

A Tiny Scrap of JavaScript

Fifteen years ago, when "Web 2.0" was a thing, [Ben Nolan's Behaviour.js](#) enlightened me to the virtues of declarative programming. Developing for a web-browser really lends itself to this approach. Using "sheets" to target particular JavaScript code at selected DOM elements provides a way to separate, in a consistent way, the behavioral aspects of a web-page. For a while I'd subscribed to his advice to use [jQuery](#) instead, and this was incredibly effective but always felt like I was cheating. Including JQuery in a web-page for a tiny bit of convenience was just heavy handed. Now, 15 years later, I've finally weened myself off of that crutch too. Below is a tiny scrap of JavaScript that leverages the native JavaScript [Document.querySelectorAll\(\)](#) function...

```
function attachBehaviors(behaviors) {
  Object.entries(behaviors).forEach(([selector, behavior], index) => {
    document.querySelectorAll(selector).forEach(elm => {
      behavior.call(this, index, elm);
    });
  });
}
export {attachBehaviors};
```

With this function available, now pass a "sheet"...

```
attachBehaviors({
  'a.externallink': function(idx, elm) {
    elm.addEventListener('click', function(evt) {
      evt.preventDefault();
      window.open(evt.target.href);
    });
  },
  'form#uploadForm': function(idx, elm) {
    elm.addEventListener('submit', function(evt) {
      evt.preventDefault();
      processFiles();
    });
  }
});
```

Published by Wyatt

Tue Aug 10 06:28:43 EDT 2021

<http://blog.wylovan.com/pebble/wyatt/2021/08/10/1628591323638.html>