# Styling buttons

I was working on a side project where I have a series of buttons but when I tried to style them the CSS did nothing.

In doing research I discovered several things you can do and that I wasn't doing and things that I was doing wrong.

I'm working with SCSS/SASS because it's easier for me to nest rules and figure out associations between elements. It will be converted to CSS at build time.

## Quick Reset

The example I'll use in the rest of the post uses a reset. This could be used in addition to [normalize.css](#) or Eric Meye's [CSS Reset](#) or as a standalone quick reset for all button elements. .

```scss
button {
  background-color: transparent;
  border: none;
  cursor: pointer;
  color: inherit;
  font: inherit;
  padding: 0;
  // outline: 3px solid blue;
}
```

## Styling buttons

I start with setting the buttons to be `inline-block` rather than inline or block to keep being able to style and align them together if I so choose. (1)

> display: inline-block brought a new way to create side by side boxes that collapse and wrap properly depending on the available space in the containing element.
>
> From: [Inline vs Inline-Block Display in CSS](#)

We align the text and remove any existing underline.

In the next block we define margins, border-radius. It appears that this is the only way to size buttons. (2)

Next we define color for the text and background. In this case we use a [linear-gradient](#) as an experiment. (3)

```css
.btn {
  display: inline-block; /* 1 */
  text-align: center; /* 1 */
  text-decoration: none; /* 1 */

  margin: 2px 0; /* 2 */
  border-radius: 4px; /* 2 */
  padding: 1em 3em; /* 2 */

  color: #ffffff; /* 3 */
  background:
    linear-gradient(#9198e5, #e66465);/* 3 */
```

The two pseudo classes, `:active` and `:hover` define two possible states for the button.

`:active` indicates the styles active when the user click the button. For this state we move the element to the right and saturate the color (make it darker) and add an outline. (4)

```css
&:active { /* 4 */
  transform: translateX(10px);
  filter: saturate(300%);
  outline: 3px solid blue;
}
```

`:hover` provides the styles for when the user hovers over the button. In this state we reverse the colors fooor backgroound and text.

```
  &:hover { /* 5 */
    color: white;
    border-color: currentColor;
    background: transparent;
    background-color: lightblue;
    outline: 3px solid blue;
  }
}
```

# Accessibility

One additional thing to consider is accessibility.

Wes Bos asked on Twitter this quesndtion about button accessibility.

> If I have a <button> and I want a screen reader to announce it
> as "Remove Eggs", what attribute do I use? title? aria-label ?
> pic.twitter.com/RRnlx733ev
>
> — Wes Bos (@wesbos) November 1, 2019

In particular pay attention to Andy Bell's answer in the Gist he provided

# Conclusion

This button is a test. All colors and actions are done as a test and will most likely change foor production. You can see a working example in this codepen