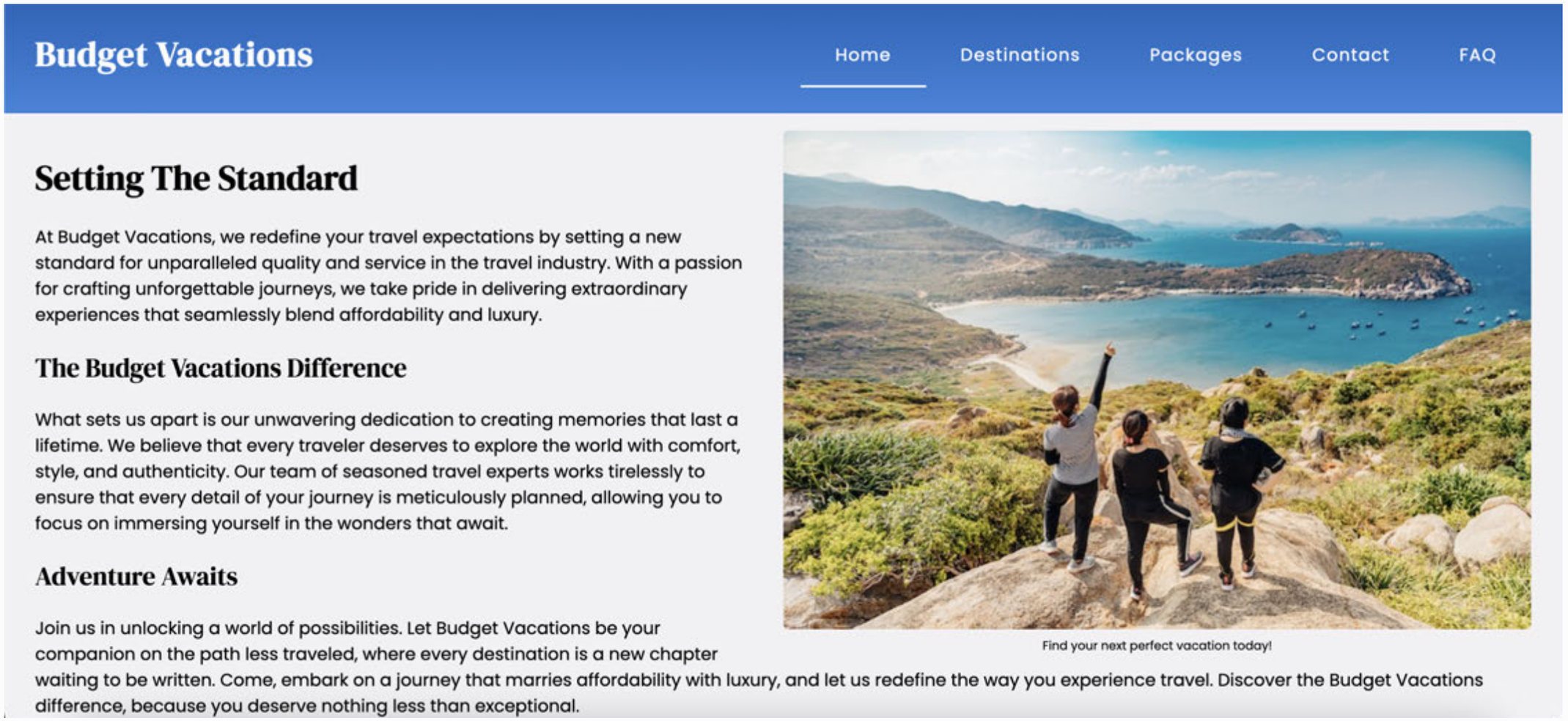
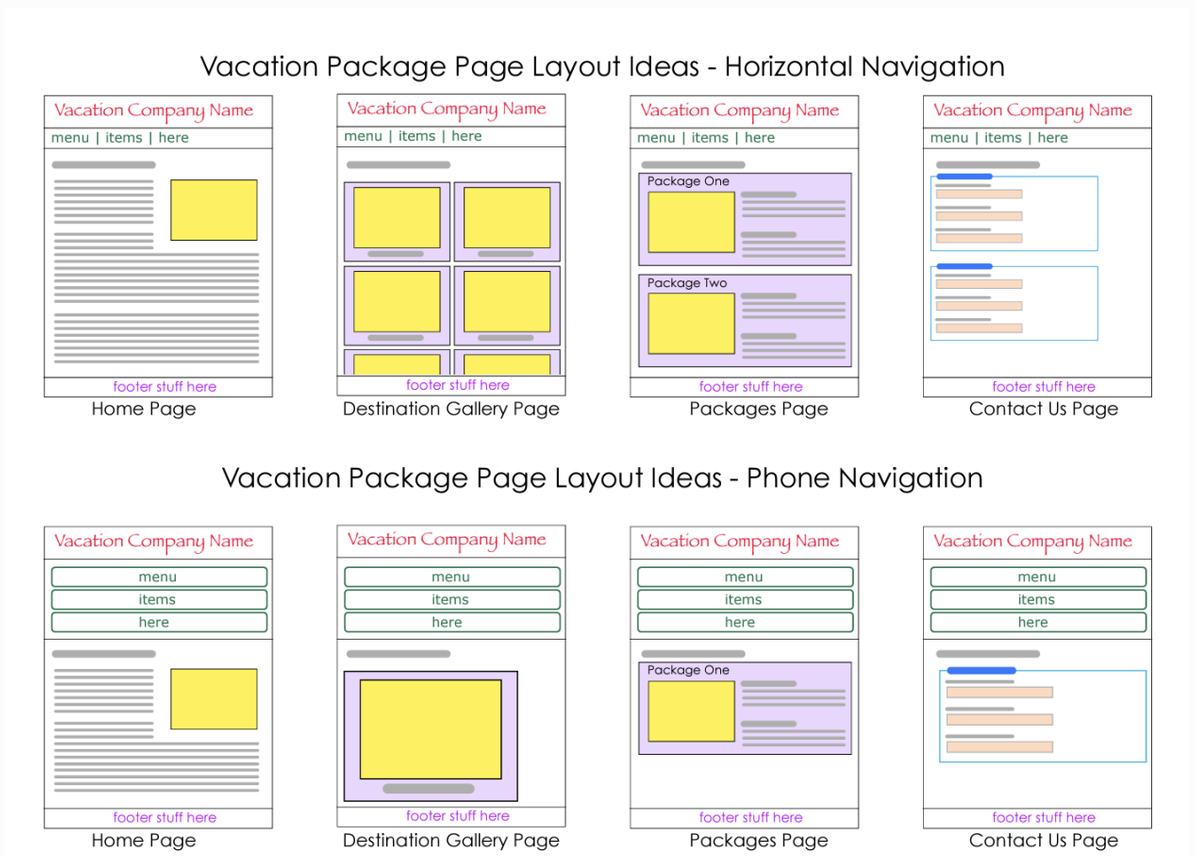
Coding a Vacation Booking Site



As the capstone project for my Web Essentials class, I was given the option to hand code a vacation booking website. As the final project for the class, I would not have the follow along videos I had for past assignments. Coding this site was optional but I saw it as a opportunity to push myself and grow my skills.

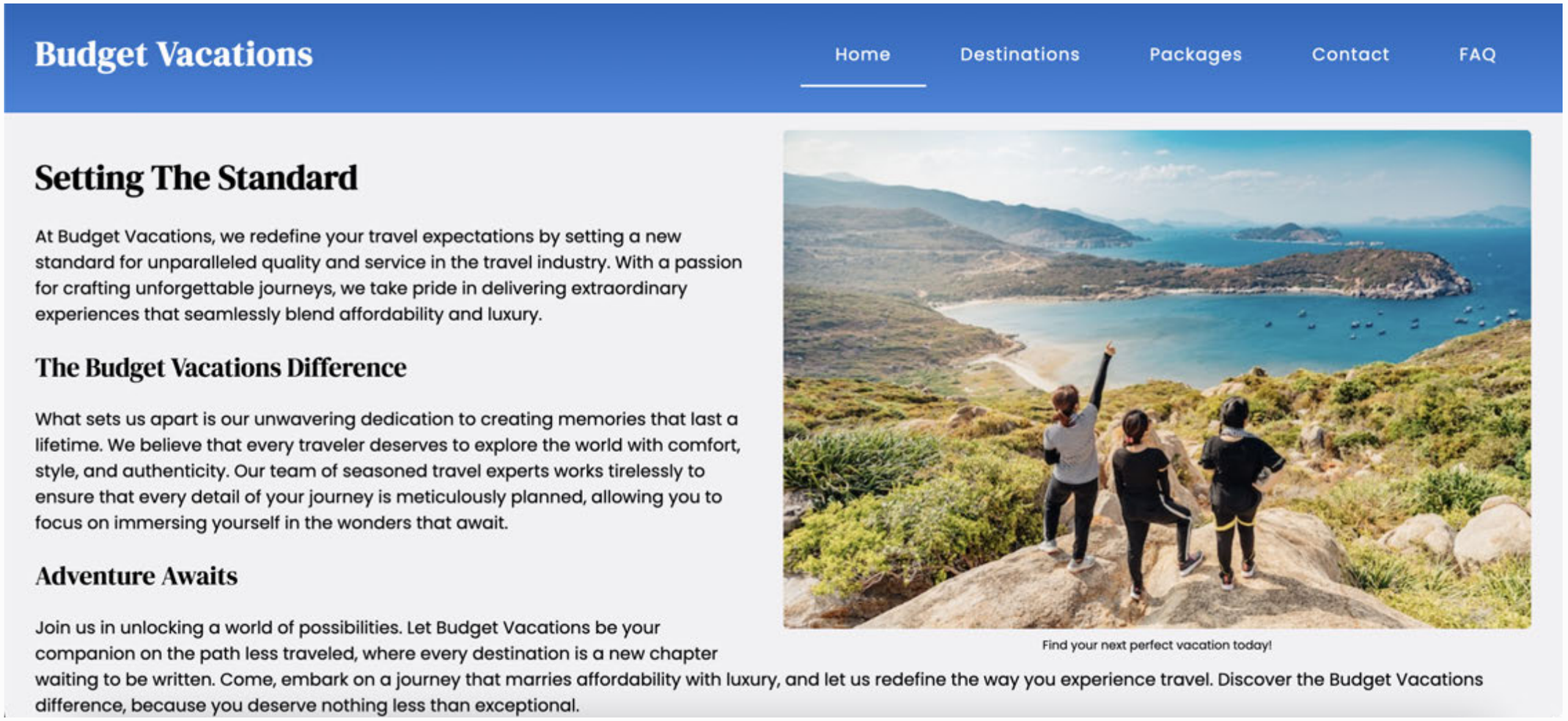
Getting Started

This project began with guidelines provided by the instructor on what pages, elements, and structure the website required. Outside of that I was free to design and code the site how I thought was best. This level of freedom made me a bit nervous getting started but focusing on one page at a time allowed me to get going slowly and with more confidence.

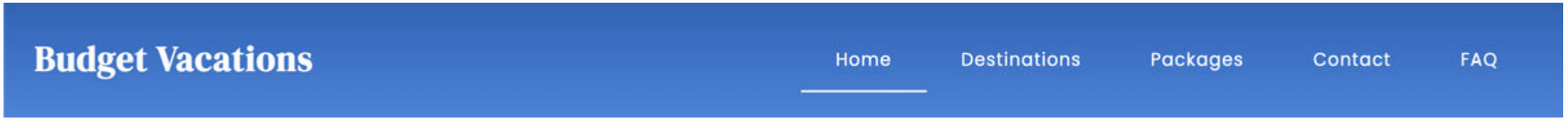


Wireframes provided by instructor laying out required pages and structure

The first page I started with was the Home page. I figured it would be best to lay the foundation by choosing a color palette and font and by coding the navigation bar, header, and footer first.



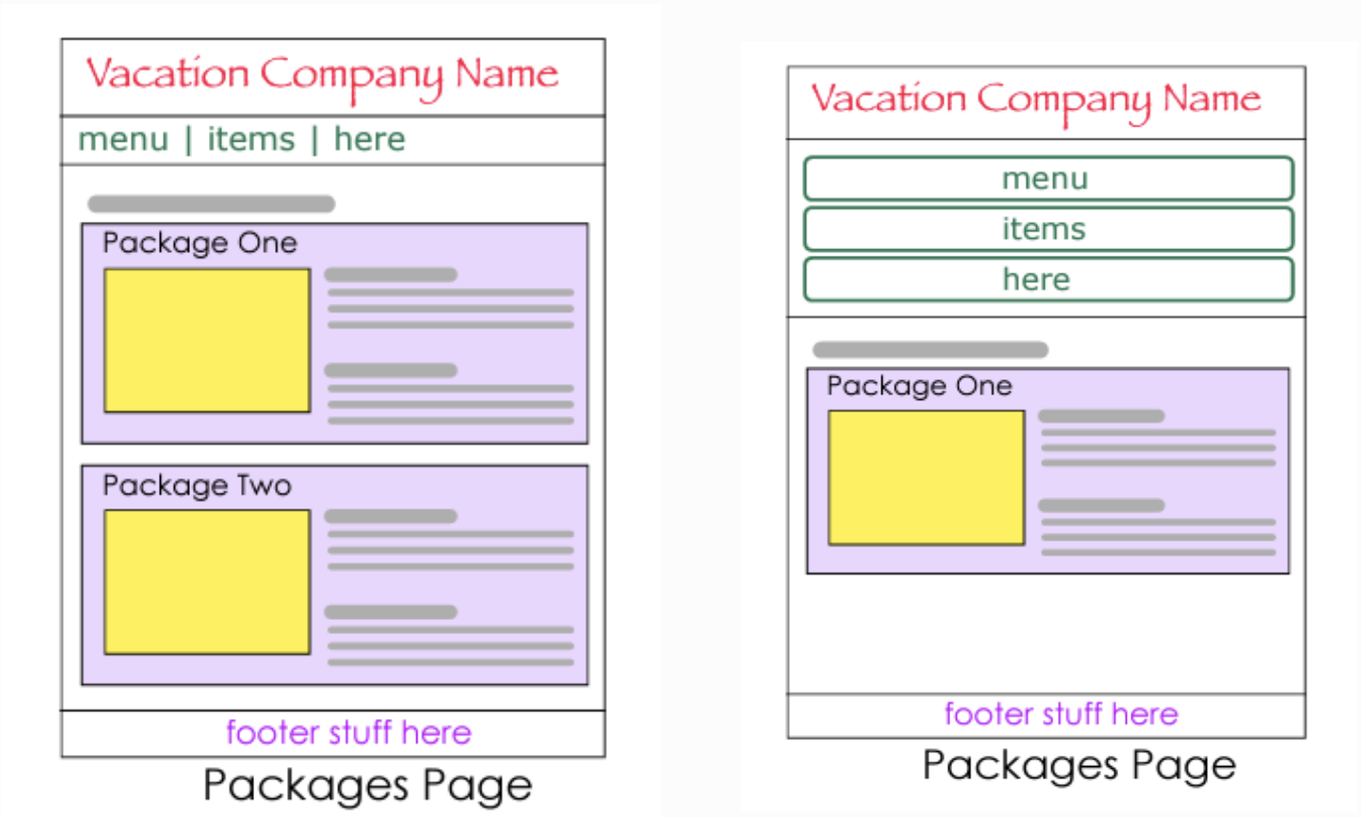
Setting up the basic foundation of the site helped me take on the project piece by piece and gave me the confidence to experiment with the layout. The wireframes communicated that the nav bar could be placed under the header but I decided to combine the two in one bar because it was always something I saw in other sites but never applied to my own in class.



For the mobile version of the nav bar I stuck closer to the wireframes. I did decide to go for a more simplified style in terms of the wayfinding and structure of the buttons.

Complications

With the foundation of the site set up I was able to get the Home page and Destinations page completed. I really began to struggle on the Vacation Packages page. The combination of an image and text box really threw me off since I hadn't had much practice with it. Along with that, making it responsive also added another layer of difficulty.



Wireframes for Packages page showing desktop and mobile designs.

```
<main>
    <h1>Vacation Packages</h1>
    <div class="package">
        <figure>
            <img src="images/cancunHotel.webp" alt="Aerial view of JW Marriott Cancun Resort">
        </figure>

        <div class="details">
            <h2>Beach Paradise Package - Cancun, Mexico</h2>
            <h3>Starting at $2,000</h3>
            <p>Explore the beautiful beaches of Cancun with this all-inclusive package. Lounge at the JW Marriott Cancun Resort and Spa, explore
            Cancun with an included rental car, and enjoy the local restaurant scene with $100 in daily credits.</p>
            <p>Includes:</p>
            <uL>
                <li>Two Bedroom Suite at the <a
                        href="https://www.marriott.com/en-us/hotels/cunjw-jw-marriott-cancun-resort-and-spa/overview/"
                        target="_blank">JW Marriott Cancun Resort and Spa</a></li>
                <li>Rental Car from <a href="https://www.enterprise.com/en/home.html" target="_blank">Enterprise</a>
                </li>
                <li>$100 Daily Restaurant Credit</li>
            </uL>

        </div>
    </div>
```

Above is how I ended up coding the HTML for a “package”. I wanted the image to be on the left side of the text card so I made the div a flexbox. The issue was that the text laying out the package details was too long so the text box kept wrapping and ending up below the image. However, if I turned off flex-wrap it wouldn’t be responsive in the way I wanted, which was to have the image stay as big as possible by wrapping to the top of the box in the mobile version.

In order to fix the issue I tried various ways of rearranging the HTML and even trying CSS grid instead of flexbox, which didn't work how I wanted. After some frustration I took a break and came back to it with fresh eyes and asked myself what the issue was in simple terms. Ultimately, the boxes were behaving the same way in the desktop and mobile version when I needed them to do different things. So that got me thinking how can I make separate rules for each version? Media queries.

```
183  /* small screen */
184  @media only screen and (max-width: 800px){
185      .package{
186          display: flex;
187          flex-wrap: wrap;
188      }
189      .package figure{
190          margin: 0;
191          width: 100%;
192      }
193  }
194
195  /* large screen */
196  @media only screen and (min-width: 801px){
197      .package figure{
198          margin: .5rem .5rem 0 0;
199          max-width: 579px;
200          width: 100%;
201      }
202  }
```

CSS code showing media queries I set up for each screen size.

Desktop Package

Vacation Packages

Beach Paradise Package - Cancun, Mexico

Starting at $2,000

Explore the beautiful beaches of Cancun with this all-inclusive package. Lounge at the JW Marriott Cancun Resort and Spa, explore Cancun with an included rental car, and enjoy the local restaurant scene with $100 in daily credits.

Includes:

- Two Bedroom Suite at the JW Marriott Cancun Resort and Spa
- Rental Car from Enterprise
- $100 Daily Restaurant Credit

Media queries allowed me to turn flex-wrap off on the larger desktop screen allowing the image to sit side by side with the package card.

Mobile Package

For the mobile version, having flex-wrap enabled allowed the image to have more space to itself and remain larger.

Results

While I faced plenty of issues in this project, the package problem was by far the hardest. Looking back the solution seems obvious but in the moment after coding for hours, it seemed like nothing I tried worked. Working on this helped me grow my troubleshooting and research skills as well as the value of knowing when to take a break.

Along with that I also increased my confidence when it comes to coding without a step by step guide. Going into this project my biggest worry was going to bold with my designs and running into an issue building it. I think now that I pushed myself to take this on and encountered that very issue helped me realize that I am growing as a developer and actually am capable of a lot more than I realize.