Step 1: Create the Basic HTML Structure

We will use semantic elements (<header>, <nav>, <main>, <section>, <footer>) for accessibility and readability.

1. **Create a folder named "Ch7" on your Desktop.**
2. **Open your code editor and create a new file and name it "sushi.html" .**
3. **Create a media query.**
4. **See the code below to create this page: (adjust the spacing between code blocks as you wish)**

```html
<!—Put your name here and the date you made the file-->
<!-- Ch. 7 Floats hw-->
<!DOCTYPE html>

<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sushi Delight</title>
    <link rel="stylesheet" href="styles.css"> <!-- External CSS -->
</head>
<body>
    <div class="wrapper">
        <header>
            Sushi Delight
        </header>
        <nav>
            <a href="sushi.html">Home</a>
            <a href="menu.html">Menu</a>
        </nav>
        <h2>Welcome to Sushi Delight!</h2>
        <main>
            <section>
            <p>Enjoy the finest sushi prepared by expert chefs with fresh
ingredients.</p>
            <img src="Sushi-Platters.webp" alt="Sushi platter">
            </section>
             <section>
              <p>Try our delicious rolls and sashimi.</p>
              <img src="rolls.webp" alt="Sushi rolls">
            </section>
         </main>
        <footer>
            &copy; 2025 Sushi Delight | Made with love
        </footer>
    </div>
</body>
</html>
```

5. Save the file in the Ch.7 folder

Step 2: Apply CSS Styling

6. Create a **styles.css** file and define styles for each section. Feel free to change color choices or add additional styles to fit your style.

Selector	CSS Explanation	Code
*	Resets the browser with margins and padding	margin: 0; padding:0; box-sizing: border-box;
.wrapper	Centers the content, sets max width, and adds padding/shadow.	width: 95%; margin: 0 auto; background: white; padding: 10px; border-radius: 5px; box-shadow: 0px 0px 10px #0000001a;
body	Sets the overall **background color**, removes margin/padding, and applies a font.	background: #d92323; color: white; text-align: center; padding: 20px; font-size: 1.5em;
header	Styles the top section with a **red background**, white text, and large font.	background: #d92323; color: white; text-align: center; padding: 20px; font-size: 1.5em;
nav	Adds a yellow background and centers navigation links.	background: #ffcc00; padding: 10px; text-align: center;

Selector	CSS Explanation	Code
nav a	Makes links stack naturally on smaller screens.	display: inline; /* Forces stacking on smaller screens */ text-decoration: none; color: black; padding: 5px; font-weight: bold;
nav a :visited	Colors nav once the page is visited.	text-decoration: underline; color:darkgreen;
nav a:hover	Colors the nav when the mouse is hovered on text.	background-color: #fff; color:#d92323;
section	Floats left (50% width), styles the sidebar.	float: left; width: 50%; padding: 10px;
main:after	Clears the floats and returns the page to normal flow.	content: ""; display: table; clear: both;
img	Makes images responsive to their containers.	height: 50%; max-width: 70%; opacity:.7; object-fit: cover;
img:hover	Makes images responsive and change the images on hover	border: 1px solid #ddc10f; opacity: 1; transform: scale(1.2); transition: transform 0.5s ease-in-out; /* Smooth scaling effect */
footer	Clears floats, applies a **red background**, and centers the white text.	clear: both; /* Fix float issues */ background: #d92323; color: white; text-align: center;

Step 3: Create a media query

A media query will add CSS depending on the size of the screen. This will stack the nav and images on a small screen like a phone.

```css
/* Mobile Styles: 768px and smaller */
@media (max-width: 768px) {
  section {
    width: 100%;
float: none;
    padding: 10px 0;
  }

  .wrapper {
    width: 100%;
    padding: 10px;
  }

  header {
    font-size: 1.2em;
    padding: 15px;
  }

  nav {
    padding: 8px;
  }

  nav a {
    display: block;
    margin: 5px 0;
    font-size: 1em;
  }

  img {
    max-width: 100%;
    height: auto;
  }
}
```
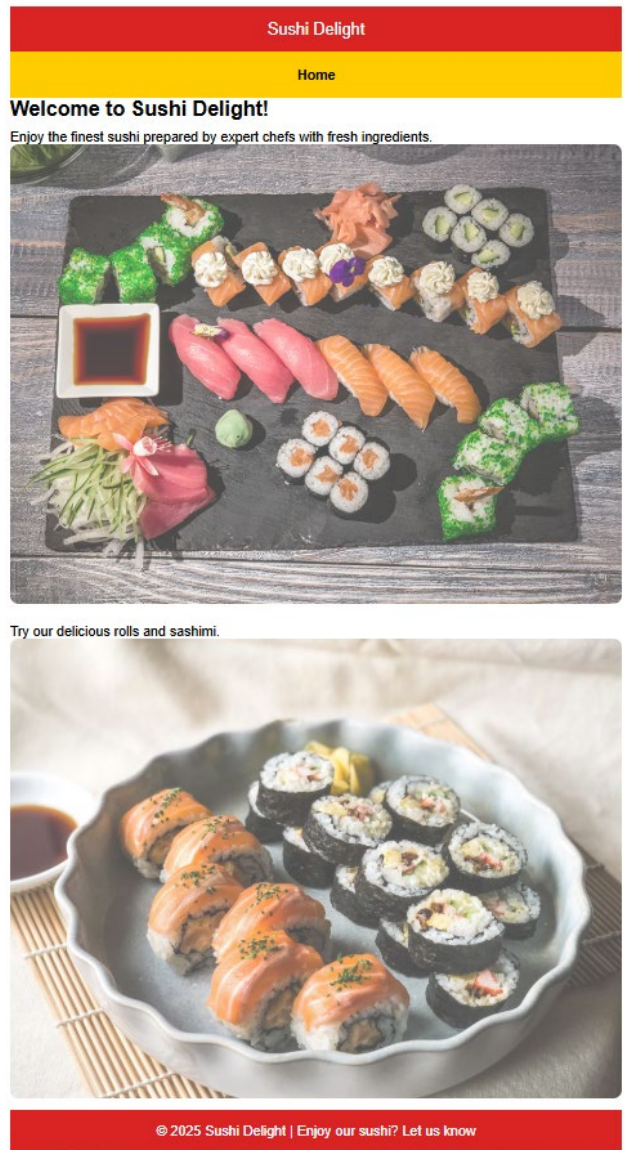
The page will look like this on a phone:

7. Save your page s sushi.html in your Ch.7 folder.
8. Preview the html and CSS pages to make sure everything looks correct.

Step 4: Submitting your work

9. Upload your Ch7 folder with your sushi.html file, styles.css file, and the two images to your GitHub account.
10. Open a browser and locate your sushi.html page by using your GitHub website.
    a. The link looks something like: webakcat.github.io/Ch7/sushi.html.
    b. GitHub is CASE SENSITIVE.
11. Submit your link to the Canvas drop box.

YOU DID IT!!