Bug 3: Multiline Highlights

In this lesson, we will try to fix the behavior of our tooltip on multiline highlights.

WE'LL COVER THE FOLLOWING

Multiline Highlight

There's a case where highlighting multiple lines make the tooltip appear in the middle of the line in which the highlight began. So it can fall outside the selected text.

My first thought is that we can get the width of the element surrounding the text and divide that by two for our x coordinate and whatever that Selection gives us for the `y` coordinate (at the beginning of selection). I inspect the rectangle that `selection.getRangeAt(0).getClientRects()[0]` gives, and compare it to when I highlight one line vs. multiple lines. I would expect the multiple line version to give a different height. But since that's not the case, I check the output of `getClientRects` and realize that it gives multiple rectangles when the selection calls for it. That makes a lot of sense since the shape of a multiline selection often can't be represented by one rectangle.

So the conditional logic for displaying the tooltip in the middle of the line that the selection started with is when `getClientRects` gives more than one rectangle, and the coordinates are given by the width of the surrounding node, and the `y` coordinate of the first rectangle.

Output

JavaScript

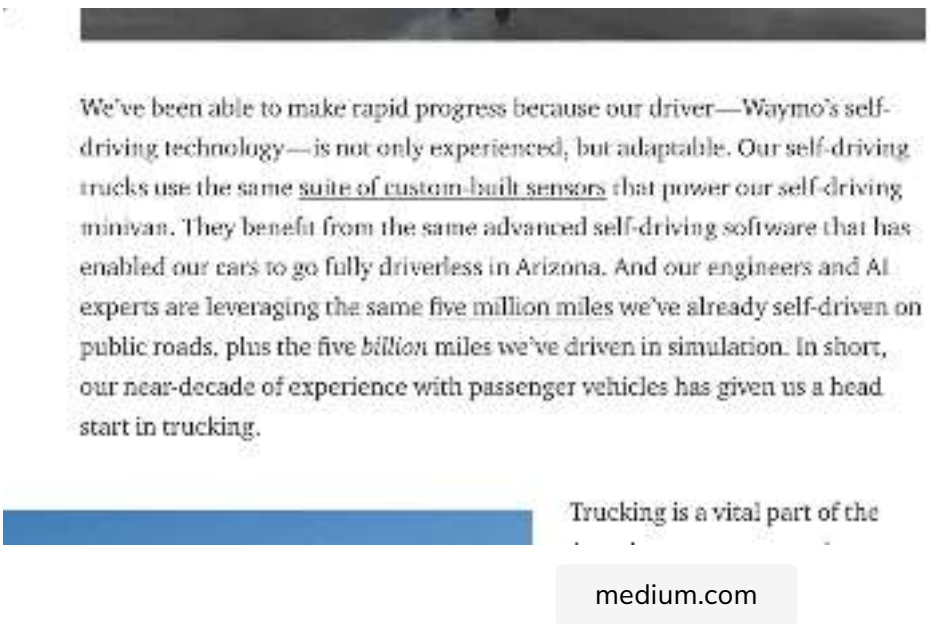
HTML

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.

That's all the bugs we'll be covering. If you try hard enough, we could potentially find others, but that's true of most things on the web. It's all held together by duct tape!



In the next lesson, we will take a look at adding CSS styling to our tooltip.