

CSS Variables and Custom Properties

CSS Variables, also called **Custom Properties**, allow you to store values in a reusable way — making your CSS more maintainable and dynamic.

They follow the pattern of:

```
--custom-name: value;
```

Declaring a CSS Variable

Variables are declared inside a selector using the `--` prefix:

```
:root {  
  --primary-color: #3498db;  
  --font-size: 16px;  
}
```

- `:root` is the highest-level selector (like `html`) — variables here are global.
- Variables declared inside `:root` can be used throughout your stylesheet.

Using a CSS Variable

Use the `var()` function to apply the variable:

```
body {  
  color: var(--primary-color);
```

```
font-size: var(--font-size);  
}
```

Why Use CSS Variables?

☒ Consistency ☒ Easy to update (change in one place) ☒ Theme support (light/dark mode) ☒ Cleaner, scalable CSS

Example: Theming with CSS Variables

```
:root {  
  --bg-color: white;  
  --text-color: black;  
}  
  
body {  
  background-color: var(--bg-color);  
  color: var(--text-color);  
}
```

You can override these in a different class for themes:

```
.dark-theme {  
  --bg-color: #121212;  
  --text-color: #ffffff;  
}
```

Now just add `class="dark-theme"` to `<body>` or a wrapper div to switch themes.

Fallback Values

If a variable isn't defined, you can specify a fallback:

```
h1 {  
  color: var(--heading-color, blue);  
}
```

If `--heading-color` is not set, `blue` will be used instead.

Scoped Variables

Variables can also be scoped to a class or element:

```
.card {  
  --border-radius: 10px;  
  border-radius: var(--border-radius);  
}
```

Only elements within `.card` can access this variable.

Real-World Example

```
:root {  
  --btn-padding: 12px 24px;  
  --btn-color: #fff;  
  --btn-bg: #2ecc71;  
}  
  
.button {  
  padding: var(--btn-padding);  
  color: var(--btn-color);  
  background-color: var(--btn-bg);  
}
```

```
border: none;  
border-radius: 6px;  
cursor: pointer;  
}
```

Update theme by just changing variables in `:root` .

Summary

- Use `--variable-name` to declare and `var(--variable-name)` to use.
- Declare in `:root` for global usage.
- Support theming, dynamic styles, and cleaner code.
- Can be scoped or overridden for flexibility.