

Adding Styling

This lesson will help you learn how to make your tooltip look pretty so it delivers a great user experience!

WE'LL COVER THE FOLLOWING ^

- Styling
- Changelogist

Styling

We've covered a number of bugs and brought the functionality to a point where it's usable. The styling isn't quite complete, so before we can call this "shipped," let's put the finishing touches and then add some animation for flare.

We'll start with the styling. Comparing the tooltip we've built to the model Medium has, I notice they add some rounded corners, add some padding, and use a dimmer color for the divider, so it doesn't stand out so much as our's does.

| Output |
|------------|
| JavaScript |
| HTML |
| CSS (SCSS) |

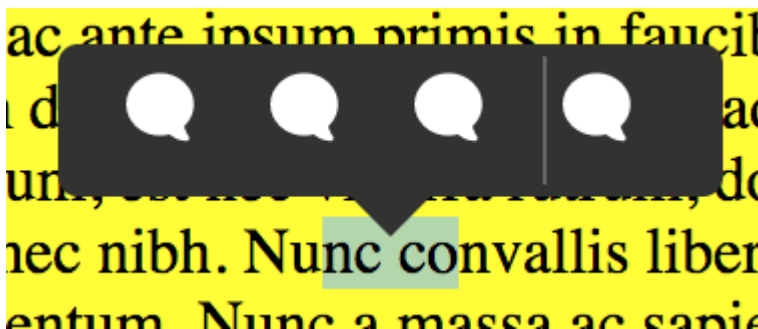
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum eget sem pellentesque, ultricies dolor egestas, malesuada mauris. Fusce lorem felis, egestas tincidunt purus et, sodales luctus lectus. Praesent a congue eros. Interdum et malesuada fames ac ante ipsum primis in faucibus. Mauris vitae malesuada urna. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut in mauris justo. Curabitur rutrum, est nec viverra rutrum, dolor est volutpat dolor, vitae auctor urna nulla nec nibh. Nunc convallis libero luctus purus auctor, sed aliquet lectus fermentum. Nunc a massa ac sapien viverra consequat.

Aliquam arcu mi, vehicula a sodales ut, dictum sit amet ex. Sed nec enim sed elit dapibus porta at eu felis. Quisque faucibus vel sem in bibendum. Nulla dictum, nunc in mattis tincidunt, nisi sapien blandit metus, ut bibendum tortor metus non nunc. Duis eu ultricies risus, quis mollis magna. Duis efficitur congue ante ut imperdiet. Vestibulum at sodales eros. Etiam id dictum ligula.



Changelist

- Defined a divider color.
 - Sometimes, even if something is used once, it could make sense to create a variable for it. This is especially true if the context is unclear
- Separate divider into its own div
 - Since you can't set percentage heights for borders, I let the divider occupy its own div to do so. Otherwise, the border would've looked like this:



- I should note there's probably hacky ways to do it, like set a gradient that only has the middle 80% colored, and the rest is set to the background color to blend in. You want to avoid these sorts of solutions as much as possible, though. How do you recognize unconventional solutions? It simply comes from experience. Usually,

it's something new you haven't done or seen before and makes you feel uneasy about its brittleness.

- Wrap the `svg`'s in divs
 - We've seen before how it was useful to define our excessive divs early on. In this case, our SVG content was not perfectly flush to the space it occupied, which made positioning more difficult
 - To demonstrate, this is the svg with the background set to blue:



-
- Add padding and rounded corners
 - Not much to say here. I will note, however, one small detail. Notice how I've defined `padding: x` and then `padding-top: y` right below it. To me, this is cleaner than `padding: y x x x`, which would've been equivalent. This is purely based on taste, though.

Now that we're done with styling our tooltip, let's learn how we can add animations to it!