# Vorbereitung

NodeJS installieren

npx create-react-app reactsample

cd reactsample

npm start

<http://localhost:3000/>

# Übung 1

## Timer.js

import React from 'react';

class Timer extends React.Component {

constructor(props) {

super(props);

}

render() {

return (

<div>{new Date().toLocaleTimeString()}</div>

);

}

}

export default Timer;

## App.js

import Timer from './Timer';

render() {

return (

<div className="App">

<header className="App-header">

<img src={logo} className="App-logo" alt="logo" />

</header>

<main>

<Timer></Timer>

</main>

</div>

);

}

## App.css

.App-logo {

height: 20vmin;

…

}

.App-header {

min-height: 40vh;

…

}

# Übung 2

## Timer.js

import React from 'react';

class Timer extends React.Component {

constructor(props) {

super(props);

this.state = { date: new Date()};

setInterval(() => this.tick(), 1000);

}

tick() {

this.setState({ date: new Date()});

}

render() {

return (

<div>{this.state.date.toLocaleTimeString()}</div>

);

}

}

export default Timer;

# Übung 3

## Login.js

import React from 'react';

class Login extends React.Component {

constructor(props) {

super(props);

this.state = { isLoggedIn: false };

this.onClick = this.onClick.bind(this);

}

onClick(e) {

e.preventDefault();

this.setState({ isLoggedIn: !this.state.isLoggedIn });

}

render() {

let welcomeMessage;

if (this.state.isLoggedIn) {

welcomeMessage = <div>{'Welcome ' + this.props.name}</div>;

}

return (

<React.Fragment>

<button onClick={this.onClick}>{this.state.isLoggedIn ? 'Logout' : 'Login'}</button>

{welcomeMessage}

</React.Fragment>

);

}

}

export default Login;

## App.js

import Login from './Login';

render() {

return (

<div className="App">

<header className="App-header">

<img src={logo} className="App-logo" alt="logo" />

</header>

<main>

<Timer></Timer>

<Login name="Markus"></Login>

</main>

</div>

);

}

# Übung 4

## Login.js

import React from 'react';

class Login extends React.Component {

constructor(props) {

super(props);

this.onClick = this.onClick.bind(this);

}

onClick(e) {

e.preventDefault();

this.props.onChange(!this.props.isLoggedIn);

}

render() {

return (

<button onClick={this.onClick}>{this.props.isLoggedIn ? 'Logout' : 'Login'}</button>

);

}

}

export default Login;

## App.js

import React, { Component } from 'react';

import logo from './logo.svg';

import './App.css';

import Timer from './Timer';

import Login from './Login';

class App extends Component {

constructor(props) {

super(props);

this.state = { isLoggedIn: false };

this.onLoginChange = this.onLoginChange.bind(this);

}

onLoginChange(value) {

this.setState({ isLoggedIn: value });

}

render() {

let welcomeMessage;

if (this.state.isLoggedIn) {

welcomeMessage = <div>{'Welcome ' + this.props.name}</div>;

}

return (

<div className="App">

<header className="App-header">

<img src={logo} className="App-logo" alt="logo" />

</header>

<main>

<Timer></Timer>

<Login isLoggedIn={this.state.isLoggedIn} onChange={this.onLoginChange}></Login>

{welcomeMessage}

</main>

</div>

);

}

}

export default App;

## index.js

ReactDOM.render(<App name="Markus" />, document.getElementById('root'));

# Übung 5

## Axios

npm install axios

npm install es6-promise

## index.js

import React from 'react';

import ReactDOM from 'react-dom';

import './index.css';

import App from './App';

import \* as serviceWorker from './serviceWorker';

require('es6-promise').polyfill();

ReactDOM.render(<App name="Markus" />, document.getElementById('root'));

// If you want your app to work offline and load faster, you can change

// unregister() to register() below. Note this comes with some pitfalls.

// Learn more about service workers: http://bit.ly/CRA-PWA

serviceWorker.unregister();

## App.js

import PersonSearchPanel from './PersonSearchPanel';

class App extends Component {

render() {

let welcomeMessage;

let personSearchPanel

if (this.state.isLoggedIn) {

welcomeMessage = <div>{'Welcome ' + this.props.name}</div>;

personSearchPanel = <PersonSearchPanel />

}

return (

<div className="App">

<header className="App-header">

<img src={logo} className="App-logo" alt="logo" />

</header>

<main>

<Timer></Timer>

<Login isLoggedIn={this.state.isLoggedIn} onChange={this.onLoginChange}></Login>

{welcomeMessage}

{personSearchPanel}

</main>

</div>

);

}

}

## PersonSearchPanel

import React from 'react';

import PersonSearchForm from './PersonSearchForm';

import PersonSearchResult from './PersonSearchResult';

import axios from 'axios';

class PersonSearchPanel extends React.Component {

constructor(props) {

super(props);

this.state = { persons : [] }

this.onChange = this.onChange.bind(this);

}

onChange(text) {

axios.get('http://localhost:8080/person?name=' + text)

.then(res => {

this.setState({ persons : res.data });

}

);

}

render() {

return (

<div>

<PersonSearchForm onChange={this.onChange} />

<PersonSearchResult persons={this.state.persons} />

</div>

);

}

}

export default PersonSearchPanel;

## PersonSearchForm

import React from 'react';

class PersonSearchForm extends React.Component {

constructor(props) {

super(props);

this.onKeyUp = this.onKeyUp.bind(this);

}

onKeyUp(e) {

e.preventDefault();

this.props.onChange(e.target.value);

}

render() {

return (

<div>

<span>Name: </span>

<input type="text" onKeyUp={this.onKeyUp} />

</div>

);

}

}

export default PersonSearchForm;

## PersonSearchResult

import React from 'react';

class PersonSearchResult extends React.Component {

constructor(props) {

super(props);

}

render() {

const rows = [];

this.props.persons.forEach(person => {

rows.push(<PersonRow key={person.name + '\_\_\_' + person.vorname} person={person} />)

});

return (

<table>

<thead>

<PersonHeader />

</thead>

<tbody>

{rows}

</tbody>

</table>

);

}

}

class PersonHeader extends React.Component {

render() {

return (

<tr>

<th>Name</th>

<th>Vorname</th>

</tr>

)

}

}

class PersonRow extends React.Component {

render() {

return (

<tr>

<td>{this.props.person.name}</td>

<td>{this.props.person.vorname}</td>

</tr>

)

}

}

export default PersonSearchResult;

# Übung 6

## ErrorBoundary.js

import React from 'react';

class ErrorBoundary extends React.Component {

constructor(props) {

super(props);

this.state = { error: null, errorInfo: null };

}

componentDidCatch(error, errorInfo) {

this.setState({

error: error,

errorInfo: errorInfo

})

}

render() {

if (this.state.errorInfo) {

return (

<div>

<p><b>Es ist ein Fehler aufgetreten. Sie müssen die Seite neu laden (F5).</b></p>

<p></p>

<p>

{this.state.error && this.state.error.toString()}

</p>

<p></p>

<pre>

{this.state.errorInfo.componentStack}

</pre>

</div>

);

} else {

return this.props.children;

}

}

}

export default ErrorBoundary;

## PersonSearchPanel.js

render() {

return (

<ErrorBoundary>

<div>

<PersonSearchForm onChange={this.onChange} />

<PersonSearchResult persons={this.state.persons} />

</div>

</ErrorBoundary>

);

}

## PersonSearchResult.js

class PersonRow extends React.Component {

render() {

if (this.props.person.name === 'XXX1') {

throw new Error('I crashed!');

}

return (

<tr>

<td>{this.props.person.name}</td>

<td>{this.props.person.vorname}</td>

</tr>

)

}

}

# Weiteres

npm install rxjs

npm install react-materialize

<head>

<!-- Import Google Icon Font -->

<link href="http://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

<!-- Import materialize.css -->

<link href="https://cdnjs.cloudflare.com/ajax/libs/materialize/0.98.0/css/materialize.min.css" rel="stylesheet">

</head>

<body>

<!-- Import jQuery before materialize.js -->

<script src="https://code.jquery.com/jquery-2.1.1.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/0.98.0/js/materialize.min.js"></script>

<!-- And then your bundled js -->

<script src="path/to/your/bundle.js"></script>

</body>