Assignments › HW 1: Create a Dynamic HTML Resume

HW 1: Create a Dynamic HTML Resume

▼ Hide Assignment Information

Turnitin™

🏛 This assignment will be submitted to Turnitin™.

Instructions

Objective:

Build a professional, interactive resume web page using advanced HTML elements and features. This assignment will test your ability to structure content, utilize semantic elements, implement media, and add interactivity.

Instructions:

Submit   **Cancel**

advanced HTML elements.

- Create a visually attractive and functional resume that includes **dynamic and interactive features.**

2. **Requirements:**

   Your resume must include the following **mandatory sections and advanced features:**

1. Header

- Your full name and professional title (use `<header>` and `<h1>` tags effectively).
- **A responsive navigation menu** with links to each section of the resume (e.g., Work Experience, Skills).
- Contact information with clickable links for:
    - Email (`mailto:`).
    - Phone number (`tel:`).
    - Social media profiles (e.g., LinkedIn, GitHub).

2. Summary/About Me/Profile

- A **dynamic summary**:
    - Include a paragraph introducing yourself, with certain words styled using `<strong>`, `<em>`, and `<mark>`.
    - Add **tooltips** using the `<abbr>` tag to explain technical jargon or certifications.

---

3. Work Experience

- For each job:
    - Include job title, company name, duration, and responsibilities.
    - Format responsibilities as an **ordered list** with **interactive tooltips** (e.g., hover over a task for more details).
- Add an **expandable section** using `<details>` and `<summary>` tags to provide additional details about specific roles.

---

4. Education

- Use a **table** to display:
    - Degree, Institution, Graduation Date, GPA, and Achievements.
- Include **icons/images** for the institutions or certifications.

---

5. Skills

- Create a **progress bar** or **rating system** using `<progress>` or `<meter>` elements for each skill (e.g., HTML: 90%, JavaScript: 75%, C++:80%).
- Group technical skills into categories using `<fieldset>` and `<legend>`.

---

6. Projects

- Include at least **three projects**, with:
    - Name, description, technologies used (use `<ul>` or `<dl>`).
    - Links to live projects (using `<a>`).
    - A preview image or **embedded media** (e.g., video demonstration using `<iframe>`).

---

7. Certifications or Achievements

- Use a **definition list (`<dl>`)** to list certifications and achievements with short descriptions.
- Highlight one certification with a **badge-like design** using custom CSS.

---

8. Interests/Hobbies

- Add a **gallery of images** or icons representing your hobbies using the `<figure>` and `<figcaption>` tags.
- Include a **hover effect** on images that reveals more information about the hobby.

9. Footer

- Include:
    - A closing statement or call-to-action (e.g., "Let's Connect").
    - A **back-to-top button** using `<a>` with an `id` and `href`.
- Add copyright information and links to your social media profiles.

Advanced Features to Implement:

1. **Semantic HTML:**
   Use `<header>`, `<nav>`, `<main>`, `<section>`, `<aside>`, `<article>`, and `<footer>` appropriately.

2. **Interactive Elements:**

    - **Forms**: Add a simple contact form with:
        - Fields for name, email, message.
        - A submit button with a hover animation.
    - **Media Embeds**: Use `<audio>` or `<video>` to embed a media file related to your projects or hobbies.

3. **Responsive Design:**

    - Make the page **mobile-friendly** using a responsive layout with `<meta>` tags and flexible content.
    - Add **media queries** to ensure proper rendering on different devices.

4. **Accessibility:**

    - Add **ARIA roles** to enhance accessibility for screen readers.
    - Use **alt attributes** for all images.
    - Include **keyboard navigation** options (e.g., using `tabindex`).

5. **Custom Styling (Optional but Encouraged):**

    - Use the `<style>` tag to implement creative and visually appealing styles.
    - Add **CSS transitions or animations** to elements (e.g., hovering over links, button clicks).
    - Use **pseudo-classes** like `:hover` and `:focus`.

Submission:

1. Save your file as `YourName_resume.html`.
2. Include comments in your HTML code explaining how you implemented the advanced features.
3. Submit your file and/or host it on a free platform like GitHub Pages or CodePen and share the link.

Helpful Resources:

- W3Schools HTML Tutorial

Due on Jan 27, 2025 11:59 PM

Available until Jan 28, 2025 11:59 PM. **Access restricted after availability ends.**

Submit Assignment

Files to submit

(0) file(s) to submit

After uploading, you must click Submit to complete the submission.

Add a File   Record Audio   Record Video

Comments