

Not all application state belongs inside your application's component hierarchy. Sometimes, you'll have values that need to be accessed by multiple unrelated components, or by a regular JavaScript module.

In Svelte, we do this with *stores*. A store is simply an object with a `subscribe` method that allows interested parties to be notified whenever the store value changes. In `App.svelte`, `count` is a store, and we're setting `countValue` in the `count.subscribe` callback.

Click the `stores.js` tab to see the definition of `count`. It's a *writable* store, which means it has `set` and `update` methods in addition to `subscribe`.

Now go to the `Incrementer.svelte` tab so that we can wire up the `+` button:

```
function increment() {  
  count.update(n => n + 1);  
}
```

Clicking the `+` button should now update the count. Do the inverse for `Decrementer.svelte`.

Finally, in `Resetter.svelte`, implement `reset`:

```
function reset() {  
  count.set(0);  
}
```