

Redux

Questions

- What is Redux?
- Where does it come from?
- What problem does it solve?
- When should I use Redux?
- What are the three principals?
- What is state?
- What is an action?
- What property must all actions have?
- What is a reducer?
- What is the store?

What is Redux?

- "a database for your React front-end"
- a pattern for state management

Patterns:

- don't mutate state (no reassignment)
- you use functions that return new values

Where does it come from?

- "Mother Russia"
- Dan Abromov

What problem does it solve?

- managing state in *really* big React apps
- when state is shared by several descendant components

When should I use Redux?

- "only if you have to"
- global state shared by several/many descendant components that also need to modify state

What are the three principals?

- state is data you never modify directly
- you can only cause state to change by issuing a "change order" (a.k.a. "sending an action")
- state is updated using "pure functions"

What is state?

- it's just an object
- the keys are labels for the information in your app
- the values are the values you use in your app

What is an action?

- In Redux terminology, is our "change order"
- a description of what to change and how to change it

An example of an action:

```
{
  type: 'WITHDRAWAL',
  amount: 100,
  accountId: 12345
}

{
  type: 'DEPOSIT',
  amount: 1000000000000000000,
  accountId: 12345
}
```

What property must all actions have?

```
{  
  type: 'INCREMENT'  
}  
  
{  
  type: 'DECREMENT'  
}
```

What is a reducer?

- a (pure) function
- accepts 2 arguments: the current state, an action that says what change to make
- returns the new version of the state
- "the bank teller"

What is the store?

- it's a Redux thing
- Redux gives you a function to create a store
- you give it a pure function that it uses to calculate the next version of the state