html
  post-1.html
  contact.html
  map.html
  assets/
    css/styles.css
    img/...
    js/main.js
```