

- ⑥ Kommunikation zwischen Komponenten
- Angular vs. React: props für beides
 - Beispiel: Timezone Offset, (Prop hinzufügen),
Timezone Offset + Input, Timezone Offset + Buttons ^{ChangeEvent}

- ⑦ Styling von Komponenten
- CSS modules - Emotion

- ⑧ JSX, ... But why ...
- Separation of concerns ?? → reduce coupling, increase cohesion
Kohäsion von Render-Logik + UI-Logik!
 - JSX + React-Komponenten ~~testen~~: hohe Kohäsion innerhalb der Komponenten, lose Kopplung zwischen Komponenten

- ⑨ JSX ... Check ...
- Template Sprache: JavaScript-Extension
 - Value UI → Inversion of control! - Beispiel: FormControlLabel

- ⑩ Übung #1 Setup + Timer

- ⑪ Uncontrolled vs Controlled

- ⑫ Custom Hooks - Beispiel: useClock, useWindowSize,
useDebounce

- ⑬ Server-Frontend-Kommunikation - Beispiel: usePromise,
React Query - Beispiel: Tour of Heroes

- ⑭ Übung #2 Tour of Heroes I

- ⑫ Routing ⑬ Formulare ⑭ Nur React ⑮ react-hook-form

- ⑯ Übung #3 Tour of Heroes II

Danke !!

- ① ich und React → Bild von Fred

- ② React → Library → Building User Interfaces,
nicht mehr, nicht weniger → kein Eingebaute
Lösung fürs Routing, Formulare, etc.

- ③ JSX(TSX) → JavaScript Extension

→ HTML-Syntax in JavaScript → UI-Böcken
können in Variablen gespeichert werden, ~~also~~

JavaScript-Ausdrücke in HTML

- Beispiele: Hallo Welt, Hallo Welt + Variablen, Hallo Welt +
Funktion, JSX als Rückgabewert einer Funktion, + Null,

JSX + Schleifen, JSX → Element Nodes einfache JS-Objekte

- ④ Exkurs: JavaScript Dekonstruktion → wird oft
Thema verwendet in React

- ⑤ Komponente: Clock

- Beispiel: 1. tick + Interval (Dev-Tools),

→ Komponenten = Funktionen, die UI zurück geben

→ ~~Per Einzelne Pure Funktionen~~

- Beispiel: "Dumme" Komponente

→ Wiederverwendbare Komponente: Überleitung zu State +
Effekten

- ⑥ useState-Hook: Aufbau, Nutzung

- Beispiel: Clock (fehlende setState).

- ⑦ useEffect-Hook: Aufbau, Nutzung - Beispiel: Clock