



Lightening and darkening colors with CSS

CSS is adopting a lot of new features that in the past were only available in pre-processors or via Javascript. One of those features is the ability to lighten and darken colors like SASS `darken()` and `lighten()` functions.

There are multiple ways to achieve the desired goal. I've chose to use the [color-mix\(\)](#) function as defined in the [CSS Color Module Level 5](#).

According to the spec:

This function takes two [color](#) specifications and returns the result of mixing them, in a given [color-space](#), by a specified amount.

So basically, to use `color-mix()` you need three things:

- The color space you want to work in, most of the time this will be sRGB but it can be any color space supported in CSS
- The colors that you want to mix
- The percentage that you want to mix them by

Right now (as of 11/25/2021) the examples below only work in Firefox Nightly. There is an entry in Chromestatus dashboard for the API, but there's no clear timeframe for when it'll be available for testing or production.

The first experiment will lighten a darken color in 10% increments.

We define a variable for our base color (`--color-charcoal`) and then we add nine more variables for the lightened colors.

Each of the lightened colors uses `color-mix()` by mixing the color we want to lighten, the base color (white) and the percentage we want to lighten the color by.

```
:root {  
  --color-charcoal: #222023;
```

```
--text1: color-mix(in srgb, var(--color-charcoal) 10%, white);
--text2: color-mix(in srgb, var(--color-charcoal) 20%, white);
--text3: color-mix(in srgb, var(--color-charcoal) 30%, white);
--text4: color-mix(in srgb, var(--color-charcoal) 40%, white);
--text5: color-mix(in srgb, var(--color-charcoal) 50%, white);
--text6: color-mix(in srgb, var(--color-charcoal) 60%, white);
--text7: color-mix(in srgb, var(--color-charcoal) 70%, white);
--text8: color-mix(in srgb, var(--color-charcoal) 80%, white);
--text9: color-mix(in srgb, var(--color-charcoal) 90%, white);
}
```

We then create classes for each of the lightened colors where we use the css custom property in the background-color property.

```
.dark-text1 {
  background-color: var(--text1);
}

.dark-text2 {
  background-color: var(--text2);
}

.dark-text3 {
  background-color: var(--text3);
}

.dark-text4 {
  background-color: var(--text4);
}

.dark-text5 {
  background-color: var(--text5);
}

.dark-text6 {
  background-color: var(--text6);
}
```

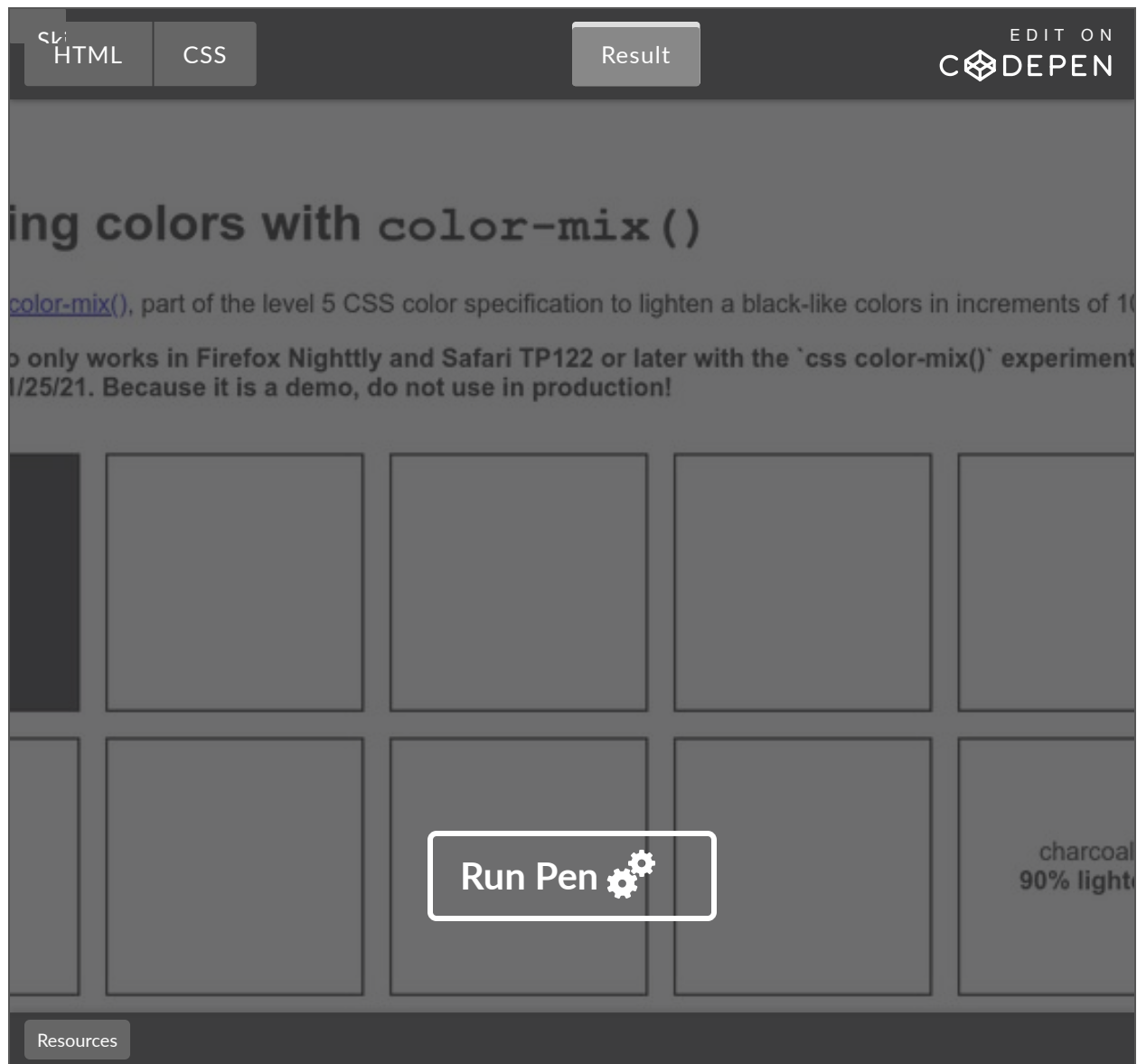
```
}

.dark-text7 {
  background-color: var(--text7);
}

.dark-text8 {
  background-color: var(--text8);
}

.dark-text9 {
  background-color: var(--text9);
  color: white;
}
```

View the following Pen in Firefox or Safari TP 122 or later with the `css color-mix()` experimental feature enabled to see the results of the code and how it lightened the color.



Darkening a color follows a similar process. We define a variable for our base color (`--color-mint`) and then we add nine more variables for the darkened colors.

```
:root {  
  --color-mint: #98ecc3;  
  
  --light-text1: color-mix(in srgb, var(--color-mint) 10%, black);  
  --light-text2: color-mix(in srgb, var(--color-mint) 20%, black);  
  --light-text3: color-mix(in srgb, var(--color-mint) 30%, black);  
  --light-text4: color-mix(in srgb, var(--color-mint) 40%, black);  
  --light-text5: color-mix(in srgb, var(--color-mint) 50%, black);  
  --light-text6: color-mix(in srgb, var(--color-mint) 60%, black);  
  --light-text7: color-mix(in srgb, var(--color-mint) 70%, black);  
}
```

```
--light-text8: color-mix(in srgb, var(--color-mint) 80%, black);  
--light-text9: color-mix(in srgb, var(--color-mint) 90%, black);  
}
```

We then create the classes that use the variables in the background-color property.

To ensure proper contrast, I've also set the text color to white in all the darkened colors.

```
.mint {  
  background-color: var(--color-mint);  
  color: black;  
}  
  
.light-text1 {  
  background-color: var(--light-text1);  
  color: white;  
}  
  
.light-text2 {  
  background-color: var(--light-text2);  
  color: white;  
}  
  
.light-text3 {  
  background-color: var(--light-text3);  
  color: white;  
}  
  
.light-text4 {  
  background-color: var(--light-text4);  
  color: white;  
}  
  
.light-text5 {  
  background-color: var(--light-text5);
```

```
    color: white;
}

.light-text6 {
  background-color: var(--light-text6);
  color: white;
}

.light-text7 {
  background-color: var(--light-text7);
  color: white;
}

.light-text8 {
  background-color: var(--light-text8);
  color: white;
}

.light-text9 {
  background-color: var(--light-text9);
  color: white;
}
```

The following pen shows the results of the code. It will only work on Firefox Nightly and Safari TP122 or later with the `css color-mix()` experimental feature enabled.

SL

HTML

CSS

Result

EDIT ON
CODEPEN

g colors with `color-mix()`

`color-mix()`, part of the level 5 CSS color specification to darken a dark green color in increments of 5%. It only works in Firefox Nightly and Safari TP122 or later with the ``css color-mix()`` experimental flag. Because it is a demo, do not use in production!

| | | | | |
|--|--|--|--|--|
| | | | | |
| | | | | |

Run Pen ⚙️

Resources

If you're interested in getting this feature in Chromium browsers, follow and star the [chromium bug](#).