

Introduction to React.js

What is React, JSX, Overview and Syntax



SoftUni Team
Technical Trainers



SoftUni



Software University

<https://softuni.bg>

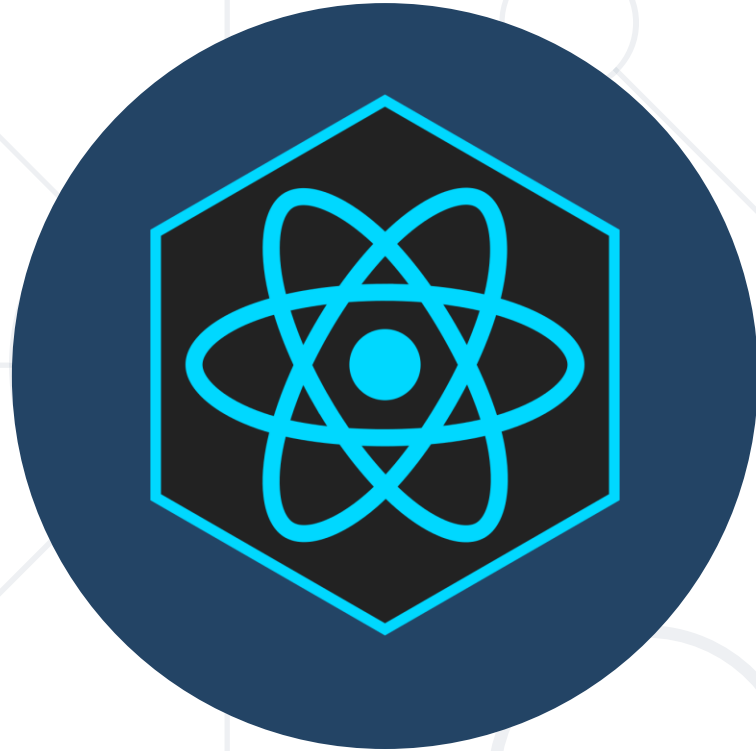
Table of Contents

1. React Overview
2. JSX Syntax
3. Installation
4. Composition



sli.do

#reactjs



React Overview

History and Philosophy

What is React.js?

- **React** is a JavaScript library for building **user interfaces** (UI)
- Focused on creating **reusable components**
- Developed by **Facebook**

```
const HelloMessage = (props) => (  
  <div>Hello {props.name});  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
  
root.render(  
  <HelloMessage name="Maria" />  
);
```

- Open-source
- Declarative
 - Design **simple** views for each **state** in your app
 - Easier to **debug**
- Component-Based
 - Encapsulated **components** that manage their **own** state
 - Keep **state** out of the **DOM**

-
- A stylized illustration of a hand holding a black smartphone. The phone's screen is blue and displays a white outline of a person's head and shoulders. Surrounding the phone are several colorful icons: an orange speech bubble with three dots, a green square with a white smartphone icon, a blue speech bubble with a white speech mark, a red speech bubble with a white camera icon, a green square with a white music note icon, a blue square with a white globe icon, a red square with a white heart icon, a blue square with a white shopping cart icon, a green square with a white checkmark icon, and a red square with a white cloud icon. The background is white with faint grey lines.

- Easy to learn
- Fast **performance**
- Use all **ES6** features
 - **Promises, Classes** and **Modules**
- Compatible with other **libraries**
- Great **error reporting**





JSX Syntax

Overview, Syntax, Advantages

- **JSX** is React's JavaScript **superset language**
 - Has all of JavaScript's **features** and more
- Unique approach to **mixing HTML and JS**
- Compiles to **plain JavaScript**

```
<div className="red">Children Text</div>
```

```
React.createElement('div',  
  { className: 'red' },  
  'Children Text'  
);
```

- Standard elements use lowercase names
 - **div, span, form, input, ...**
- Custom components **always** use Pascal case
 - **MyCustomComponent, Greeting, ScoreBoard, ...**
- Component name cannot be an expression
 - Use a **variable** instead
- There must be a **root element**
- More info at: <https://react.dev/learn/writing-markup-with-jsx>

- JSX **compiles** to function calls

```
<div className="red">Children Text</div>
```

Element type (HTML tag name)

```
React.createElement('div',  
  { className: 'red' },  
  "Children Text" [, ...]);
```

Properties object

List of children



Live Demo



React Installation

Packages, Setup, Structure

- Instant Feedback - **rapid reloading** for efficient development
- Only one dependency - no complicated version mismatches
- No Lock-In - under the hood **Webpack, Babel, ESLint**



Install and Run Vite

- Run the React app creator

```
npm create vite my-app
```

- Select Framework >> **React**

- Select Variant >> **JavaScript**

- Starts your React app from the command line

```
cd my-app
```

```
npm install
```

```
npm run dev
```

- Browse your app from <http://localhost:5173>

- Visit the **official website**

<https://reactjs.org/>

- Documentation

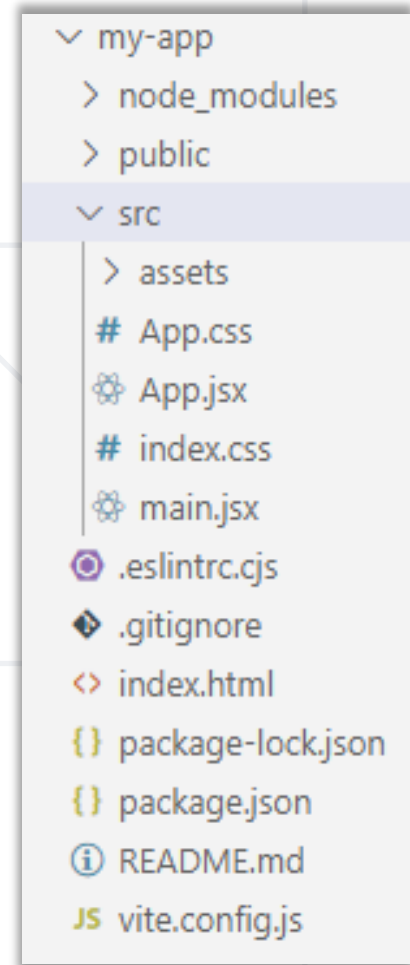
<https://react.dev/learn/installation>

- Online sandbox

<https://codesandbox.io/>

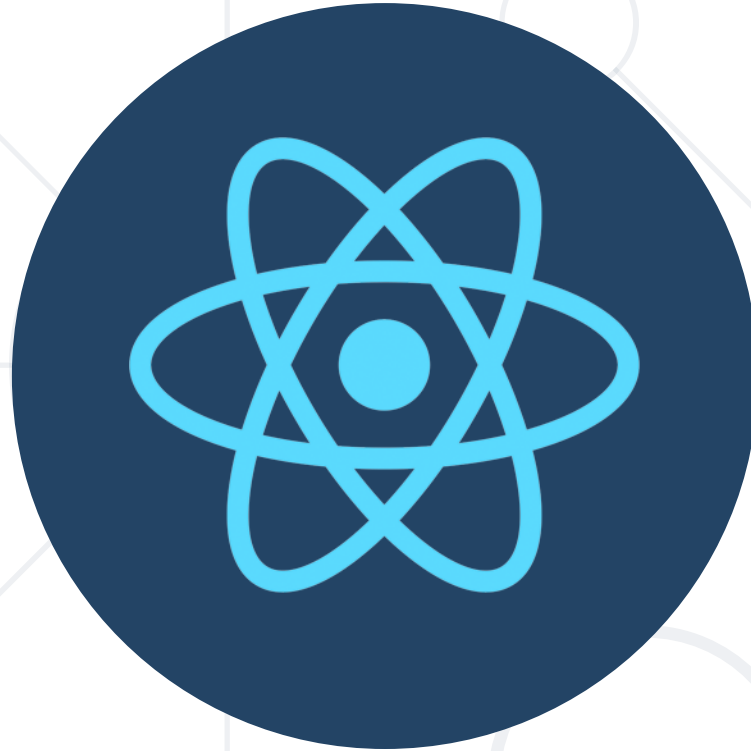
React App Structure

- **package.json** - project configuration
 - Module name, dependencies, build actions
- **index.html**
 - App main HTML file
- **main.jsx**
 - App main JS file (startup script)
- **App.jsx, App.css,**
- React component "**App**"





Live Demo



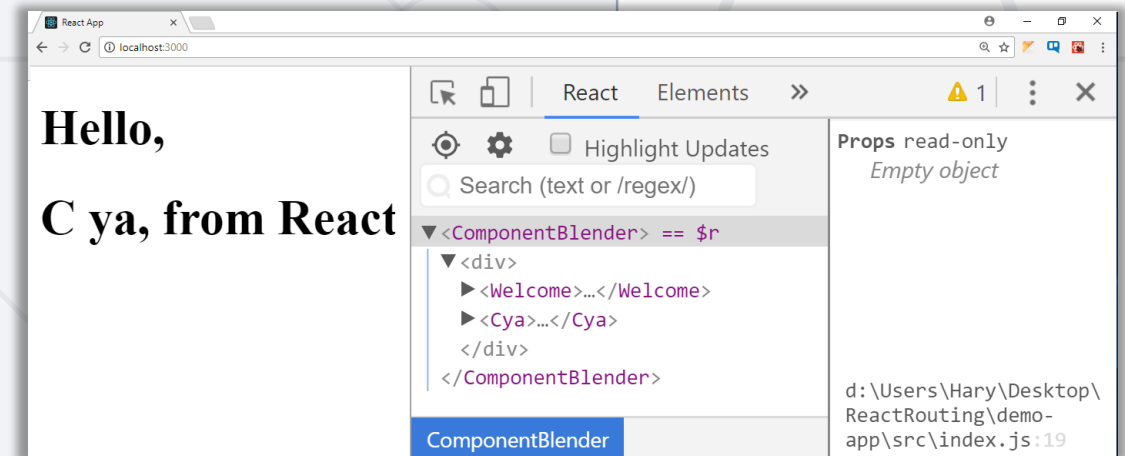
Composition

Definition and Advantages

- React components can be **nested**, like DOM elements

```
function Welcome() {  
  return <h1>Hello, from React</h1>;  
}  
function Cya() {  
  return <h1>C ya, from React</h1>;  
}  
function ComponentBlender() {  
  return (  
    <div>  
      <Welcome />  
      <Cya />  
    </div>  
  );  
}
```

```
ReactDOM.createRoot(  
  document.getElementById('root')  
)  
.render(<ComponentBlender />);
```



- Names always start with **uppercase**
- Tags must be **closed**
 - If there are no children - use **self-closing tags**
- **Information** is passed via **props**

```
<Heading size="large">Hello React</Heading>
```

```
<UserHead name="homeHeader" />
```

- Encapsulate logic
- **Separate** your code
 - Easier to **maintain** and **debug**
 - Allows **reusability**
- Components are neat



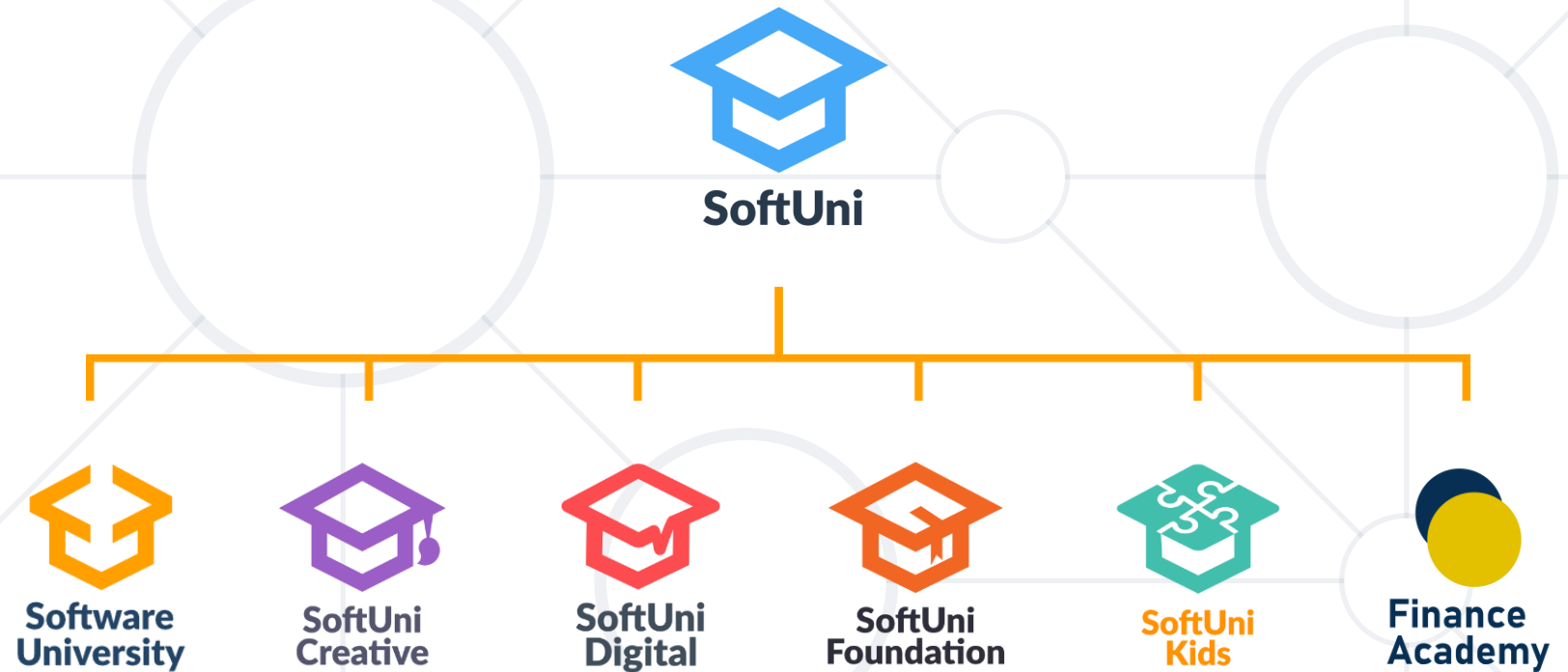


Live Demo

- **React** is a JavaScript library for building user interfaces
- **React** uses all **ES6** features
- **JSX** is React's JavaScript **superset**
- React **components** can be nested (composition)



Questions?



SoftUni Diamond Partners

**SUPER
HOSTING
.BG**



**Coca-Cola HBC
Bulgaria**

 **Flutter**TM
International

INDEAVR
Serving the high achievers



AMBITIONED

 **DRAFT
KINGS**



BOSCH

 **Postbank**
Решения за твоето утре

 **PHAR
VISION**



SmartIT

DXC
TECHNOLOGY

createX

- Software University – High-Quality Education, Profession and Job for Software Developers

- softuni.bg, about.softuni.bg

- Software University Foundation

- softuni.foundation

- Software University @ Facebook

- facebook.com/SoftwareUniversity

- Software University Forums

- forum.softuni.bg



- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – <https://about.softuni.bg>
- © Software University – <https://softuni.bg>

