

Webtechnologien Übungen React 2

Vorbereitung: Öffne in VS Code den Projektordner react2, navigiere in VS Code mit der Console in den Ordner react2 und führe den Befehl `npm install` aus. In diesem Ordner siehst du die in der React 1 Übung behandelten Struktur. Öffne die App.js Datei, in dieser wirst du folgende Aufgaben erledigen.

Aufgabe 1:

Übersicht der Todos: React importieren, Klasse zur Komponente machen, im Browser austesten

Bearbeite App.js so, dass es sich danach um eine React-Komponente handelt. Gehe dafür in die vorbereitete App.js-Klasse und erweitere diese um die Component-Klasse von React. Du musst Aufgabe 1 abschließen damit App.js mit dem Befehl `npm run start` fehlerfrei ausgeführt wird und korrekt im Browser angezeigt wird. Überprüfe ob es geklappt hat indem du `localhost:3000` auf bzw. die IP aus der Console aufrufst.

Aufgabe 2:

Wo wird der Zustand einer Komponente gespeichert und **Wie** wird dieser Zustand verwaltet? Gib kurz ein Beispiel für eine Initialisierung eines Zustands und anschließend die Veränderung während der Ausführung durch bspw. klicken eines Buttons. Teste dein Beispiel in dem du es in der App.js schreibst und dir einen Wert aus dem Zustand ausgeben lässt und manipulierst.

Klicke einmal mit gedrückter Strg-Taste in VS Code in der App.js auf Component. Damit öffnest du die Klasse Component von React. Dort findest du neben der Hülle für die render()-Funktion auch den state.

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

Aufgabe 3:

Übersicht der Todos: addItem(), deleteItem(), Hinzufügen-Button-Funktion, Löschen-Button, Array in Zustand, Darstellung einer Liste des Arrays

Verändere die Komponente so, dass initial nur ein leeres Array im Zustand enthalten ist. Lösche den Button und die Ausgabe aus Aufgabe 2. Füge dem Button *Element hinzufügen* eine Funktion hinzu. Es soll das Array also beim Klicken des Hinzufügen-Buttons um einen Eintrag mit dem Text aus dem Textfeld erweitert werden, schreibe dafür eine eigene Funktion, die zunächst die Eingabe aus dem gegebenen Textfeld ausliest und im Komponenten-Zustand in dem erstellten Array hinzufügt. In einer listenartigen Darstellung sollen alle Inhalte aus dem Array angezeigt werden. Für jedes erstellte Listenelement soll ein Löschen-Button erstellt werden, der das dieses Element aus dem Komponenten-Zustand löscht.

Aufgabe 4 (optional):

Erstelle eine neue Datei im src Ordner dieser Aufgabe und nenne Sie *lichtschalter*. Installiere dann *ES7 React/Redux/GraphQL/React-Native snippets* in VS Code und versuche verschiedene Befehle aus, die dir bei der Erstellung einer Komponente helfen. Hier ist die Kürzel-Liste:

<https://marketplace.visualstudio.com/items?itemName=dsznajder.es7-react-js-snippets>

Nach dem Installieren musst du lediglich in Visual Code das Kürzel schreiben und dann Tab drücken. Nutze einmal folgende Befehle

imr	cmmb	est	ren	sst	exp
-----	------	-----	-----	-----	-----

Abbildung 1: Befehle React-js-Snippets

Wenn das funktioniert hat, erweitere die Render-Funktion um einen Button („Lichtschalter“), der beim Klick die Hintergrundfarbe der Komponente jeweils hell bzw. dunkel macht. Vergleichbar mit einem Tag- / Nachtmodus in Apps. Importiere die Komponente und platziere die neu geschriebene Komponente innerhalb der render-Funktion der App.js mit dem HTML-Tag: <Komponentenname />.

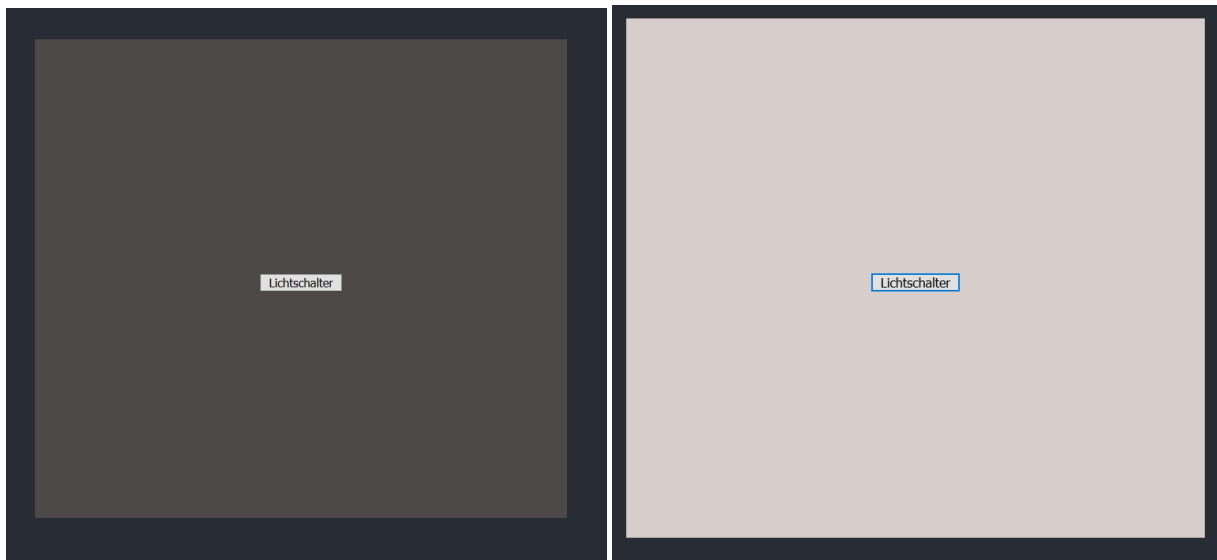


Abbildung 2: Beispiel für ein mögliches Ergebnis