



Live Cohort

Notes Day 7



Day 5 : Responsive with CSS

SCSS (Sassy CSS)

Relative Units

Purpose: Ensures scalable and responsive typography and spacing.

Techniques:

- % - Percentage relative to parent.
- rem - Root em, relative to <html> font size (preferred for global scaling).
- em - Relative to the current element's font size.

Example:

```
html {  
  font-size: 16px;  
}  
  
p {  
  font-size: 1.2rem; // Scales with root font size  
}
```

Day 5 : Responsive with CSS

SCSS (Sassy CSS)

Nesting

Purpose: Improves readability by grouping related styles together.

Techniques:

- Nest selectors following HTML structure.
- Limit nesting depth (max 3-4 levels).

Example:

```
.nav {  
    background: #333;  
  
    ul {  
        list-style: none;  
        li {  
            display: inline-block;  
            a {  
                color: white;  
            }  
        }  
    }  
}
```

Day 5 : Responsive with CSS

SCSS (Sassy CSS)

Variables

Purpose: Store reusable values for consistency and easy updates.

Techniques:

- Define global variables in _variables.scss.
- Use \$variable-name: value;

Example:

```
$primary-color: #3498db;

.button {
  background: $primary-color;
}
```

Day 5 : Responsive with CSS

SCSS (Sassy CSS)

Mixins

Purpose: Create reusable blocks of CSS code for flexibility.

Techniques:

- Define with @mixin and include with @include.
- Use parameters for flexibility.

Example:

```
@mixin flex-center {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}  
  
.container {  
  @include flex-center;  
}
```

Day 5 : Responsive with CSS

SCSS (Sassy CSS)

Partials & @import

Purpose: Helps organize and modularize SCSS files.

Techniques:

- Store styles in _variables.scss, _mixins.scss, etc.
- Import using @import 'filename';

Example:

```
// _variables.scss
$primary-color: #ff5733;

// main.scss
@import 'variables';
body {
  background: $primary-color;
}
```

Day 5 : Responsive with CSS

Performance Optimization

Optimize Images

Purpose: Reduces file size for faster loading.

Techniques:

- Use tools like TinyPNG, ImageOptim.
- Choose appropriate formats: JPEG, PNG, WebP, SVG.
- Resize images for different screen sizes.
- Use responsive images with srcset.

Example:

```

```

Day 5 : Responsive with CSS

Performance Optimization

Lazy Loading

Purpose: Defers loading of offscreen images to improve performance.

Techniques:

- Use loading="lazy" for and <iframe>.
- Use JavaScript Intersection Observer for advanced loading.

Example:

```

```