1. In the JavaScript code for the opening page (page titled “Openinghtm”l), the “const left”, “const right” and “const container” followed by the “document.querySelector(‘.right’)” (or (‘.container’, (‘.left’) selects the css element .right, .left or .container respectively.
2. Then, when we have .left or .right “addEventListener” for (‘mouseenter’) or (‘mouseleave’) the container has a “classeList.add(‘hover-left’)” or ”classList.remove(‘hover-right’)”. This makes it so when the mouse hovers over the right or left side it will add or remove the class “hover-right” and “hover-left” to the container class. This will cause the css rules which apply to .hover-right and .hover-left to take effect which is to change the width of these div classes depending on which side is hovered over by the mouse.
3. For the search button, (page titled Search ) the css rule for “.split.left” and “.split.right” of “.btn:hover” causes the color and border-color of the button to change when the mouse is hovered over the button for both the right side and the left side.
4. For the search icon in the JavaScript code the “.addEventListener” was used to make it that for the button and the search bar (which was given a class of “btn” and “search” in html and a const of ”btn” and “search” in the JavaScript) the class “.active” would be applied to it which in the css has rules making the with of the search expand and the button to move horizontally creating the effect that it has.
5. A transition of “ease” was used in the css for “.search.input” making it having a smooth transition when clicked on.
6. For the expanding panels (page titled Watches) in html we linked images and classed as panels and had one with an active class. For each panel we had an “addEventListener” with the action of “(‘click’)” which added the active class to the panel clicked. We formatted the size of our pictures and the transitions in css with different css styling for active class versus just a panel class.
7. There was a “removeActiveClasses” so that when each panel was clicked the class “active “ was removed from the other panel.
8. For the scroll animation (page titled Products) in html we made a series of divs in a class of box. In css we formatted the text in the boxes. The “transform” property was used with a value of “transalteX(400%)” and “(-400%)” so that it would be able to scroll on and off the page.
9. In Javascript we used a query selector and an “addEventListener” that when you “(‘scroll’)” it activates the “checkboxes” function.
10. The “checkboxes” function uses the “box.getBoundingClientRect().top” to get the positioning of the box relative to the top. The code uses an if statement to see if the “boxTop” is less than the “triggerBottom”. If it is, then the class of “(‘.show’)” is added. If not, then it is removed.