How to run lifecycle events

1. Open App.js and look at the top of the file. In it we're doing something like this:

```
useEffect(() => {
  dispatch(actions.fetchInitialData());
}, []);
```

Now that you know what useEffect is for, let's study this code a bit more. This particular useEffect says to run its callback function one time only after render. The dispatch statement tells Redux to fetch all of the initial data.

Hopefully now this code makes more sense and you know why it was included -- we wanted you to be able to work with data from the API from the very start.

Firing off an Ajax call in a component

Eventually, when we our app is finished, PickSeats will display a map of all the seats in the theater and allow the user to reserve the one they want. So clearly we need to know which seats are already reserved.

If we loaded all reservations for all showings, that could be a lot of data. So we won't preload all reservations. Instead, we'll load the PickSeats page, then we'll read the showingId and then we'll fire off a request for Redux to fetch the Showings.

It might look something like this:

```
useEffect(() => {
  dispatch(actions.fetchReservationsForShowing(showingId));
});
```

Hint: Remember, the dispatch won't work until you const dispatch = useDispatch().

2. Go ahead and hardcode a showingId of 1 or 2 or anything that is an actual showing. Try this out but be ready to kill the process because it will throw you into an infinite loop -- the page renders, then it dispatches which causes a re-render, which dispatches, which causes a re-render, and so on and so on.

What we need instead is to say "Watch showingId. When that changes, go fetch the reservations for it." We need useEffect to depend on showingId.

Here's how we'll do that; add an array with showingId in it.

```
// Once and only once, start the fetch to get all reservations for this showing
useEffect(() => {
  dispatch(actions.fetchReservationsForShowing(showingId));
}, [showingId]);
```

3. Go ahead and implement that.
4. Run and test. Feel free to put in a console.log() or two if that will help because you won't see any change to the UI yet.