**AI Prompting**

The first step I take in every web development project is crafting a simple AI prompt.

“Create a HTML skeleton.”

Response:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Simple HTML Skeleton</title>

</head>

<body>

<header>

<h1>Welcome to My Website</h1>

</header>

<main>

<section>

<h2>About Us</h2>

<p>This is a simple HTML skeleton example.</p>

</section>

</main>

<footer>

<p>&copy; 2024 Your Website</p>

</footer>

</body>

</html>

You will see this simple page if you run the code without any additions:

A white background with black text

Description automatically generated

Although this assignment only requires the construction of a single component, I aim to give it a sense of completeness.

I have chosen to explore creating an image carousel in collaboration with ChatGPT.

AI Prompt: Create a JavaScript image carousel with autoplay and navigation features. Please include a pause on hovering. (I find myself often including simple human politeness when working in concert with AI.)

Response:

HTML:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Image Carousel</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="carousel-container">

<div class="carousel">

<img src="https://via.placeholder.com/600x400/FF5733/FFFFFF?text=Image+1" alt="Image 1">

<img src="https://via.placeholder.com/600x400/33FF57/FFFFFF?text=Image+2" alt="Image 2">

<img src="https://via.placeholder.com/600x400/3357FF/FFFFFF?text=Image+3" alt="Image 3">

<img src="https://via.placeholder.com/600x400/FF33F6/FFFFFF?text=Image+4" alt="Image 4">

</div>

<!-- Navigation Buttons -->

<button class="prev" onclick="moveSlide(-1)">&#10094;</button>

<button class="next" onclick="moveSlide(1)">&#10095;</button>

</div>

<script src="script.js"></script>

</body>

</html>

CSS:

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

}

.carousel-container {

position: relative;

width: 600px;

margin: 50px auto;

overflow: hidden;

}

.carousel {

display: flex;

transition: transform 0.5s ease-in-out;

}

.carousel img {

width: 100%;

height: auto;

}

button {

position: absolute;

top: 50%;

background-color: rgba(0, 0, 0, 0.5);

color: white;

border: none;

font-size: 30px;

padding: 10px;

cursor: pointer;

transform: translateY(-50%);

}

button:hover {

background-color: rgba(0, 0, 0, 0.8);

}

.prev {

left: 10px;

}

.next {

right: 10px;

}

JavaScript:

let index = 0;

const slides = document.querySelectorAll('.carousel img');

const totalSlides = slides.length;

let slideInterval;

function showSlide(i) {

if (i >= totalSlides) index = 0;

if (i < 0) index = totalSlides - 1;

const carousel = document.querySelector('.carousel');

carousel.style.transform = `translateX(${-index \* 600}px)`; // Adjusts the carousel position

}

function moveSlide(n) {

showSlide(index += n);

}

// Autoplay function

function startAutoplay() {

slideInterval = setInterval(() => {

showSlide(index += 1);

}, 3000); // Change slide every 3 seconds

}

// Pause autoplay on hover

const carouselContainer = document.querySelector('.carousel-container');

carouselContainer.addEventListener('mouseover', () => {

clearInterval(slideInterval); // Stops the autoplay when mouse hovers

});

carouselContainer.addEventListener('mouseleave', () => {

startAutoplay(); // Resumes autoplay when mouse leaves

});

// Start the autoplay when the page loads

startAutoplay();

I erased this:

A screenshot of a computer code

Description automatically generated

And added this HTML to that section as so: (I also added CSS and JavaScript file links as well)

A screen shot of a computer code

Description automatically generated

I then created a CSS File and a JavaScript file.

CSS:

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

}

.carousel-container {

position: relative;

width: 600px;

margin: 50px auto;

overflow: hidden;

}

.carousel {

display: flex;

transition: transform 0.5s ease-in-out;

}

.carousel img {

width: 100%;

height: auto;

}

button {

position: absolute;

top: 50%;

background-color: rgba(0, 0, 0, 0.5);

color: white;

border: none;

font-size: 30px;

padding: 10px;

cursor: pointer;

transform: translateY(-50%);

}

button:hover {

background-color: rgba(0, 0, 0, 0.8);

}

.prev {

left: 10px;

}

.next {

right: 10px;

}

JavaScript:

let index = 0;

const slides = document.querySelectorAll('.carousel img');

const totalSlides = slides.length;

let slideInterval;

function showSlide(i) {

if (i >= totalSlides) index = 0;

if (i < 0) index = totalSlides - 1;

const carousel = document.querySelector('.carousel');

carousel.style.transform = `translateX(${-index \* 600}px)`; // Adjusts the carousel position

}

function moveSlide(n) {

showSlide(index += n);

}

// Autoplay function

function startAutoplay() {

slideInterval = setInterval(() => {

showSlide(index += 1);

}, 3000); // Change slide every 3 seconds

}

// Pause autoplay on hover

const carouselContainer = document.querySelector('.carousel-container');

carouselContainer.addEventListener('mouseover', () => {

clearInterval(slideInterval); // Stops the autoplay when mouse hovers

});

carouselContainer.addEventListener('mouseleave', () => {

startAutoplay(); // Resumes autoplay when mouse leaves

});

// Start the autoplay when the page loads

startAutoplay();

This is how the webpage appeared when the code was run:

A screenshot of a computer

Description automatically generated

Of course, I had to include actual images to fully test it. I turned to Microsoft Copilot for some AI-generated images.

AI Image Prompts:

AI Prompt 1: Pictures of AI dancing:

A group of robots in a room

Description automatically generated

AI Prompt 2 (I wanted four images to match the number of images the generated code utilized, but alas I must do it the long way, because now Microsoft Copilot only provides one image at a time, where it used to provide four.) Can I have another one:

A robot standing on a circle in front of a crowd of people

Description automatically generated

AI Prompt 3: Another please:

A robot running in a futuristic setting

Description automatically generated

Ai Prompt 4: One more and make it the best:

A robot dancing in front of a crowd of people

Description automatically generated

Finally, I have all my images. I added each one to my visual code studio “AI in code generation” folder and replaced the image source paths with my artists images.

Lastly, I prompted the AI (ChatGPT model) with “Using the above code how do I center everything and make it mobile friendly?”

Response: It deleted the header and main sections from the HTML file, and in the CSS it adjusted the specifications for the body, and .carousel container, added H1 and Footer CSS specifications along with a media query which will adjust the webpage to handle smaller screens.

Summary: It functions beautifully without any adjustments needed on my part for the current deliverable. However, if I were to use it in a webpage as an advertisement, or a step-by-step how to, I would need to adjust it to fit where I would like it too in the flow of whichever design I would be utilizing to make a webpage.

To Reflect:

What was helpful about using AI for this task?

The website itself was done in 10 minutes, while providing my process and creating a walkthrough for it took an hour and a half. I am in rapture over how quickly AI’s get through tasks. The productivity increase alone will be worth billions.

Did the AI provided code that was immediately useable, or did it require significant changes?

It was fired up and ready to go on output, excepting the pictures and then the adjustments to center it and make it mobile friendly, which the AI did as well with just a few more prompts, all within ten minutes.

What limitations did you notice in AI’s generated code?

If you cannot read the code, you will not be able to understand the AI explanations of the code and will have significant issues with implementing it.

To conclude this Hermione sized document, I have enjoyed this lesson out of all my classes this semester right behind Devtools. AI is an incredible invention and makes learning and memorization a breeze. It not only gives you a code that will work (if you know what your doing), it tells you why it will work and reinforces coding and computer science terminology.