

Customizing System Themes

This lesson shows how to override the default system themes definition.

Customizing the default system themes involves defining the `$ct-themes` map in your **variables** file. Here's the *default* definition inside `css-theming`:

```
$ct-themes: (  
  'default': (  
    'brightness': 'light',  
    // 'colors': overrides $ct-colors-[light|dark]  
    // 'semantic-colors': overrides $ct-semantic-colors  
    // 'font': overrides $ct-font  
    // 'text-color': overrides $ct-text-color-[light|dark]  
    // 'bg-main': overrides $ct-bg-main-[light|dark]  
    // 'fg-target': overrides $ct-fg-target-[light|dark]  
    // 'fg-swatches': overrides $ct-fg-swatches-[light|dark]  
    // 'bg': overrides $ct-bg-[light|dark]  
    // 'bg-focus': overrides $ct-bg-focus-[light|dark]  
    // 'bd': overrides $ct-bd-[light|dark]  
    // 'bd-focus': overrides $ct-bd-focus-[light|dark]  
  ),  
  'default-dark': (  
    'brightness': 'dark',  
  ),  
) !default;
```

Some of the theme props are well known by `css-theming` and it'll try to fall back to defaults when you don't override them. All those props are mentioned in the code snippet above as comments. The only required prop is `brightness`.

In addition to those well-known props, you can add whatever prop you want to change between themes. We'll discuss this [later](#). Next, let's focus on customizing colors.

