



Rev. 08Aug2019

# Exercise 39

## EVENT BUBBLING

In Exercise 37, we began using JavaScript to assign an *event listener* programmatically to an element. In this exercise, we're going to look at a case where we have several elements that we want to monitor for events. Let's look at [index.html](#):

```
<h2>Which flavor of ice cream would you like?</h2>
<div id="flavors">
  <button value="vanilla">Vanilla</button>
  <button value="chocolate">Chocolate</button>
  <button value="stawberry">Strawberry</button>
  <button value="coffee">Coffee</button>
  <button value="mint chocolate chip">Mint Chocolate Chip</button>
  <button value="neapolitan">Neapolitan</button>
  <button value="rocky road">Rocky Road</button>
  <button value="butternut">Butternut</button>
</div>
```

We have eight buttons that we want to listen to for a click. We could do something similar to what we did in the last exercise: get all of the buttons as a nodelist, convert that to an array, then loop over the array attaching event listeners. But there's a better way.

You know that HTML represents a hierarchy of elements. That's why we talk about parents and grandparents, *etc.* As it turns out, an event that happens to one element, if ignored, *bubbles up* to the parent and we can trap it there.

That means we can put an event listener on `<div id="flavors">` (since that is parent to all the buttons) and "trap" the event there. You ready to do this?

- ❑ In [index.js](#), you'll see three comment instructions. Carry those out, run the repl (checking the console tab).