

Warum sollte man JSDoc nutzen ?

JSDoc bietet die Möglichkeit, ältere Codebasen zu modernisieren, ohne dass diese einen kompletten Rewrite unterzogen werden müssen durch TypeScript. Darüber hinaus bricht JSDoc den natürlichen dynamischen Codefluss von JavaScript nicht, sondern "ergänzt" diesen in Kombination mit `//@ts-check` durch simples Type-Checking.

Die Regeln der Anwendung sind denkbar einfach und JavaScript Codebasen profitieren bei der Verwendung von JSDoc in Form besserer Autovervollständigung und der Qualität aufkommender Fehlermeldungen in der IDE.

Zudem beschreibt JSDoc den genauen Aufbau und die Verwendung einer Funktion, wodurch diese besser verständlich und wesentlich einfacher nutzbar wird.

```
function useCountdown(start?: number, counterInterval?: number): number
  Decrementing Countdown
  @param start — default is 5
  @param counterInterval — in ms, default is 1000
  @returns — decremented counter value
```

Der entsprechende JSDoc-Kommentar dazu sieht wie folgt aus:

```
/**
 * Decrementing Countdown
 *
 * @param {number} start default is 5
 * @param {number} counterInterval in ms, default is 1000
 * @returns decremented counter value
 */
```

Die Funktion selbst bleibt dabei in gewöhnlichem JavaScript geschrieben:

```
export function useCountdown(start = 5, counterInterval = 1000) {
  const [count, setCount] = useState(start);

  useEffect(() => {
    // create timer
```

```
const timer1 = setInterval(() => {
  setCount((c) => c - 1);
}, counterInterval);

// clear timer, otherwise multiple timers are running
return () => clearInterval(timer1);
}, []);

return count;
}
```

Revision #5

Created 2026-05-04 08:43:45 UTC by st3ve

Updated 2026-05-04 10:01:30 UTC by st3ve