Table of Contents

[​ Create React App 2](#__RefHeading___Toc278_4021959227)

[​ React JSX 3](#__RefHeading___Toc280_4021959227)

[​ React Components 3](#__RefHeading___Toc282_4021959227)

React Documentation

**React** is a JavaScript library for rendering user interfaces (UI). React applications are built from isolated pieces of UI called components. A React component is a JavaScript function that you can sprinkle with markup. Components can be as small as a button, or as large as an entire page.

* React is a JavaScript library for building user interfaces.
* React is used to build single-page applications.
* React allows us to create reusable UI components.

# Create React App

For creating a react applications, [Node.js](https://nodejs.org/) is required.

Open the terminal in the directory we would like to create our application.

Run this command to create a React application named my-react-app

* npx create-react-app my-react-app

create-react-app will set up everything you need to run a React application.

Run this command to move to the my-react-app directory.

* cd my-react-app

Run this command to run the React application my-react-app

* npm start

Arrow Function Syntax

hello = (val) => "Hello " + val;

Ternary Operator

authenticated ? renderApp() : renderLogin();

# **React JSX**

Use attribute className instead of class in JSX.

React supports if statements, but not inside JSX. Use ternary expressions instead.

# **React Components**

Components are like functions that return HTML elements. There are two types of components like as class component and function component. The component also requires a render() method.

A class component must include the extends React.Component statement. This statement creates an inheritance to React.Component, and gives our component access to React.Component's functions.

**Example of Class Component.**  
  
class Car extends React.Component {

render() {

return <h2>Hi, I am a Car!</h2>;

}

}

**Example of function component.**

function Car() {

return <h2>Hi, I am a Car!</h2>;

}

class Car extends React.Component {

constructor(props) {

super(props);

this.state = {brand: "Ford"};

}

# React Props

Props are like function arguments and we send them into the component as attributes.

props stands for properties.