

Creating a Redux STORE

For Redux to act as a state container, we first need to build it. Let's use the Redux library to create a STORE for our app.

If the `<App />` won't manage our application's state, then we have to create a Redux STORE to manage our state object.

For a Bank Vault, a couple mechanical engineers were probably hired to create a secure money-keeping facility.

To create a manageable state-keeping facility for our application, we don't need mechanical engineers per se. We'll do so programmatically using some of the APIs Redux avails us.

Here's the code used to create a Redux STORE:

```
import { createStore } from "redux"; //an import from the redux library
const store = createStore(); // an incomplete solution - for now.
```



First, we import the **createStore** factory function from Redux. Then we invoke the function, **createStore()** to create the store.

Now, the createStore function takes in a few arguments. The first being a reducer.

So, a more complete store creation would be represented like this:

```
createStore(reducer)
```

Now, let me explain why we've got a **reducer** in there.