

Further Styling an Autocomplete Enabled Engine

In this lesson, we'll be modifying our backend component look like Google's.

Let's write some actual code! The first thing we'll be doing is adjusting positions and dimensions:

Output

HTML

CSS (SCSS)

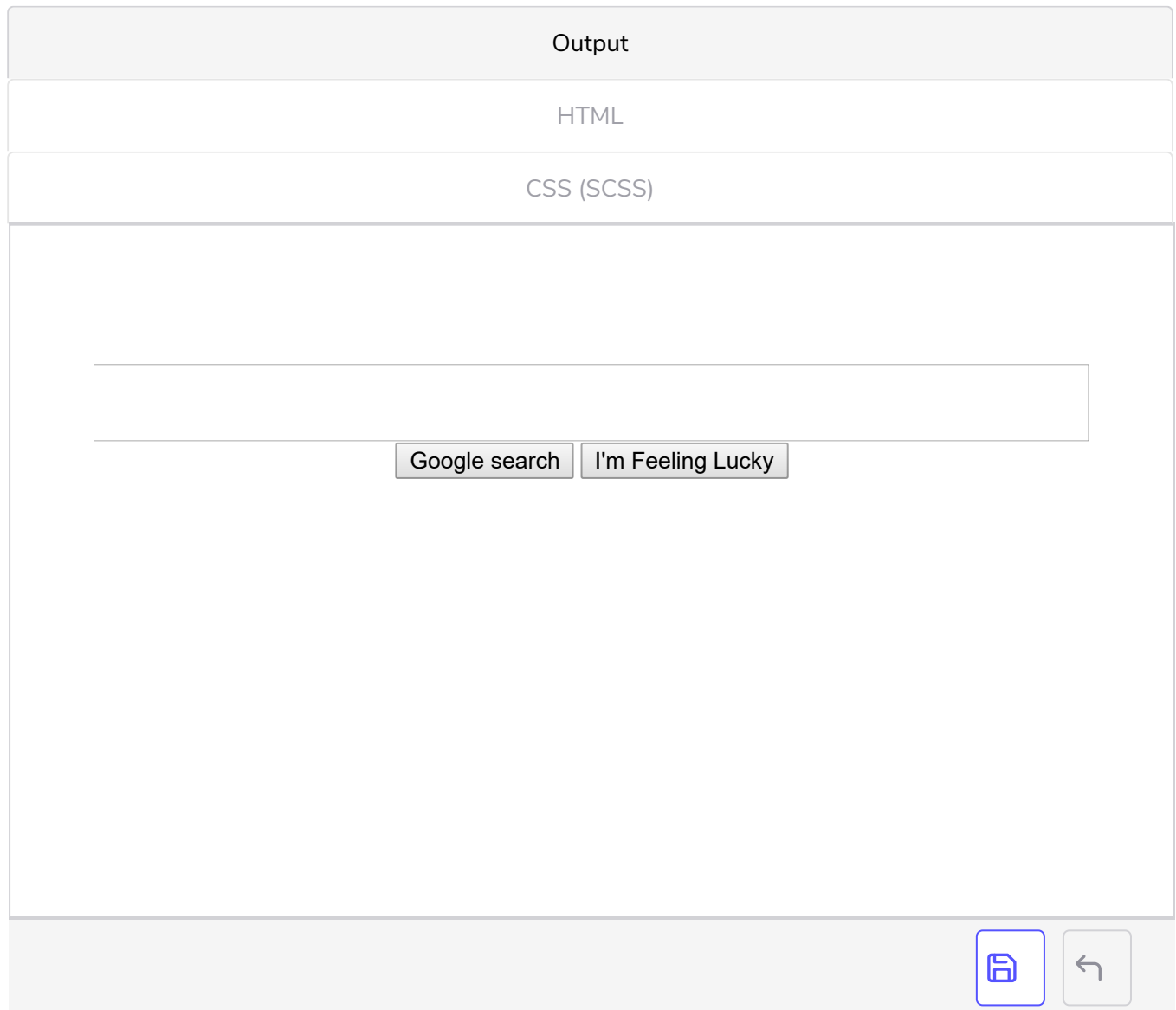
Google searchI'm Feeling Lucky

It's a bit awkward to demonstrate an example with too much height change in an iframe like above, so I'll adapt the component to have less margin. On a regular webpage, 5rem is probably too little and leaves too much space below.

Two things have made reappearances from the last lesson:

- Coloring of subcomponents in development. There's a pixel or a margin between the search input and the component edges.
- `calc` to dynamically adjust the width. If it had just been `width:100%`, the padding would make the input look as if it extends past the `search` component, and the buttons would no longer look centered.

Next, we'll format the autosuggest. We need to remove the default list decorations, give each row the greyish hover that Google has, and put a bit of padding between each item.



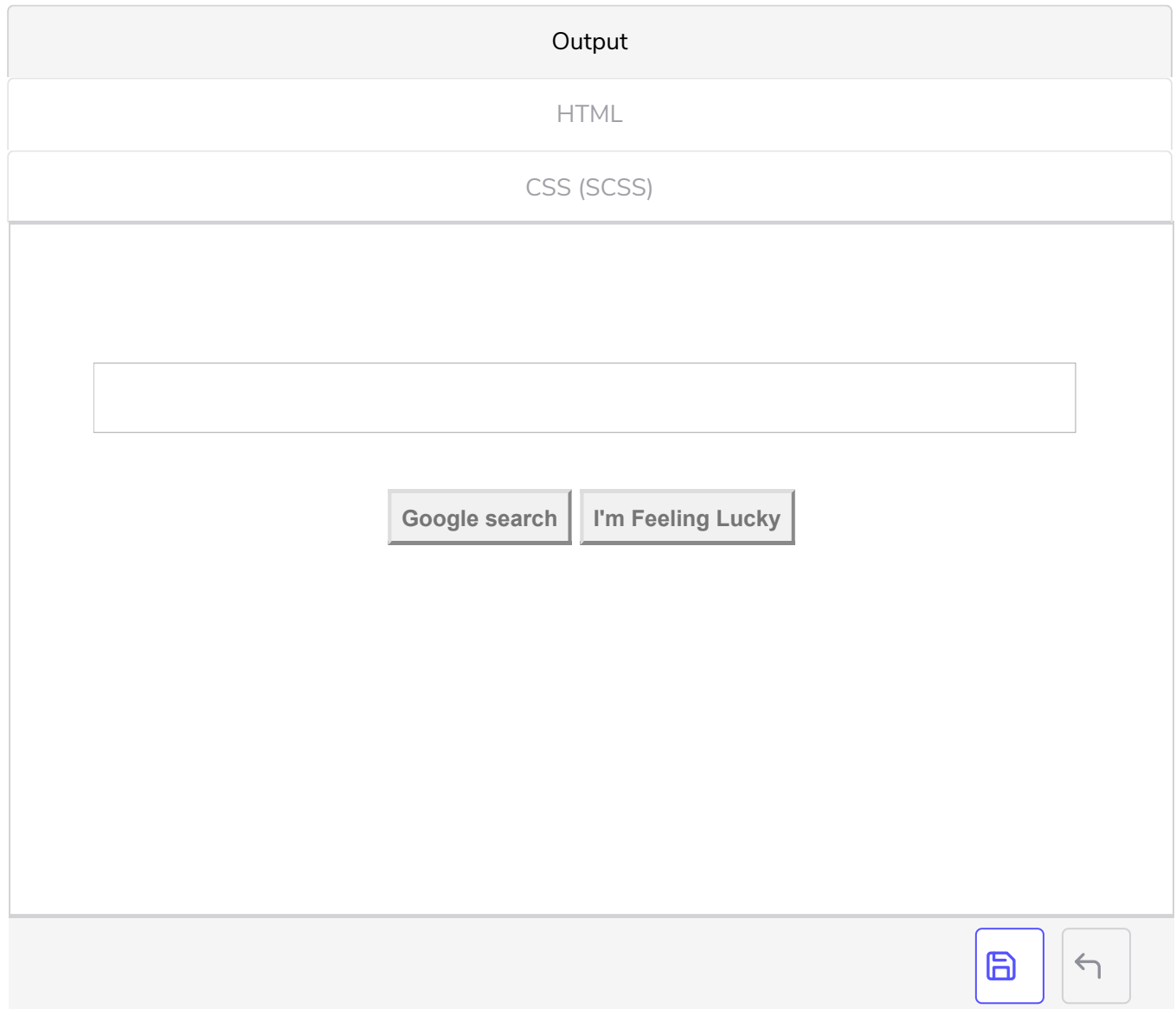
At this point, I realize that some of my results exceed one line, which feels like a bad user experience because Google doesn't have borders between results, so users have to assume that each new line is a new result. The limits I set for my server-generated results were just approximations, so I'll go back to adjust them as needed.

Before we go on to making things pixel-perfect, we should finalize the font. Different fonts have different spacings between characters, and if you change

fonts once you’ve positioned, you might risk having to redo things.

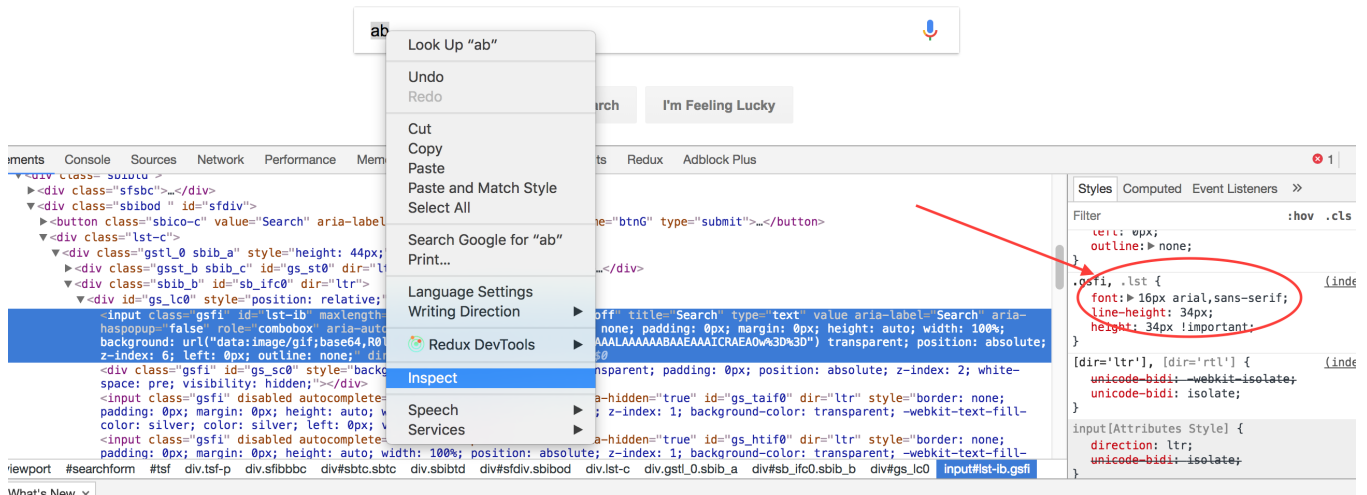
If you’re ever wondering what fonts website use, by the way, there are browser extensions to provide that convenience.

Let’s apply the font changes globally, along with some more sizing changes.



Discovering font and color through easy-to-use extensions is great and all, but sometimes you’re looking for further details on how an effect is achieved. For example, what font size should I use? How thick is that border?

Modern browsers provide a suite of tools for web developers, easily accessible through right-click menu options. I briefly mentioned this in an earlier lesson, but this time I’m going to the “Styles” tab and finding out how much the font size is.



I also noticed that the border for the autosuggest stands out when there's no autocomplete results, so instead of giving that property to the `ul`, we gave it to the `last-child` of the list results.

Google search I'm Feeling Lucky

Alright, we're getting there! Let's continue making our frontend look like Google's in the next lesson starting with a little microphone icon.