**Project Description/Summary**

This project involves creating a grocery list that uses local storage to save items.  When an item is added you will receive feedback using the Bootstrap alert-danger or the alert-success classes. This project also uses event propagation to transverse the DOM.

**New Things Learned or Refreshed**

I've stepped away from JavaScript for about a month because I was caught up at work. So, it was good to get back into the grove of coding again. Most all of this project was a refresh for me. This time, however, it was good have my memory refreshed on how event propagation works in JavaScript. If you follow the hosted project and then add an item to the cart, you'll see that each item has a trash can icon. When you click on the trash can icon, it actually has a few parent elements. Using the parent elements is how the list items are accessed. Specifically, the code is this: //delete one item listItems.addEventListener('click', function(event){ if(event.target.parentElement.classList.contains('remove-icon')){ let parent = event.target.parentElement.parentElement; listItems.removeChild(parent); let text = event.target.parentElement.previousElementSibling.textContent; clearSingle(text); } }); This event listener listens for a click on the trash icon, which has a parent element with a class of remove icon. Once this item is retrieved, its parent..and the parent of its parent…is then put into the parent variable. This allows for the remove of the entire element. Pretty clever if you ask me. he other thing I was refreshed on was the use of [localStorage](https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage). It's not complicated once you learn it, but it's easy to forget if you don't use it. So, I'm glad I looked at this again.

**Time to Code**

This took about two hours to code.  I watched John's solution for the most part and coded along as he wired up the project.

**Biggest Take Away(s)**

None this project.