**Explanation**

In order to change the color of the new price in red and calculate the discount, I had to solve a few issues:

1. Some of the elements that were easier to work with were being removed from the dom through ajax
2. The second one had to do with interacting with those checkboxes.

Firstly, I’ve noticed that a set of elements were being added and removed dynamically through ajax. As I result I had to rely on some elements that were anchored more firmly in the document such as the “.main-wrapper” element.

Due to the way events propagate through the DOM I chose to attach my event listener on a ascendant element that seemed to be well anchored in the document and then every time this element would receive the click event the handler will do the work and update the correct product.

Inside my event handler I’ve made a call to a setTimeout function and passed in:

1. A function that will do the actual work: identify the right element based on some selector, get the price attribute, process the price (if string parse it, if number all good, if array throw an error) and call a “doDiscount” function that will return the discounted price
2. A 1 s timeout that I assumed it would suffice to get the data back from the backend

The setTimeout call was necessary because of the time that took for the products to be filtered and the DOM to be updated.

One issue that I have come across was the ability to execute my function on page load. According to MDN the load event is fired when the whole page has loaded. Because I didn’t have the ability to simply insert a script tag inside the source code and listen to that event I had to resort to a different alternative in order to test my code and I have simulated this event using the “dispatchEvent” function that now is available in most modern browsers.