

# State and subscriptions lab

Your store will notify you whenever state is changed. And let's face it, when state changes our UI probably should be refreshed with the new information. Let's make that happen in this lab.

1. Open App.js, find the useEffect at the top and add a subscription to it:

```
store.subscribe(() => console.log("Subscription fired"));
```

This subscription will merely write to the console when anything is dispatched.

2. Run and test. Look in the browser's console. You shouldn't see anything ... yet.

3. Now add a dispatch:

```
store.dispatch({type:"F00"});
```

4. Again, look in the console. Now your message appears, proving that a dispatch() will fire subscriptions.

## Unsubscribing

Remember that if we subscribe on component creation, we should also unsubscribe on component teardown.

5. Change your useEffect to look like this:

```
useEffect(() => {  
  const unsubscribe = store.subscribe(() => console.log("Subscription fired."));  
  store.dispatch({ type: "F00" });  
  return unsubscribe; // <-- Add this line  
}, []);
```

This new *return* will tell the component to unsubscribe from the redux store when the component is disposed of.

## Refreshing the UI on state change

Finally, let's do something useful with that subscribe. Let's say that when the store updates, we'll refresh this UI. And setState does that.

6. Change the subscribe to do this:

```
const newState = store.getState(); // Get the new state from the store  
const unsubscribe = store.subscribe(() => setState(newState)); // Set it
```

7. Bonus! If you want to combine those two lines into one by getting rid of newState, go for it.

8. Finally, in preparation for future labs, find the <main> element and add this inside it:

```
<section>  
  {state.films.map(film => <div key={film.id}>{film.title} - {film.tagline}</div>)}  
</section>
```

Again, this is just some React which we'll cover later. But for the curious, this will iterate through all of the state's films and show each title and tagline in a <div>.