



INTRODUCING REACT

QUICK INTRODUCTION ON WHY
REACT IS DIFFERENT

WHAT IS REACT?

- DEVELOPED AND MAINTAINED BY FACEBOOK AND INSTAGRAM
- JAVASCRIPT LIBRARY TO BUILD USER INTERFACES
- REPRESENT THE V OF THE MVC PATTERN
- BASED ON COMPONENTS
- CAN BE PARTIALLY USED ON AN EXISTING STACK
- REACT NATIVE FOR MOBILE APPS.

BASED ON COMPONENTS

- COMPONENTS ARE INDEPENDENT, REUSABLE PIECES.
- EACH PIECE IS IN ISOLATION.
- CONTAINER > TABLE > TITLES & PRODUCTS
- > SEARCH
- → 5 COMPONENTS

The image shows a UI component layout with the following elements:

- A search bar with the placeholder text "Search..."
- A checkbox labeled "Only show products in stock"
- A table with two columns: "Name" and "Price".

The table is divided into two sections:

- Sporting Goods**
 - Football \$49.99
 - Baseball \$9.99
 - Basketball \$29.99
- Electronics**
 - iPod Touch \$99.99
 - iPhone 5 \$399.99
 - Nexus 7 \$199.99

CREATED WITH JSX

- JSX IS A JAVASCRIPT PREPROCESSOR
- ADDS XML SYNTAX TO JAVASCRIPT
- IT'S NOT HTML OR JAVASCRIPT SYNTAX,
IT'S JSX

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


A COMPONENT HAVE METHODS

- EACH COMPONENT HAVE 15 METHODS

- MOUNTING
- UNMOUNTING
- UPDATING



LIFECYCLE

- MANDATORY:
 - RENDER()

```
// Create a component named MessageComponent
var MessageComponent = React.createClass({
  render: function() {
    return (
      <div>{this.props.message}</div>
    );
  }
});

// Render an instance of MessageComponent into document.body
ReactDOM.render(
  <MessageComponent message="Hello!" />,
  document.body
);
```

A COMPONENT HAVE PROPS

- PROPS ARE READ-ONLY
- CAN BE COMPARED TO DEPENDENCY INJECTION
- PROPS POPULATED BY OWNER COMPONENT
- TOP-DOWN DATA FLOW

```
// Create a component named MessageComponent
var MessageComponent = React.createClass({
  render: function() {
    return (
      <div>{this.props.message}</div>
    );
  }
});

// Render an instance of MessageComponent into document.body
ReactDOM.render(
  <MessageComponent message="Hello!" />,
  document.body
);
```

A red arrow points from the `message="Hello!"` prop in the parent component's JSX to the `this.props.message` access in the child component's render function. The parent's prop is circled in red.

A COMPONENT HAVE A STATE

- STATE IS PRIVATE AND FULLY CONTROLLED BY COMPONENT
- UPDATING STATE RE-RENDERS THE COMPONENT
- IF IT'S NOT USED IN RENDER(), IT SHOULDN'T BE IN THE STATE

```
class Clock extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {date: new Date()};  
  }  
  
  componentDidMount() {  
    this.timerID = setInterval(  
      () => this.tick(),  
      1000  
    );  
  }  
  
  componentWillUnmount() {  
    clearInterval(this.timerID);  
  }  
  
  tick() {  
    this.setState({  
      date: new Date()  
    });  
  }  
  
  render() {  
    return (  
      <div>  
        <h1>Hello, world!</h1>  
        <h2>It is {this.state.date.toLocaleTimeString()}.</h2>  
      </div>  
    );  
  }  
}
```



CODE EXAMPLE

- RECAT: AN APP TO LOAD CAT PICTURES USING REACT AND A THIRD PARTY API
- URL:
[HTTPS://GITHUB.COM/GRENOULT/RECAT](https://github.com/grenoult/recat)

