Actions Dispatched. Does this Thing Work?

In the last lesson, we dispatched our onClick action to the reducer. Let's check the reducer's console log to see if it receives the action.

Firstly, here’s a short quiz question.

Q

Upon clicking a button and consequently dispatching an action, what happens next within Redux? i.e which

Redux actors come into play?

COMPLETED 0%

1 of 1

So the answer is the reducer. You can find the explanation up above, but to prove this further, I’ll log whatever action comes into the reducer.

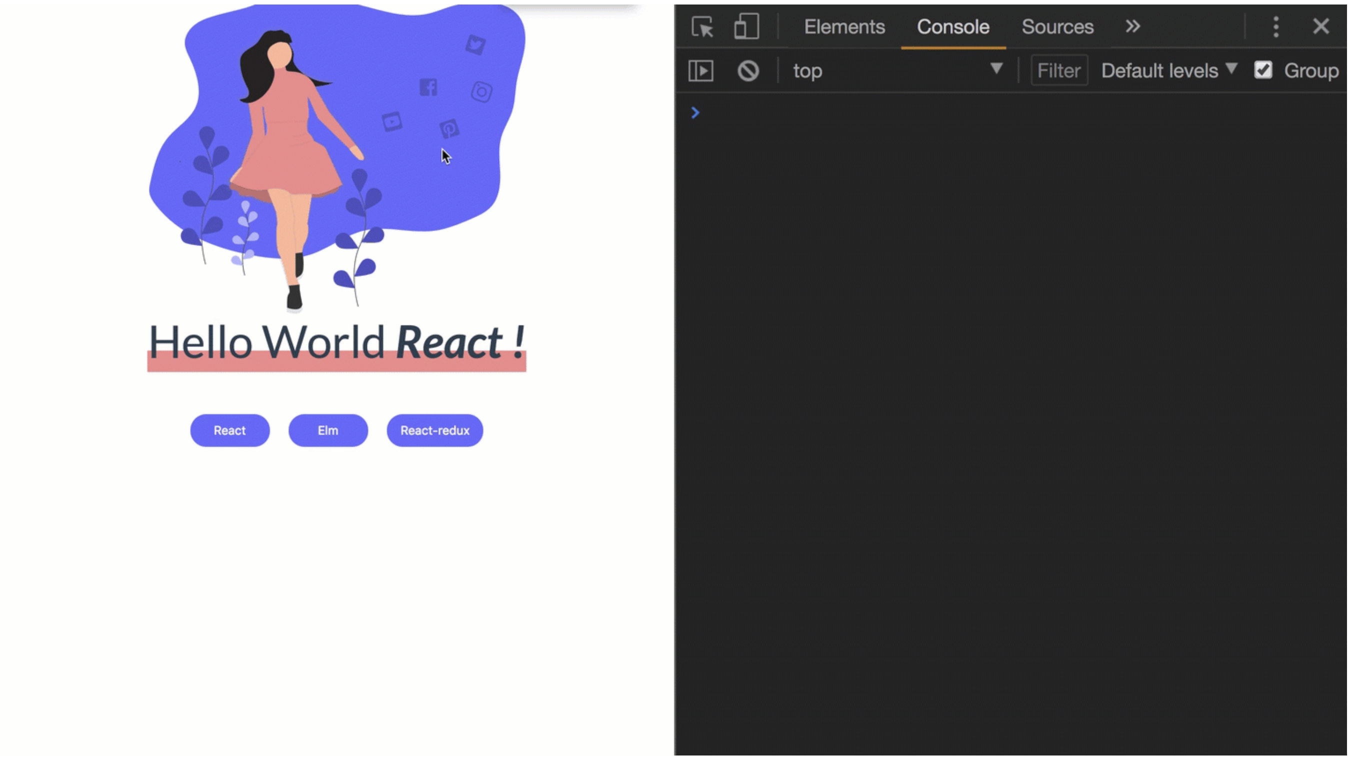
reducers/index.js:

```
export default (state, action) => {
  console.log(action);
  return state;
};
```

reducers/index.js

The reducer then returns the new state of the app. In our particular case, we're just returning the same initial **state**.

With the **console.log()** in the reducer, let's have a look at what happens when we click.



Oh, yeah! The actions are logged when the buttons are clicked. Which proves that the actions indeed go through the Reducer. Amazing!

There's one more thing though. As soon as the app starts, there's a weird action being logged as well. It looks like this:

```
{type: "@@redux/INITu.r.5.b.c"}
```

What's that?

Well, do not concern yourself so much about that. It is an action passed by Redux itself when setting up your app. It is usually called the Redux init action, and it is passed into the reducer when Redux initialises your application with the initial state of the app.

Now, we are sure that the actions indeed pass through the Reducer. Great!

While that's exciting, the only reason you go to the Cashier with a withdrawal request is because you want money. If the Reducer isn’t taking the action we pass in and doing something with our action, of what value is it?

In the next lesson, we’ll learn how to make effective use of the Reducer.