Training 01
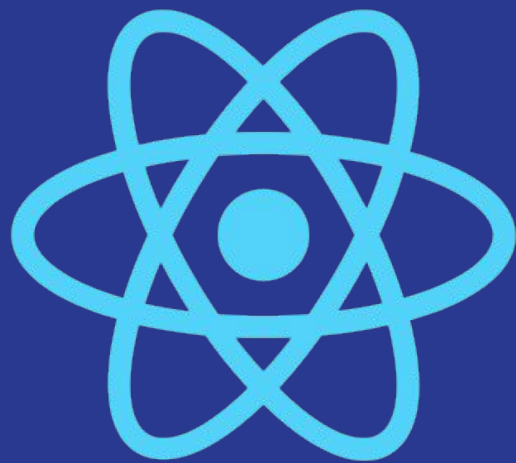
Getting Started

Why we love React

- Functional Programming is Awesome

- Opinionated, but not Dictatorial

- JSX lets you mix JavaScript and HTML
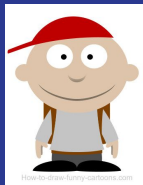
- Reusability

What we'll cover

- Components: Class Based vs Functional

- State vs Props

- Handling Actions

Let's Code!

Component Lifecycle

	Mounting	Updating	Unmounting

"Render Phase"

Pure and has no side effects. May be paused, aborted or restarted by React.
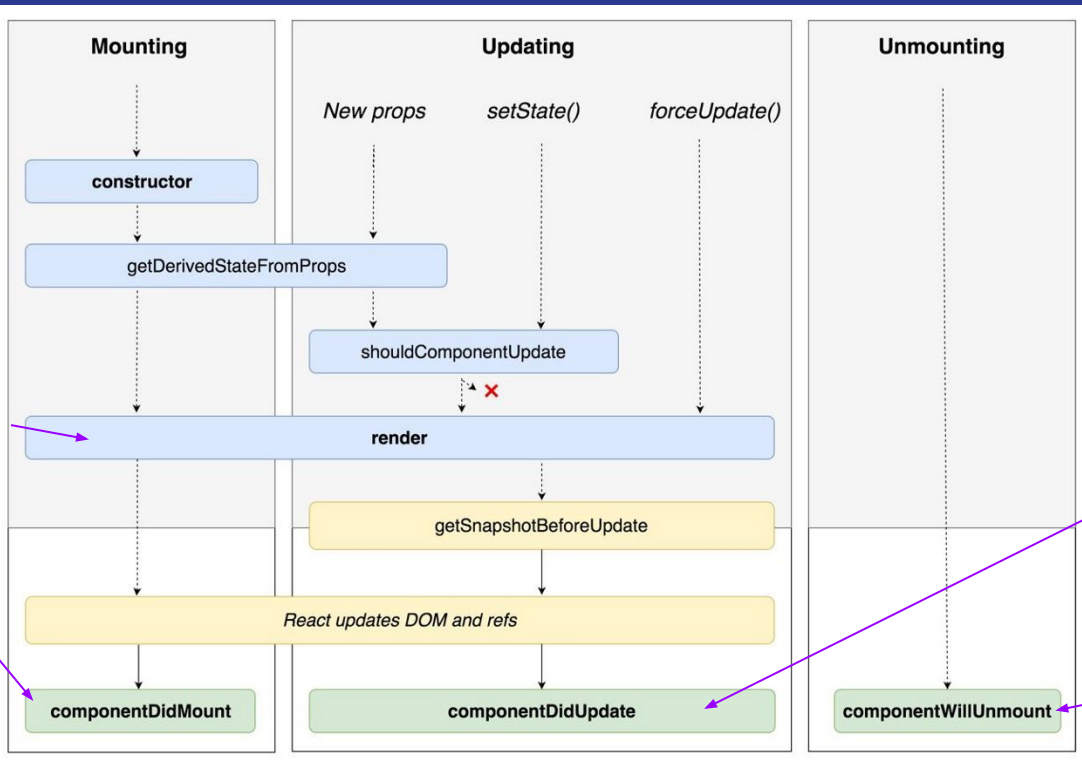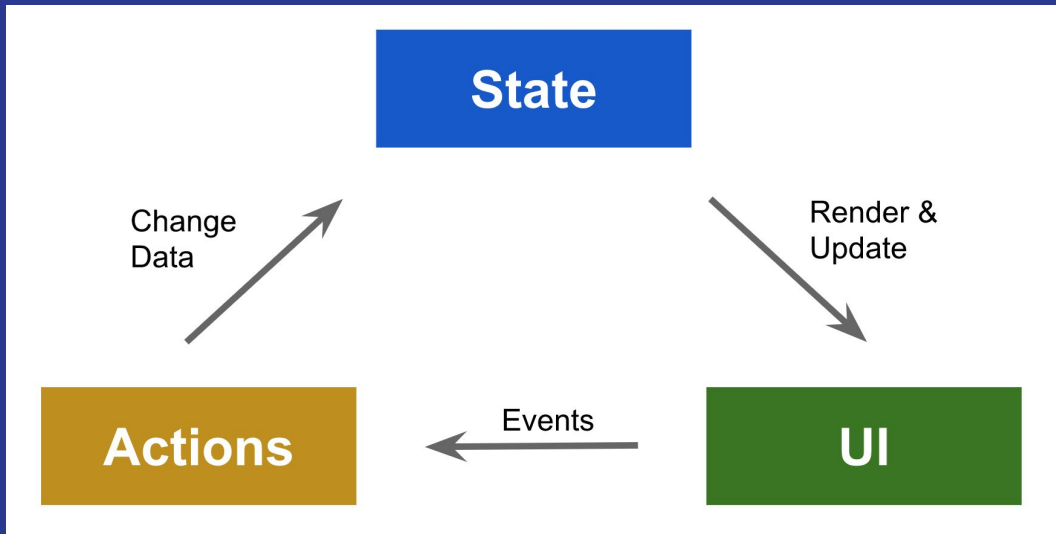
Don't setState here!

- - - - - - - - - - - -

"Pre-Commit Phase"

Can read the DOM.

Good place for initial API calls

"Commit Phase"

Can work with DOM, run side effects, schedule updates.

Mounting:
- constructor
- getDerivedStateFromProps
- render
- componentDidMount

Updating:
- New props → getDerivedStateFromProps
- setState() / forceUpdate() → shouldComponentUpdate ✕
- render
- getSnapshotBeforeUpdate
- React updates DOM and refs
- componentDidUpdate

Unmounting:
- componentWillUnmount

Watch out for infinite setState by loops comparing previous and current state/props!

Put clean up actions here

State & Events

State Management Tips:

- Don't duplicate state!
  - Each piece of state should have only one owner
  - Single source of truth
- State shared by multiple components, need to have a common ancestor!

"Data Down, Events Up"

Review

- What's the difference between state and props?

- What's the difference between a Class Component and a Functional Component?

- Can you name a few lifecycle methods?