

A large, faded, dark gray React.js logo is visible in the background on the left side of the image. It features a central circle with six triangles pointing towards it, all enclosed within a hexagonal border.

React.js

React Installation

Install node.js

Ubuntu

```
curl -sL https://deb.nodesource.com/setup_10.x | sudo -E bash -  
sudo apt-get install -y nodejs
```

Other

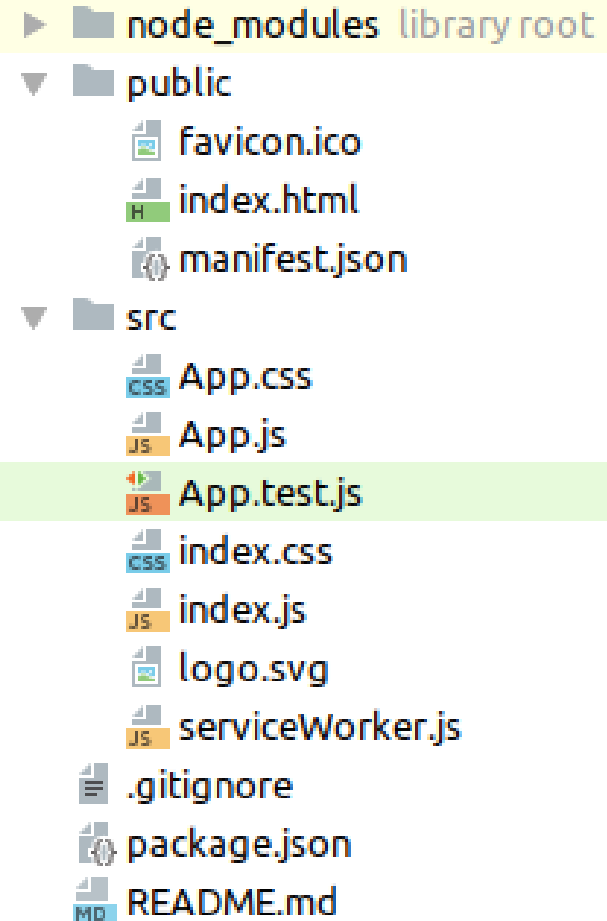
<https://nodejs.org/en/download/package-manager/>

Install Create React App

```
sudo npm install --global create-react-app
```

Start first react app

```
create-react-app lesson1  
cd lesson1  
npm start
```



JSX syntax

```
const navBar = <nav>I am a nav bar</nav>;
```

```
const myTeam = {  
  center: <li>Benzo Walli</li>,  
  powerForward: <li>Rasha Loa</li>,  
  smallForward: <li>Tayshaun Dasmoto</li>,  
  shootingGuard: <li>Colmar Cumberbatch</li>,  
  pointGuard: <li>Femi Billon</li>  
};
```

JSX syntax

//right

```
const br = <br />;
```

//wrong

```
const br = <br>;
```

```
const theExample = (  
  <a href="https://www.example.com">  
    <h1>  
      Click me!  
    </h1>  
  </a>  
)
```

JSX syntax

```
//wrong
const wrong = (
  
  <h1>
    Welcome to Dan's Blog!
  </h1>
  <article>
    Wow I had the tastiest sandwich today. I <strong>literally</strong>
  </article>
);
```

```
//right
const right = (
  <div>
    
    <h1>
      Welcome to Dan's Blog!
    </h1>
    <article>
      I <strong>literally</strong> almost freaked out.
    </article>
  </div>
);
```

Use JSX in react.js

```
import React from 'react';
import ReactDOM from 'react-dom';

const toDoList = (
  <ol>
    <li>Learn React</li>
    <li>Become a Developer</li>
  </ol>
);

ReactDOM.render(
  toDoList,
  document.getElementById('app')
);
```

React.js

```
import React from 'react';
import ReactDOM from 'react-dom';

class FirstClass extends React.Component {
  render() {
    return <h1>Hello world</h1>;
  }
}

ReactDOM.render(
  <FirstClass />,
  document.getElementById('app')
);
```


React.js

List.js

```
import React from 'react';  
  
class List extends React.Component {}  
  
export default List;
```

// or

```
import React from 'react';  
  
export default class List extends React.Component {}
```

React.js

// App.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import List from './list';
```

```
class App extends React.Component {
```

```
  render() {
    return (
      <div>
        <List />
      </div>
    );
  }
}
```

```
ReactDOM.render(
  <App />,
  document.getElementById('app')
);
```

// List.js

```
import React from 'react';
```

```
export default class List extends React.Component {
```

```
  render() {
    return (
      <ol className="itemsList">
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
        <li>Item 4</li>
      </ol>
    );
  }
}
```

Prop Types

```
npm install --save prop-types
```

<https://reactjs.org/docs/typechecking-with-proptypes.html#proptypes>

```
import PropTypes from 'prop-types';

MyComponent.propTypes = {
  // You can declare that a prop is a specific JS type. By default, these
  // are all optional.
  optionalArray: PropTypes.array,
  optionalBool: PropTypes.bool,
  optionalFunc: PropTypes.func,
  optionalNumber: PropTypes.number,
  optionalObject: PropTypes.object,
  optionalString: PropTypes.string,
  optionalSymbol: PropTypes.symbol,

  // Anything that can be rendered: numbers, strings, elements or an array
  // (or fragment) containing these types.
  optionalNode: PropTypes.node,

  // A React element.
  optionalElement: PropTypes.element,

  //.....
}
```

Prop Types

```
import React from 'react';
import PropTypes from 'prop-types';

export default class List extends React.Component {
  static propTypes = {
    className: PropTypes.string.isRequired,
    listFirstItem: PropTypes.string
  }
  render() {
    return (
      <ol className={this.props.className}>
        <li>{this.props.listFirstItem || 'Item 1'}</li>
        <li>Item 2</li>
        <li>Item 3</li>
        <li>Item 4</li>
      </ol>
    );
  }
}
```

```
import React from 'react';
import PropTypes from 'prop-types';

export default class List extends React.Component {
  static propTypes = {
    className: PropTypes.string.isRequired,
    listFirstItem: PropTypes.string
  }
  static defaultProps = {
    listFirstItem: 'Item 1'
  }
  render() {
    return (
      <ol className={this.props.className}>
        <li>{this.props.listFirstItem}</li>
        <li>Item 2</li>
        <li>Item 3</li>
      </ol>
    );
  }
}
```

React Loop

```
//not recommended
const list = []
for (const item of list) {
  |   list.push(<ol key={item.id}>{item.name}</ol>)
}

```

```
//not recommended
const list = this.props.list.map((item, index) => (
  |   <ol key={index}>{item.name}</ol>
));

```

```
//recommended
const list = this.props.list.map((item, index) => (
  |   <ol key={item.id}>{item.name}</ol>
));

```

React Events

onKeyDown

onKeyPress

onKeyUp

onFocus

onBlur

onChange

onInput

onInvalid

onSubmit

onClick

onContextMenu

onDoubleClick

onDrag

onDragEnd

onDragEnter

onDragExit

onDragLeave

onDragOver

onDragStart

onDrop

onMouseDown

onMouseEnter

onMouseLeave

onMouseMove

onMouseOut

onMouseOver

onMouseUp

....

<https://reactjs.org/docs/events.html>

React state

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

React state

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


React state

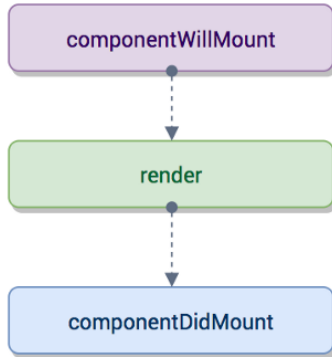
```
//.....  
componentDidMount() {  
  this.timerID = setInterval(  
    () => this.tick(),  
    1000  
  );  
}  
  
componentWillUnmount() {  
  clearInterval(this.timerID);  
}  
  
tick() {  
  this.setState({  
    date: new Date()  
  });  
}  
  
//.....
```

React lifecycle

Initialization

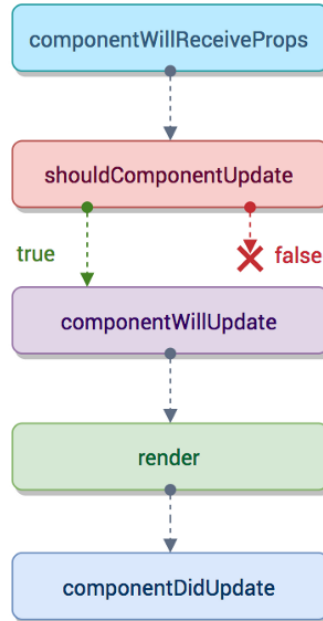
setup props and state

Mounting

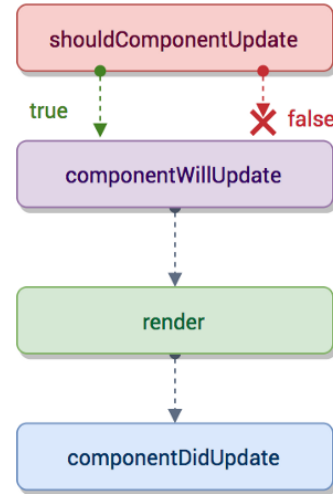


Update

props



states



Unmounting

`componentWillUnmount`