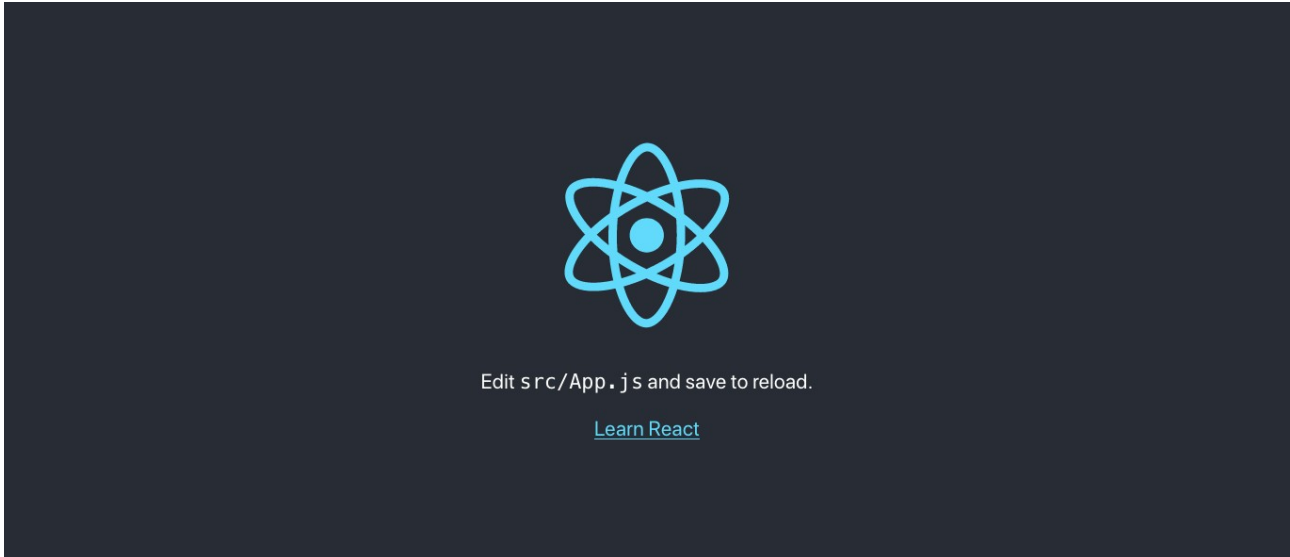


## React Project Create

```
npx create-react-app first_app
```

```
cd first_app  
npm start
```



## React Project Structure

### 0\_install

- first\_app -
  - node\_modules
  - package.json
  - package-lock.json
  - public
  - **src** -
    - App.css
    - App.js
    - index.css
    - index.js
    - reportWebVitals.js
    - components -
      - header.js
      - footer.js

### 1\_hello\_react\_app

- src

### 2\_components

- src

### 3\_props

- src

### 4\_events

- src

### 5\_state

- src

## (5) React State Assignment



Copy 0\_install/first\_app/src folder to 5\_state folder.  
Create components/about.js file.

```
import React, {Component} from 'react';

class AboutUs extends Component {
  state = {
    name: '...'
  }
  setName = evt => {
    this.state.name = evt.target.value;
  }
  clickMe = evt => {
    this.setState({name: this.state.name});
  }

  render() {
    return (
      <div>
        <div>
          <p> {this.state.name} </p>
        </div>
        <div>
          name : <input onChange={this.setName} type="text" />
        </div>
        <button onClick={this.clickMe}> Click </button>
      </div>
    )
  }
}

export default AboutUs;
```

Change App.js file.

```
import './App.css';
import AboutUs from './components/about.js';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <AboutUs />
      </header>
    </div>
  );
}

export default App;
```

```
cd 0_install/first_app
```

Run react server.

```
npm start
```

Create 5\_state/assignment folder.

Copy 0\_install/first\_app/src folder to 5\_state/assignment folder.

## (6) React Life Cycle

Change about.js file.

```
class AboutUs extends Component {
  state = {
  }

  constructor(props) {
    super(props);
    console.log('constructor call');
  }

  componentDidMount() {
    console.log('componentDidMount call');
  }

  componentDidUpdate() {
    console.log('componentDidUpdate call');
  }

  componentWillUpdate() {
    console.log('componentWillUpdate call');
  }

  componentWillUnmount() {
    console.log('componentWillUnmount call');
  }

  render() {
    console.log('render call');
    return (
      ...
    )
  }
}
```

```
cd 0_install/first_app
```

Run react server.

```
npm start
```

Create 6\_life\_cycle folder.

Copy 0\_install/first\_app/src folder to 6\_life\_cycle folder.

## (7) React Conditional

### Example 1

Even Odd Number using conditional.



Create component/even\_odd.js file.

```
class EvenOdd extends Component {
  state = {
    num: 0
  }
  setNum = evt => {
    this.state.num = evt.target.value;
  }
  click = evt => {
    this.setState({num: this.state.num});
  }
  render() {
    return (
      <div>
        <div>
          <input onChange={ this.setNum } type="text" />
        </div>
        <div>
          <button onClick={ this.click }> Click </button>
        </div>
        <div>
          { this.state.num % 2 == 0 ? "Even Number" : "Odd Number"}
        </div>
      </div>
    )
  }
}
export default EvenOdd;
```

Change App.js file.


```
import './App.css';
import EvenOdd from './components/even_odd.js';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <EvenOdd />
      </header>
    </div>
  );
}

export default App;
```

## Example 2

Login and Logout screen using conditional.



A dark-themed login form is displayed on a black background. It consists of two labels, 'username:' and 'password:', each followed by a white rectangular input field. Below the password field is a small, light gray button with the text 'Login' in a dark font.

Copy even\_odd.js and rename component/login.js file.

```

class Login extends Component {
  state = {
    isLogin: false,
    username: "",
  }
  setUser = evt => {
    this.state.username = evt.target.value;
  }
  clickLogin = evt => {
    if (this.state.username == 'kyaw' && this.state.passwd == '123') {
      this.setState({isLogin: true});
    }
  }
  clickLogout = evt => {
    this.setState({isLogin: false});
  }
  render() {
    return (
      <div>
        { this.state.isLogin ? (
          <div>
            <div>
              Welcome home screen !
            </div>
            <div>
              <button onClick={ this.clickLogout }> Logout </button>
            </div>
          </div>
        ) : (
          <div>
            <div>
              username: <input onChange={ this.setUser } type="text" />
            </div>
            <div>
              <button onClick={ this.clickLogin }> Login </button>
            </div>
          </div>
        )
      )
    </div>
  )
}
export default Login;

```

Create 7\_conditional folder.

Copy 0\_install/first\_app/src folder to 7\_conditional folder.

## (8) React Keys

Create component/home.js file.

```
import React, { Component } from 'react';

class Home extends Component {
  render() {
    const employees = ['aung aung', 'mg mg', 'kyaw kyaw', 'aye aye']
    return (
      <div>
        { employees.map(employee => {
          return (
            <div>
              <h1> { employee } </h1>
            </div>
          )
        }) }
      </div>
    )
  }
}

export default Home;
```

Inspect browser console.

**Warning: Each child in a list should have a unique "key" prop.**

Check the render method of `Home`. See <https://reactjs.org/link/warning-keys> for more information.

div

Home@http://localhost:3000/static/js/bundle.js:354:1

header

div

App



Create component/new\_home.js file.

```
import React, { Component } from 'react';

class NewHome extends Component {

  render() {
    const employees = [
      {id: 1, name: 'aung aung'},
      {id: 2, name: 'mg mg'},
      {id: 3, name: 'kyaw kyaw'},
      {id: 4, name: 'aye aye'}
    ]

    return (
      <div>
        { employees.map(employee => {
          return (
            <div key={employee.id}>
              <h1> { employee.name } </h1>
            </div>
          )
        })
        }
      </div>
    )
  }
}

export default NewHome;
```

**aung aung**

**mg mg**

**kyaw kyaw**

**aye aye**

Create 8\_keys folder.

Copy 0\_install/first\_app/src folder to 8\_keys folder.

## (9) React Router

```
npm install react-router-dom
```

Check in package.json file.

```
npm list
```

Create component/menu.js file.

```
import { BrowserRouter, Routes, Route } from "react-router-dom";
import EvenOdd from './even_odd.js';
import Login from './login.js';
import Home from './new_home.js';

const Menu = () => {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="login" element={<Login />} />
        <Route path="home" element={<Home />} />
        <Route path="evenodd" element={<EvenOdd />} />
      </Routes>
    </BrowserRouter>
  )
};

export default Menu;
```

Run in browser -

```
localhost:3000/login
localhost:3000/home
localhost:3000/evenodd
```

Create 9\_router folder.

Copy 0\_install/first\_app/src folder to 9\_router folder.

## (10) React Hooks

### useState

Copy component/login.js and Create component/new\_login.js file.  
Change class state instead function useState and remove this keyword.

```
import { useState } from 'react';

function NewLogin() {
  const [islogin, setlogin] = useState(false);
  const [username, setUsername] = useState("");

  const setUsername = evt => {
    setUsername(evt.target.value);
  }

  const clickLogin = evt => {
    if (username == 'kyaw' && passwd == '123') {
      setlogin(true);
    } else {
      alert('sorry, invalid username and password !');
    }
  }

  const clickLogout = evt => {
    setlogin(false);
  }
  ...
}

export default NewLogin;
```

Change Menu.js file.

```
...
import Login from './new_login.js';
...
```

## useEffect

Change new\_login.js file.

```
import { useState, useEffect } from 'react';

function NewLogin() {
  ...
  useEffect(() => {
    // run after every rendering
    if (islogin && username) {
      console.log(username, 'logged in.');
```

Add new users.

```
username = aung
passwd == 321

username = aye
passwd == 456
```

## useContext

Copy component/menu.js and Create component/new\_menu.js file.

```
import { BrowserRouter, Routes, Route } from "react-router-dom";
import EvenOdd from './even_odd.js';
import Login from './new_login.js';
import Home from './new_home.js';
import { createContext } from 'react';

export const Context = createContext();

const NewMenu = () => {

  const app_name = 'First App';

  return (
    <BrowserRouter>
      <Context.Provider value={app_name}>
        <Routes>
          ...
        </Routes>
      </Context.Provider>
    </BrowserRouter>
  )
};

export default NewMenu;
```

Change new\_login.js file.

```
...
import { Context } from './new_menu.js';
...
return (
  <div>
    <Context.Consumer>
      { value => <span> { value } </span> }
    </Context.Consumer>
    { islogin ? (
  ...
```

Add Context Consumer in **new\_home.js** and **even\_odd.js** and app\_name change to **Second App**.

Create 10\_router folder.

Copy 0\_install/first\_app/src folder to 10\_router folder.

## (11) React Styling

### CSS Stylesheet

Create src/Login.css file.

```
.Login {  
  background-color: blue;  
}
```

Change new\_login.js file.

```
...  
import './Login.css';  
...  
return (  
  <div className="Login">  
    <Context.Consumer>  
  </div>  
...)
```

### Style Object

Change even\_odd.js file.

```
...  
render() {  
  const myStyle = {  
    backgroundColor: "orange",  
  };  
  
  return (  
    <div style={myStyle}>  
      <Context.Consumer>  
    </div>  
  </div>  
...)
```

### Inline Style

Change new\_home.js file.

```
...  
return (  
  <div style={{ backgroundColor: "green" }}>  
    <Context.Consumer>  
  </div>  
...)
```

Create 11\_style folder.

Copy 0\_install/first\_app/src folder to 11\_style folder.