REACT FOR ANGULAR DEVELOPERS

Created by Abdelrhman Safwat

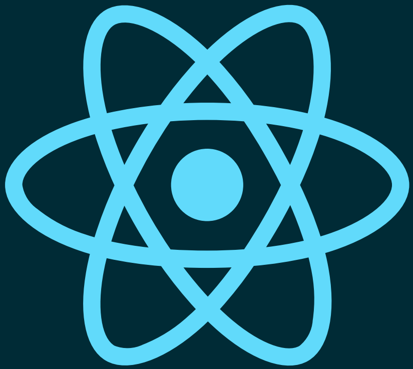
AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

AGENDA

- What is React?
    - Why React?
    - What is JSX?
    - Components and Rendering in React
    - Component Props
    - Component Lifecycle

WHAT IS REACT?



REACT

REACT

React is A JavaScript library for building and rendering user interfaces

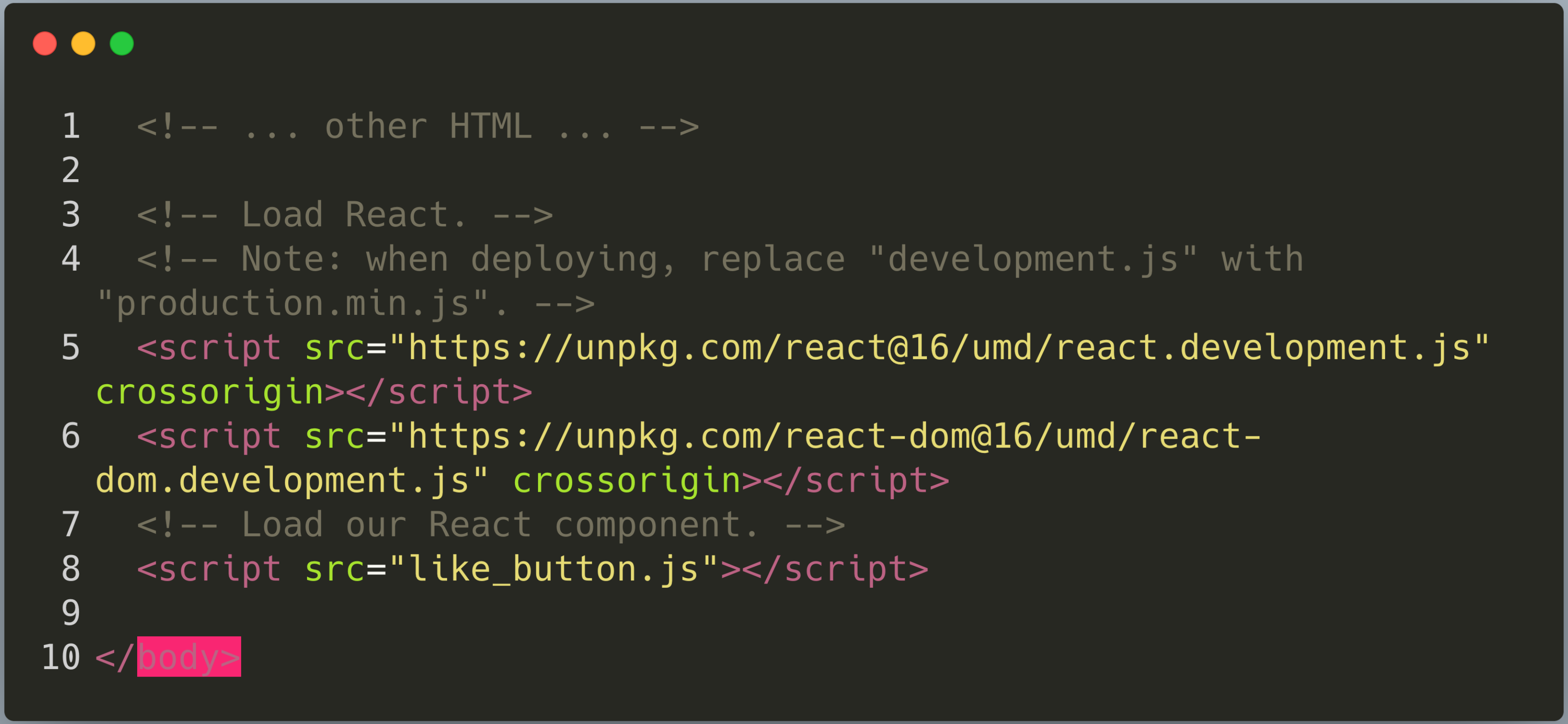
REACT

React is A JavaScript library for building and rendering user interfaces

```
import React from 'react';
const Hello = ({name}) => (
<h1>Hello {name}</h1>
<p>
  Welcome to React, A UI component library
  for deterministic view renders
</p>
);
```

WHAT IS THE DIFFERENCE BETWEEN LIBRARY AND FRAMEWORK?

Example of a library: React



```html
  <!-- ... other HTML ... -->

  <!-- Load React. -->
  <!-- Note: when deploying, replace "development.js" with "production.min.js". -->
  <script src="https://unpkg.com/react@16/umd/react.development.js" crossorigin></script>
  <script src="https://unpkg.com/react-dom@16/umd/react-dom.development.js" crossorigin></script>
  <!-- Load our React component. -->
  <script src="like_button.js"></script>

</body>
```

Example of a framework: Angular

```
    {
"name": "angular",
"version": "0.0.0",
"private": true,
"scripts": {
"ng": "ng",
"start": "ng serve",
"build": "ng build",
"test": "ng test",
"lint": "ng lint",
"e2e": "ng e2e"
},
"dependencies": {
"rxjs": "6.5.5",
"tslib": "1.13.0",
"zone.js": "0.10.3"
```

Example of a framework: Angular

```
"@angular/core": "9.1.12",
"@angular/forms": "9.1.12",
"@angular/common": "9.1.12",
"@angular/router": "9.1.12",
"jasmine-marbles": "0.6.0",
"@angular/compiler": "9.1.12",
"@angular/animations": "9.1.12",
"@angular/platform-browser": "9.1.12",
"angular-in-memory-web-api": "0.11.0",
"@angular/platform-browser-dynamic": "9.1.12"
},
"devDependencies": {
"@angular-devkit/build-angular": "~0.901.1",
"@angular/cli": "~9.1.1",
"@angular/compiler-cli": "~9.1.1",
"@angular/language-service": "~9.1.1",
```

Example of a framework: Angular

```
"@angular/platform-browser": "9.1.12",
"angular-in-memory-web-api": "0.11.0",
"@angular/platform-browser-dynamic": "9.1.12"
},
"devDependencies": {
"@angular-devkit/build-angular": "~0.901.1",
"@angular/cli": "~9.1.1",
"@angular/compiler-cli": "~9.1.1",
"@angular/language-service": "~9.1.1",
"@types/node": "^12.11.1",
"@types/jasmine": "~3.5.0",
"@types/jasminewd2": "~2.0.3",
"codelyzer": "^5.1.2",
"jasmine-core": "~3.5.0",
"jasmine-spec-reporter": "~4.2.1",
"karma": "~4.4.1"
```

Example of a framework: Angular

```
"build": "ng build",
"test": "ng test",
"lint": "ng lint",
"e2e": "ng e2e"
},
"dependencies": {
"rxjs": "6.5.5",
"tslib": "1.13.0",
"zone.js": "0.10.3",
"jasmine-core": "3.5.0",
"@angular/core": "9.1.12",
"@angular/forms": "9.1.12",
"@angular/common": "9.1.12",
"@angular/router": "9.1.12",
"jasmine-marbles": "0.6.0",
"@angular/compiler": "9.1.12"
```

CREATE REACT APP

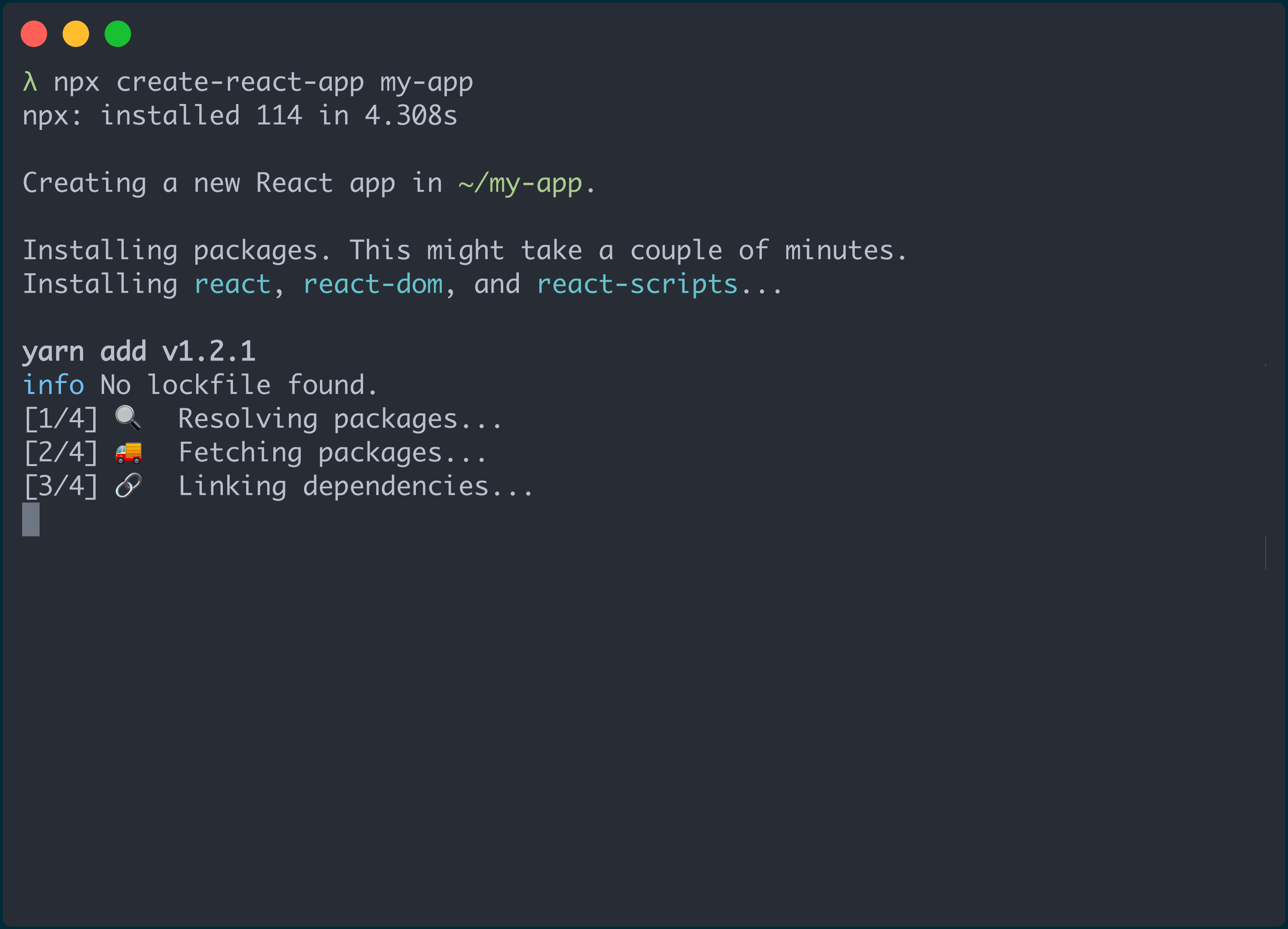
CREATE REACT APP

CREATE REACT APP

Is an opininated collection of libraries to create single-page React applications.

CREATE REACT APP

Is an opininated collection of libraries to create single-page React applications.



```
λ npx create-react-app my-app
npx: installed 114 in 4.308s

Creating a new React app in ~/my-app.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts...

yarn add v1.2.1
info No lockfile found.
[1/4] 🔍  Resolving packages...
[2/4] 🚚  Fetching packages...
[3/4] 🔗  Linking dependencies...
```

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

AGENDA

- What is React?
- **Why React?**
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

UNIDIRECTIONAL DATA FLOW/BINDING

UNIDIRECTIONAL DATA FLOW/BINDING

↓

UNIDIRECTIONAL DATA FLOW/BINDING

Life is simpler when UI components are unaware of the network, business logic, or app state. Given the same props, always render the same data.

COMPONENT STATE IS IMMUTABLE

COMPONENT STATE IS IMMUTABLE

COMPONENT STATE IS IMMUTABLE

Once set, the state of a component can’t be changed. State changes don’t change existing view state. Instead, they trigger a new view render with a new state.

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

JSX

```
const element = <h1>Hello, world!</h1>;
```

JSX

```
const ItemList = ({ items }) => (
<ul>
  {items.map((item) => (
  <li key={item.id}>
    <div>{item.name}</div>
  </li>
  ))}
</ul>
);
```

JSX

```
const element = (
<h1 className="greeting">
  Hello, world!
</h1>
```

```
const element = React.createElement(
'h1',
{className: 'greeting'},
'Hello, world!'
);
```

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

COMPONENTS AND RENDERING IN REACT

```
 const element = <h1>Hello, world</h1>;
ReactDOM.render(element, document.getElementById('root'));
```

COMPONENTS AND RENDERING IN REACT

```
 const element = <h1>Hello, world</h1>;
ReactDOM.render(element, document.getElementById('root'));
```

Try it out here

COMPONENTS AND RENDERING IN REACT

```
function tick() {
const element = (
<div>
  <h1>Hello, world!</h1>
  <h2>It is {new Date().toLocaleTimeString()}.</h2>
</div>
);
ReactDOM.render(element, document.getElementById('root'));
}

setInterval(tick, 1000);
```

COMPONENTS AND RENDERING IN REACT

```
function tick() {
const element = (
<div>
  <h1>Hello, world!</h1>
  <h2>It is {new Date().toLocaleTimeString()}.</h2>
</div>
);
ReactDOM.render(element, document.getElementById('root'));
}

setInterval(tick, 1000);
```

Try it out here

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

COMPONENT PROPS

```
function Welcome(props) {
return <h1>Hello, {props.name}</h1>;
}

function App() {
return (
<div>
  <Welcome name="Sara" />
  <Welcome name="Cahal" />
  <Welcome name="Edite" />
</div>
);
}
ReactDOM.render(
<App />,document.getElementById('root')
);
```

Try it out here

COMPONENT PROPS

```
function Welcome(props) {
return <h1>Hello, {props.name}</h1>;
}

function App() {
return (
<div>
  <Welcome name="Sara" />
  <Welcome name="Cahal" />
  <Welcome name="Edite" />
</div>
);
}
ReactDOM.render(
<App />,document.getElementById('root')
);
```

Try it out here

COMPONENT PROPS

```
function Welcome(props) {
return <h1>Hello, {props.name}</h1>;
}

function App() {
return (
<div>
  <Welcome name="Sara" />
  <Welcome name="Cahal" />
  <Welcome name="Edite" />
</div>
);
}
ReactDOM.render(
<App />,document.getElementById('root')
);
```

Try it out here

⚠ PROPS ARE READ-ONLY

⚠ ALL REACT COMPONENTS MUST ACT LIKE PUR
FUNCTIONS WITH RESPECT TO THEIR PROPS.

AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

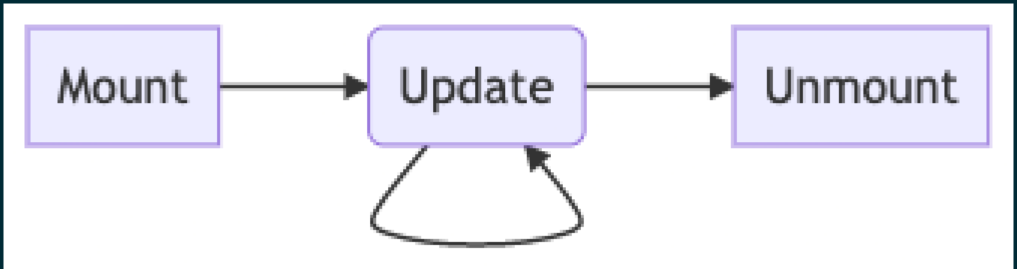
AGENDA

- What is React?
- Why React?
- What is JSX?
- Components and Rendering in React
- Component Props
- Component Lifecycle

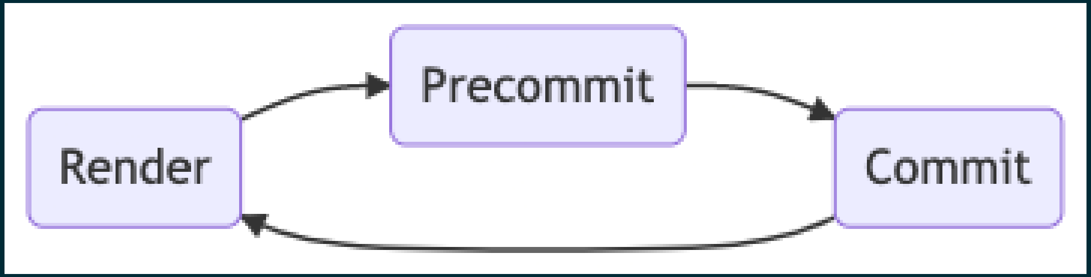
COMPONENT LIFECYCLE

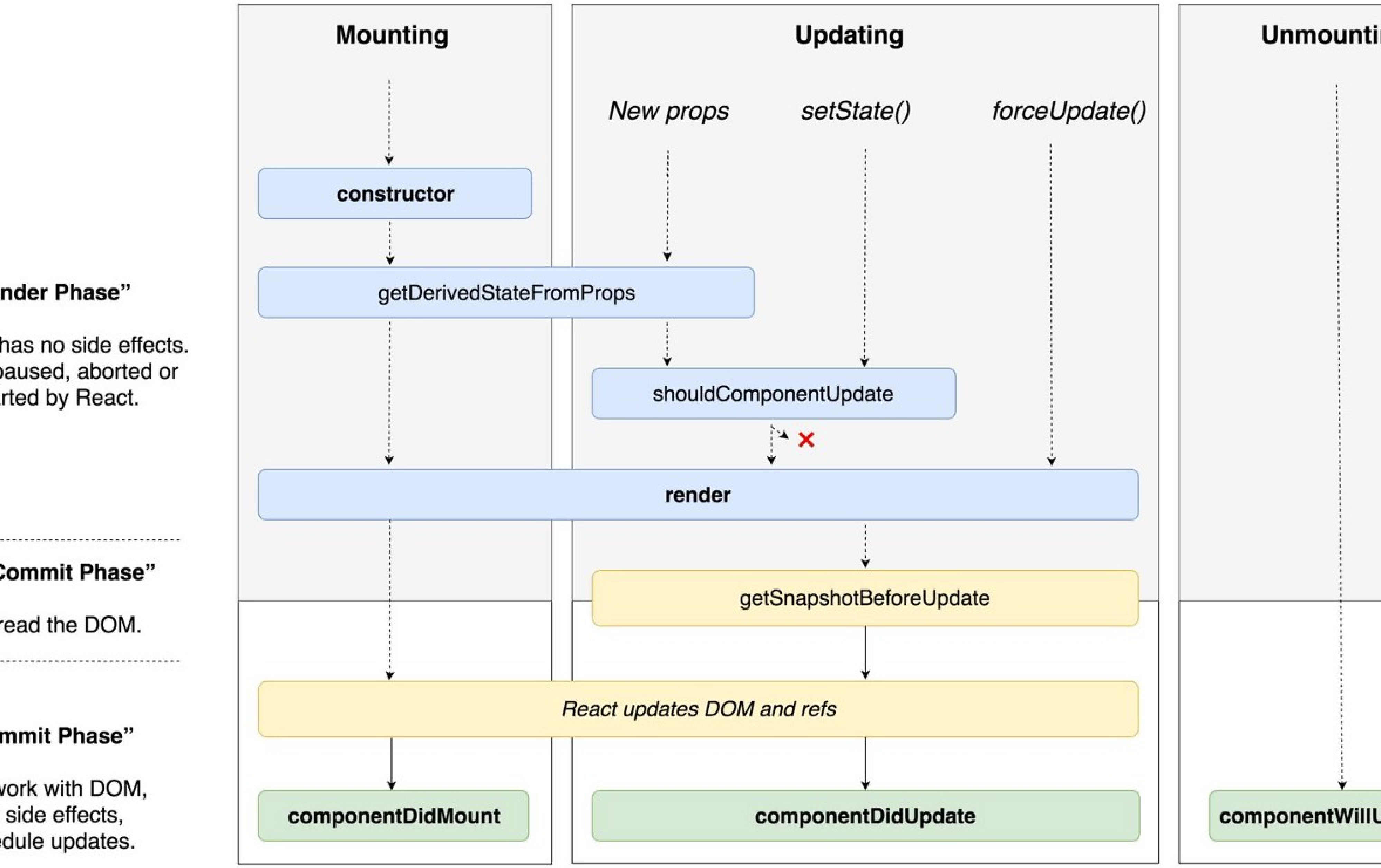
COMPONENT LIFECYCLE

COMPONENT LIFECYCLE



COMPONENT LIFECYCLE





nder Phase"

has no side effects.
paused, aborted or
rted by React.

Commit Phase"

read the DOM.

mmit Phase"

work with DOM,
side effects,
edule updates.

THE END

Refrences:

- React official documentation
- Create React App
- The missing Introduction to React
- 10 Tips for Better Redux Architecture
- 10 Composing Software: The Book
- React Component Patterns
- React book