

# Tag 4 - JavaScript: User Interaction

---

## Einleitung

In vielen Programmen müssen Benutzer Daten eingeben oder Entscheidungen treffen. Bei clientseitigem JavaScript (also im Browser) gibt es dafür einfache Mittel: sogenannte **Dialogfenster**. Diese bieten eine Möglichkeit, **Benutzereingaben abzufragen**, **Nachrichten anzuzeigen** oder **Entscheidungen einzuholen** – ohne HTML oder DOM-Kenntnisse vorauszusetzen.

---

## 1. Die drei Dialogboxen im Überblick

Funktion	Methode	Rückgabebetyp	Beschreibung
Hinweis	<code>alert()</code>	<code>undefined</code>	Zeigt nur Text an
Bestätigung	<code>confirm()</code>	<code>true</code> / <code>false</code>	OK oder Abbrechen
Eingabe	<code>prompt()</code>	<code>string</code> oder <code>null</code>	Eingabe + OK/Abbrechen

---

## 2. `alert()` – Einfacher Hinweis

```
alert("Willkommen auf der Seite!");  
alert(4 * 7);  
alert(true);
```

- Zeigt eine einfache Meldung mit **OK-Button**.
- Es kann nur **ein Parameter** übergeben werden. Weitere Parameter werden ignoriert.

```
alert("Text 1", "Text 2"); // nur "Text 1" wird angezeigt
```

- Rückgabewert ist `undefined`.
  - Die Ausführung des Codes **pausiert**, bis der Dialog geschlossen wird.
- 

## 3. `confirm()` – Bestätigungsdialog

```
let antwort = confirm("Willst du wirklich fortfahren?");  
console.log(antwort); // true bei OK, false bei Abbrechen
```

- Zeigt Text mit **zwei Buttons**: OK und Abbrechen
- Gibt `true` zurück bei OK, `false` bei Abbrechen

Anwendungsbeispiel mit ternärem Operator:

```
let löschen = confirm("Alle Daten löschen?");  
let meldung = löschen ? "Daten werden gelöscht" : "Abgebrochen";  
console.log(meldung);
```

---

## 4. `prompt()` – Texteingabe vom Nutzer

```
let name = prompt("Wie heißt du?", "Max Mustermann");
console.log(name);
```

- Öffnet ein Eingabefeld + OK + Abbrechen
- Zwei optionale Parameter:
  1. **Fragetext**
  2. **Standardwert (optional)**

Rückgabewerte:

Benutzeraktion	Rückgabewert	Typ
Eingabe + OK	z. B. "Alice"	<code>string</code>
OK ohne Eingabe	<code>""</code> (leerer String)	<code>string</code>
Abbrechen	<code>null</code>	<code>object</code> (Typ von <code>null</code> )

Beispiel mit Prüfung:

```
let name = prompt("Was ist dein Name?", "John Doe");
name = name ? name : "anonym";

let alter = prompt("Hallo " + name + ", wie alt bist du?");
alert(name + " ist " + alter + " Jahre alt.");
```

Wichtige Hinweise:

- Rückgabewert ist immer ein **String** oder `null`
- Zahlen müssen ggf. mit `Number()` umgewandelt werden:

```
let alter = Number(prompt("Wie alt bist du?"));
```

- Immer auf **ungültige oder leere Eingaben** vorbereitet sein:

```
if (alter && !isNaN(alter)) {
  console.log("Alter gültig: " + alter);
} else {
  console.log("Ungültige Eingabe!");
}
```

---

## 5. Warnung: Verwendung in realen Anwendungen

- Dialogboxen blockieren den Codefluss und die Benutzeroberfläche → **modal**
- Werden in **modernen Web-Apps kaum mehr verwendet**
- In diesem Kurs dienen sie vor allem zum **Lernen, Testen und Experimentieren**

---

## 6. Übungen zur Benutzerinteraktion

### Aufgabe 1: Begrüßung

Frage den Namen per `prompt()` ab und begrüße den Benutzer mit `alert()`. Wenn der Benutzer nichts eingibt oder abbricht, zeige „Willkommen, Gast!“ an.

### Aufgabe 2: Entscheidung

Frage mit `confirm()`, ob Daten gelöscht werden sollen. Zeige je nach Antwort eine entsprechende Nachricht in der Konsole an.

### Aufgabe 3: Zahleneingabe prüfen

Frage mit `prompt()` nach dem Alter. Wenn die Eingabe keine Zahl ist, zeige „Ungültige Eingabe“.

### Aufgabe 4: Zusammenspiel mehrerer Dialoge

Frage Name und Alter ab, bestätige beides per `confirm()`. Zeige die Zusammenfassung mit `alert()` nur, wenn bestätigt wurde.

### Aufgabe 5: Default-Werte bei `prompt()` nutzen

Frage den Benutzer nach seinem Beruf. Verwende "Frontend Developer" als Standardwert. Zeige das Ergebnis in der Konsole.

---