

Grey paper – Elazar Kagan – website project 1

1) JS:

```
const color_icons = document.querySelectorAll('.cicon')

color_icons.forEach(cicon => {
  cicon.addEventListener('click', () => {
    removeActiveClass()
    cicon.classList.add('clicked')
  })
})

function removeActiveClass() {
  color_icons.forEach(cicon => {
    cicon.classList.remove('clicked')
  })
}
```

CSS:

```
.buywords img {
  position: relative;
  margin: 6px;
  height: 8%;
  top: -50px;
  border-radius: 50%;
}

.clicked {
  border: 3px solid #9615db;
}
```

This is the javascript for the action of clicking one of the color options for it to have a purple border around it. First, I have it find all the color icons which all have the class of cicon, then I have it listen for the user to click the icon, when that happens the JS will add a class of clicked to the icon that got clicked. In the CSS I wrote that anything that has the class of clicked will have a purple border. I also had to create the border-radius property and set it to 50%, so the boarder goes around the circular icon and not a box on top of it. Back to the JS, it also listens for another icon to be clicked, if yes it will remove the clicked class from the first icon and also add it to the new icon now making the new choice have the purple border.

2) HTML:

```
<div class="main">
  
```

```

<div class="words">
  <div class="title">
    <h1>Welcome to the future, <br>Introducing: <span id="colored">Piano Gloves</span></h1>
  </div>
  <div class="subtitle">
    <h3>The one-of-a-kind experience allowing you to play piano on any surface</h3>
  </div>
  <div class="mainbtn">
    <button>Buy now</button>
    <button><i class="fas fa-play"></i>&nbsp;&nbsp;&nbsp;How it works</button>
  </div>
</div>
</div>

```

The HTML of the of the first section, it includes the main picture and the words that will be displayed next to it, it also has the 2 buttons and a play icon inside the second button. I used divs to make it easier to add css to each part.

3) CSS:

```

.container{
  background-color: #1a1a1c;
  position: absolute;
  height: max-content;
}

.main img {
  width: 70vw;
  float: right;
  z-index: 1;
  position: relative;
  top: 80px;
}

```

The CSS is used to format the picture of the first section, the background of the image is a black gradient but the left edge of the picture has the same color of #1a1a1c, so I pushed the image to the right and set the background to #1a1a1c to have the picture flow into the background.

4) CSS:

```

.title{
  position: relative;
  padding-top: 10%;
}

```

```

font-family: 'azonix', sans-serif;
font-size: 130%;
text-transform: uppercase;
font-weight: light;
overflow: hidden;
width: fit-content;
}

```

This is the CSS for the opening text, I wanted to have it beside the image and overlapping a little, so I set the div that houses the words as a relative position, also I set the font to azonix using the font-family property while still having sans-serif as a backup. The size of the font was made bigger using the font-size property, also I transformed the text to all uppercase using the text-transform property.

5) CSS:

```

.main button{
    background-color: #9615db;
    border-radius: 4px;
    border-style: none;
    color: #FFFFFF;
    cursor: pointer;
    padding: 15px 15px;
    margin-right: 4%;
    transition: color 100ms;
    width: 200px;
    height: auto;
}

.main button:hover,
.main button:focus {
    background-color: #d2d3d3;
    color: #9615db;
}

```

I styled the 2 main buttons the same, staying with my color scheme I used #9615db (purple) as the background, gave it some curved edges with the border-radius property, made the button bigger by giving the words inside some padding with the padding property. Also I added :hover selector to make the color of the background and the color of the text to switch, so when hovering over the button the background will turn grey while the text turns purple.

6) CSS:

```

.top {
    background-color: white;
    height: auto;
}

```

```

width: 100vw;
position: fixed;
text-align: center;
color: #9615db;
font-family: 'azonix', sans-serif;
text-transform: uppercase;
display: flex;
justify-content: center;
z-index: 50;
}

```

This code is for the top bar, I used the flex declaration so I format the logo in the middle, the search bar on the right, and the navigation button on the left. Also, I wanted this top bar to stay at the top of the screen even if the user scrolls down, so I used the position: fixed declaration to have it stay on the top of the user viewport always.

7) HTML:

```

<div class="popup" onclick="popupFunction()">
  <button id="sizechart">size chart</button>
  <div class="popupchart" id="myPopup">
    <table>
      <tr>
        <th>Size</th>
        <th>Width</th>
        <th>Length</th>
      </tr>
      <tr>
        <td>small</td>
        <td>10cm</td>
        <td>19 cm</td>
      </tr>
      <tr>
        <td>medium</td>
        <td>10.5 cm</td>
        <td>19.5 cm</td>
      </tr>
      <tr>
        <td>large</td>
        <td>11 cm</td>
        <td>20 cm</td>
      </tr>
    </table>
  </div>
</div>

```

For the table needed on my website I created a chart that give measurements for sizes so when the user is going to pick a size they have the option to click on a size button that will popup the chart. So I have the popupFunction, which was coded in JS, set to on click so it pops up. The chart tells you based on the length and width of your hand which size to get.

8) CSS:

```
.dropdown {
  position: relative;
  left: 20px;
  width: 120px;
  height: 40px;
  color: #fff;
  border: 1px solid #fff;
  background-color: #4b5054;
  border-radius: 4px;
  padding: 10px;
}
```

On the buying section of the website, I have a dropdown option asking the user to select a size, I wanted to position it next to the button for the chart explaining the measurements of the sizes, so I gave the dropdown a relative position and manually used the left and height properties to set in pixels exactly where the dropdown should be. Also, I added some style to the dropdown setting the background color to the same grey as the background to give it a transparent look, also i added a boarder and softened the edges and added some padding between the words and the edge of the box.

9) CSS:

```
#addtocart {
  position: relative;
  top: -20px;
  margin: auto;
  margin-top: 5px;
  background-color: #fff;
  border-radius: 4px;
  border-style: none;
  color: #9615db;
  cursor: pointer;
  padding: 15px 155px;
  text-transform: uppercase;
}

#addtocart:hover {
  color: #fff;
  background-color: #9615db;
}
```

I gave the add to cart button an id so I can edit it directly, it is set to relative position so it can be right under all the other text of the buy page. I wanted to have the button stretch all the way across where the words were, so I set the padding to have 155 pixels on the sides. Also, I set the cursor to pointer so it looks like you can click the button. The button has a white background with purple words, but in the next lines I added the :hover to switch the colors when hovering over the button so if the user hovers over the button it will change to purple background and white words.

10) CSS:

```
.footer {  
  text-align: center;  
  position: relative;  
  bottom: 150px;  
  margin: -90px;  
}
```

There was a lot of extra space at the bottom of the website (probably because I used the relative position a lot). So, for the footer I set the margin to a negative value so it will take away the extra space around the words.