

CSCI 1170 Assignment 3

Choose 2 pieces of code that can be improved by using JS. For each example, present (a) the code snippet, (b) current outcome, (c) desired outcome, and then (d) present a code snippet that would fix the problem and finally, (e) present the new outcome of the code. Maximum 500 words each example (not including code snippets), include as a PDF within your zip file.

Part 1: Code Review

Example 1: Adding JS to Projects Section to easily add more projects and experience in the future

(a)

Code Snippet:

```
<section id="projects">
  <h2>Projects</h2>
  <table class="projects-table">
    <tr>
      <th>Name</th>
      <th>Description</th>
    </tr>
    <tr>
      <td><strong>OddJobs Website</strong></td>
      <td>Helped build a website using React and node.js where users could find part-time jobs such as mowing the lawn, cleaning windows and various other jobs. </td>
    </tr>
    <tr>
      <td><strong>Preview Days Game</strong></td>
      <td>Currently designing an educational java platformer game for high school students using a framework called libGDX where we teach how variables work for future computer science students. </td>
    </tr>
    <tr>
      <td><strong> Smart Contract for NFT Project &nbsp;&nbsp;&nbsp;</strong></td>
      <td>Constructed an ERC 721 smart contract for an NFT project which would allow users to mint an NFT for a specific price from the projects website. </td>
    </tr>
  </table>
</section>
```

(b) Current outcome: Shows a list of projects the person has worked by hardcoding it in HTML

Projects

Name	Description
OddJobs Website	Helped build a website using React and node.js where users could find part-time jobs such as mowing the lawn, cleaning windows and various other jobs.
Preview Days Game	Currently designing an educational java platformer game for high school students using a framework called libGDX where we teach how variables work for future computer science students.
Smart Contract for NFT Project	Constructed an ERC 721 smart contract for an NFT project which would allow users to mint an NFT for a specific price from the projects website.

(c)

Desired outcome: Have the table dynamically generated from a JavaScript file, so that the user can easily add, edit, or delete projects without having to manually edit the HTML file. This will make it a lot easier for the person to edit their portfolio website in the future when they get more experience.

(d)

HTML Code Snippet:

```
<section id="projects">
  <h2>Projects</h2>
  <table id="projects-table" class="projects-table">
    <thead>
      <tr>
        <th>Name</th>
        <th>Description</th>
      </tr>
    </thead>
    <tbody></tbody>
  </table>
</section>
```

JS Code Snippet:

```
const projects = [
  {
    name: "OddJobs Website",
    description: "Helped build a website using React and node.js where users
could find part-time jobs such as mowing the lawn, cleaning windows and various
other jobs."
  },
  {
    name: "Preview Days Game",
    description: "Currently designing an educational java platformer game for
high school students using a framework called libGDX where we teach how variables
work for future computer science students."
  },
  {
    name: "Smart Contract for NFT Project",
    description: "Constructed an ERC 721 smart contract for an NFT project
which would allow users to mint an NFT for a specific price from the projects
website."
  },
  {
    name: "Designed an E-commerce website for a client",
    description: "Helped build and design an E-Commerce website that
generated over 20,000 USD in sales"
  }
];

const projectsTable = document.querySelector("#projects-table tbody");

projects.forEach((project) => {
  const row = projectsTable.insertRow();
  const nameCell = row.insertCell();
  const descriptionCell = row.insertCell();

  nameCell.innerText = project.name;
  descriptionCell.innerText = project.description;
});
```

(e)

New outcome: The projects table will be dynamically generated from the JavaScript file. This makes it easier for the user to add, edit or delete projects by simply modifying the array of projects in the JavaScript file which will make it easier for them to edit in the future.

mysql

Projects

Name	Description
OddJobs Website	Helped build a website using React and node.js where users could find part-time jobs such as mowing the lawn, cleaning windows and various other jobs.
Preview Days Game	Currently designing an educational java platformer game for high school students using a framework called libGDX where we teach how variables work for future computer science students.
Smart Contract for NFT Project	Constructed an ERC 721 smart contract for an NFT project which would allow users to mint an NFT for a specific price from the projects website.
Designed an E-commerce website for a client	Helped build and design an E-Commerce website that generated over 20,000 USD in sales

Example 2: Adding JS to Skills Section to easily add more skills in the future

(a)

Code Snippet:

```
<section id="skills">
  <h2>Skills</h2>
  <ul>
    <li>JavaScript</li><br>
    <li>React</li><br>
    <li>Node.js</li><br>
    <li>Java</li><br>
    <li>Python</li><br>
    <li>MySQL</li>
  </ul>
</section>
```

(b) Current outcome: Shows a list of skills the person has by hardcoding it in HTML

(c)

Desired outcome: Have the skills dynamically generated from a JavaScript file, so that the user can easily add, edit, or delete skills without having to manually edit the HTML file. This will make it a lot easier for the person to edit their portfolio website in the future when they learn more.

(d)

HTML Code Snippet:

```
<section id="skills">
  <h2>Skills</h2>
  <ul id="skills-list"></ul>
</section>
```

JS Code Snippet:

```
const skills = [
  "JavaScript",
  "React",
  "Node.js",
  "Java",
  "Python",
  "MySQL",
  "Sony Vegas"
];

const skillsList = document.getElementById("skills-list");

skills.forEach((skill) => {
  const li = document.createElement("li");
  li.textContent = skill;
  skillsList.appendChild(li);
});
```

(e)

New Outcome: The skills list will be dynamically generated from the JavaScript file. This makes it easier for the user to add, edit or delete skills by simply modifying the array of skills in the JavaScript file which will make it easier for them to edit in the future.

Skills

JavaScript

React

Node.js

Java

Python

MySQL

Sony Vegas