

Events

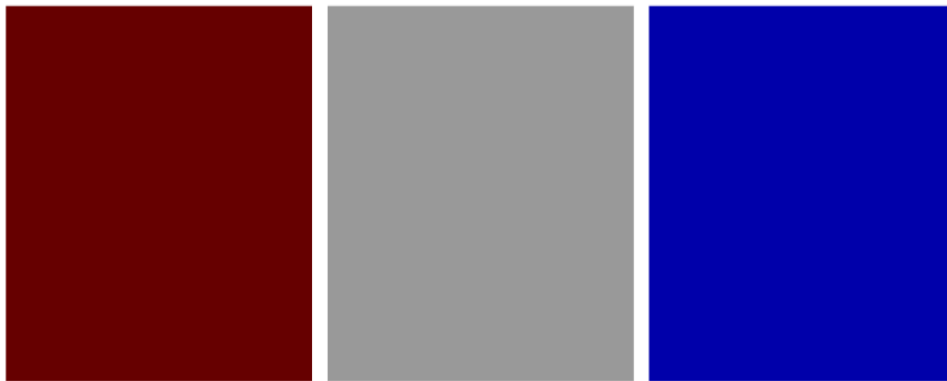
Due Tuesday by 11:59pm Points 10

Color Changer

Create an application with three grey, square divs, in a row. They should be 200px by 200px, and all floated left. Give them a margin of 5px.

Using only one event handler, write event listeners to respond to a click on each element. Each element should change to a different color: one red, one green, and one blue. Use a data attribute on the elements to store the color to be changed to.

The result, after clicking on the first and third, should look something like this:



Color Mixer

This is an application that will start a div at a black color, and as a user presses buttons associated with RGB, the div will change to new colors. The button presses should affect additively - that is. If a div is currently blue, pressing +red buttons will change the div to a purple color.

9 buttons (associated with red green and blue). There should be a +1, +5, and +10 button for each color.

1 div that will change colors to the rgb color calculated

1 div that shows the current calculated rgb color

You **must** use attributes on the buttons for the values to change the colors by.



current color: rgb(154,110,212)

Red:	+1	+5	+10
Green:	+1	+5	+10
Blue:	+1	+5	+10

Flash Cards

Markup / HTML: One div that will be used for answer output. A decent ID for this div might be "dvAnswer"

Data: Create data for at least **two** question-answer pairs in an array of objects, like so:

```
let questions = [ { question: "What are the four Bartle player types?", answer: "Killer, socializer, achiever, explorer" }];
```

Then, using a loop, create buttons on the page (using `document.createElement`) that puts the question in the button's `innerHTML`, and the answer in a custom attribute (using `setAttribute`).

Interaction: When a question button is pressed, `dvAnswer` should show the answer to the question. You'll need to use an `addEventListener` to make this interaction work.

Character Maker

Using the files in: [characterImages.zip](#), create write the markup to place two images on the page. Both should be styled to have a position of `absolute`, a top of `0px` and a left of `0px`. Make sure the "shirt" image is on top - this is done by making it the last tag in the HTML order.

Then, write the code to make it so that when you click on the shirt image, it changes to the next in the series. Thus, as you click on the image it will display `shirt1`, then `shirt 2`, then `shirt3`. On a click of `shirt 3`, return back to `shirt one`.

This interaction should be done using one event handler and the `setAttribute` method.