INTERNSHIP REPORT

WEEK 3 DAY 2

Submitted to:	**Ali Hyder**	Submission Date:	**9th July, 2025**
Internship Domain:	**Front Development**	Internship Name:	**ProSensia**
Student Name:	**Yasal Qamar**	Roll No.	**S25031**

Responsive Design with Media Queries

Objective

To understand and implement **responsive design principles** using **CSS media queries** to optimize web layouts for multiple screen **sizes (mobile, tablet, and desktop).** The goal was to build a multi-layout project that demonstrates how web content should adapt based on the user's device or screen resolution.

Introduction:

Today I focused on mastering responsive web design through the use of CSS media queries. The goal was to design web layouts that adapt seamlessly to different screen sizes including mobile (≤480px), tablet (481px–1024px), and desktop (≥1025px).

To accomplish this, I built four separate HTML pages:

- **index.html (Landing Page)**

Key Learning:

- U**nderstanding Media Queries**
  Learned how to use @media rules to apply different CSS styles based on the device's screen width, enabling a responsive and adaptive layout.
- **Mobile-First Approach**
  Designed layouts to start with the smallest screen (≤480px) and progressively enhanced them for larger devices (tablet and desktop), ensuring optimal performance and usability on mobile.
- **Device-Specific Breakpoints**
  Applied breakpoints using:
  - @media (max-width: 480px) for mobile
  - @media (min-width: 481px) and (max-width: 768px) for tablets

- @media (min-width: 769px) for desktops

- **Typography Scaling**
  Adjusted font sizes dynamically for headings and paragraphs to maintain readability across devices.
- **Flexible Layouts with CSS**
  Created layouts that adapt content structure (e.g., text, images, buttons) without breaking, using techniques like Flexbox and percentage-based widths.
- **Visual Testing with DevTools**
  Used browser Developer Tools (like Chrome's responsive mode) to simulate and test mobile, tablet, and desktop views without needing physical devices.
- **Improved User Experience**
  Ensured a consistent and user-friendly interface regardless of screen size, helping build accessible, real-world responsive web pages.

Coding:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Responsive Design Example</title>
  <style>
    * {
      box-sizing: border-box;
      margin: 0;
      padding: 0;
    }

    body {
      font-family: Arial, sans-serif;
      background-color: #f1f1f1;
      color: #333;
    }

    header {
      background: #0077b6;
      color: white;
      padding: 40px 20px;
      text-align: center;
    }
```

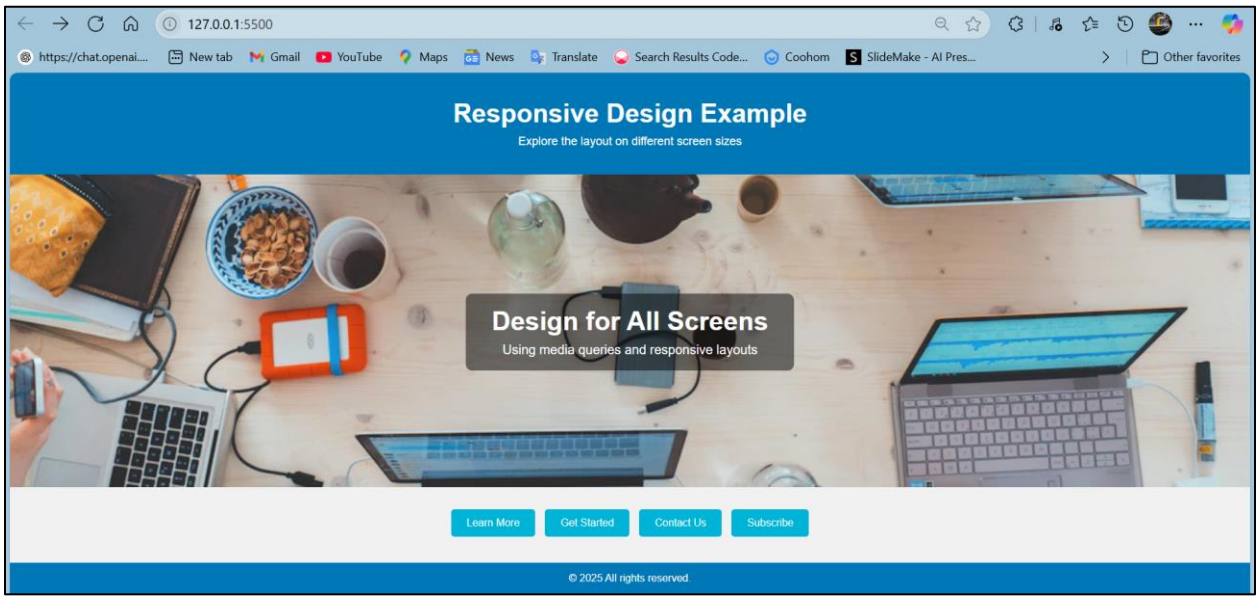
```css
header h1 {
  font-size: 40px;
}

header p {
  font-size: 18px;
  margin-top: 10px;
}

.hero {
  position: relative;
  width: 100%;
  height: auto;
  overflow: hidden;
}

.hero img {
  width: 100%;
  height: auto;
  max-height: 480px;
  object-fit: cover;
}

.hero-text {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  background-color: rgba(0, 0, 0, 0.5);
  color: white;
  padding: 20px 40px;
  border-radius: 10px;
  text-align: center;
}

.hero-text h2 {
  font-size: 36px;
  margin-bottom: 10px;
}

.hero-text p {
  font-size: 18px;
}

.buttons {
```

```
    font-size: medium;
    display: flex;
    justify-content: center;
    margin: 30px 0;
    gap: 15px;
    flex-wrap: wrap;
  }

  .buttons button {
    background-color: #00b4d8;
    border: none;
    color: white;
    padding: 12px 24px;
    font-size: 16px;
    cursor: pointer;
    border-radius: 5px;
    transition: background 0.3s;
  }

  .buttons button:hover {
    background-color: #0077b6;
  }

  footer {
    background: #0077b6;
    color: white;
    text-align: center;
    padding: 15px 10px;
    margin-top: 40px;
  }

  /* Mobile View Hidden Simulation */
  @media (max-width: 480px) {
    body {
      display: block;
    }
  }

  /* Tablet View */
  @media (min-width: 481px) and (max-width: 768px) {
    .hero-text h2 {
      font-size: 28px;
    }
    .hero-text p {
      font-size: 16px;
```

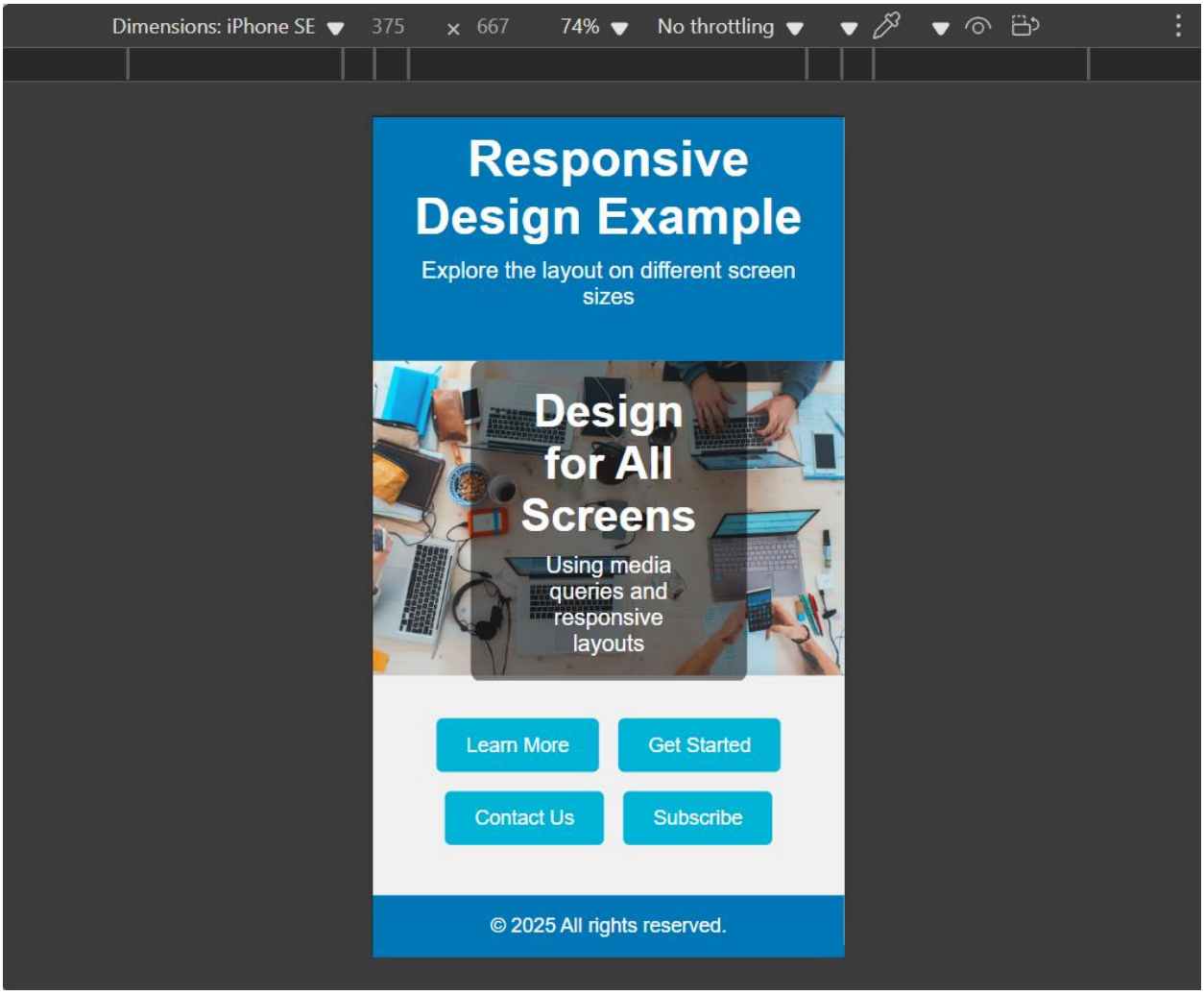
```
      }
    }

    /* Desktop View */
    @media (min-width: 769px) {
      .hero-text h2 {
        font-size: 40px;
      }
      .hero-text p {
        font-size: 20px;
      }
    }
  </style>
</head>
<body>

  <header>
    <h1>Responsive Design Example</h1>
    <p>Explore the layout on different screen sizes</p>
  </header>

  <section class="hero">
    <img src="https://images.unsplash.com/photo-1519389950473-
47ba0277781c?q=80&w=1170&auto=format&fit=crop&ixlib=rb-
4.1.0&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D" alt="Landing
Image">
    <div class="hero-text">
      <h2>Design for All Screens</h2>
      <p>Using media queries and responsive layouts</p>
    </div>
  </section>

    <div class="buttons">
        <button>Learn More</button>
        <button>Get Started</button>
        <button>Contact Us</button>
        <button>Subscribe</button>
    </div>

  <footer>
    <p>&copy; 2025 All rights reserved.</p>
  </footer>

</body>
</html>
```

Output:

Desktop View:



Mobile View:



Explanation:

- **Media Queries (Responsive Design)**

```
@media (max-width: 480px) {
 body {
  display: block; /* Previously was display: none (hidden), now fixed */
 }
}
```

- **Used to target mobile screens. This ensures the body remains visible (fixing earlier issue).**

```
@media (min-width: 481px) and (max-width: 768px) {
  .hero-text h2 {
    font-size: 28px;
  }
  .hero-text p {
    font-size: 16px;
  }
}
```

- **Adjusts typography for tablet view to ensure readability and proper scaling.**

```
    @media (min-width: 769px) {
      .hero-text h2 {
        font-size: 40px;
      }
      .hero-text p {
        font-size: 20px;
      }
    }
```

Conclusion:

Today's work concluded my responsive layout implementation and enhanced my understanding of how **media queries** can deliver dynamic and user-centered web design. This practical experience is crucial for modern front-end development and forms the foundation for building real-world, production-ready web applications.