

LU01d - How to comment JS code

Einleitung

Eine saubere und verständliche Dokumentation von JavaScript-Code ist entscheidend, um Projekte wartbar, erweiterbar und für andere Entwickler nachvollziehbar zu halten. Dokumentation hilft nicht nur im Team, sondern auch dabei, den eigenen Code nach Wochen oder Monaten schnell wieder zu verstehen.

Warum Code dokumentieren?

- Erleichtert das Verständnis für andere Entwickler.
- Hilft bei der Fehlersuche und Wartung.
- Ermöglicht eine einfachere Übergabe von Projekten.
- Verhindert doppelten oder unnötigen Code.

Formen der Dokumentation

1. Inline-Kommentare: Kurze Hinweise im Code, um komplexe Stellen zu erklären.
2. Block-Kommentare: Mehrzeilige Erklärungen für Funktionen, Klassen oder komplexe Logik.
3. JSDoc: Standardisierte Syntax für Kommentare, um Funktionen, Parameter und Rückgabewerte zu beschreiben.
4. README-Dateien: Übersichtsdokumentation für das gesamte Projekt.

Innerhalb des Unterrichts werden wir die ersten beiden Varianten anwenden.

[Kommentare in JS](#)

Best Practices wann kommentiert werden sollte

- Kommentare prägnant und verständlich halten. Nicht WAS (offensichtlich), sondern WARUM (Gedanken und Gründe)
- Nur dokumentieren, was nicht selbsterklärend ist.
- Funktionen und Variablen mit sprechenden Namen versehen.
- Einheitliche Dokumentationsstandards (z. B. JSDoc) verwenden.
- Dokumentation aktuell halten und bei Änderungen anpassen.

Beispiel einzelilige Kommentare

```
// Einfache Variablen-Definition
let name = "Max";
// Ausgabe im Browser-Log
```

```
console.log("Hallo " + name); // Begrüßung ausgeben
```

Beispiel für mehrzeilige Kommentare

```
/*  
Diese Funktion berechnet den Durchschnitt  
aus zwei Zahlen. Sie nimmt zwei Parameter entgegen  
und gibt den arithmetischen Mittelwert zurück.  
*/  
function average(a, b) {  
    return (a + b) / 2;  
}  
console.log(average(10, 20)); // Ausgabe: 15
```

Beispiel JSDoc-Kommentar

```
/**  
 * Berechnet die Summe zweier Zahlen.  
 * @param {number} a - Erste Zahl.  
 * @param {number} b - Zweite Zahl.  
 * @returns {number} Summe von a und b.  
 */  
function add(a, b) {  
    return a + b;  
}
```

Zusatzmaterial

- [W3School-js comments](#)
- [SelfHTML-JS Kommentare](#)



Volkan Demir

From:
<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:
<https://wiki.bzz.ch/de/modul/m288/learningunits/lu01/04>

Last update: **2025/09/01 10:48**

