

Modul 294 - Aufgabenblatt 02

von Lukas Meier

Unterricht vom 02.12.2025

1 Vorbereitung

Erstellen Sie einen Ordner an einem Ort Ihrer Wahl und speichern Sie die Datei `index.html` welche Sie auf dem Netzwerkordner finden darin.

2 Array-Loop

Im script-Tag der `index.html`-Datei finden Sie bereits ein vordefiniertes Array. Schreiben Sie insgesamt 3 Loops welche der Reihe nach alle Zahlen aus dem Array in der Konsole des Browsers ausgeben. Nutzen sie `while`, `for` und `forEach` für die Schleifen...

3 Reverse

Passen Sie einen Loop ihrer Wahl so an, dass die Zahlen in der umgekehrten Reihenfolge ausgegeben werden. (Empfehlung: `for` oder `while`).

4 Mathematische Operatoren

Erweitern Sie einen Loop ihrer Wahl so, dass anstelle der Zahl im Array, die Zahl plus 8 ausgegeben wird.

5 Array-Loop mit Conditional

Erweitern Sie einen Loop ihrer Wahl so, dass nur noch die Zahlen ausgegeben werden, welche gerade sind.

6 Prüfen des Indexes

Erweitern Sie einen anderen Loop ihrer Wahl so, dass nur noch jede dritte Zahl ausgegeben wird.

7 Mathematische Operatoren mit Index

Ändern Sie den Loop aus der Aufgabe **Mathematische Operatoren** so, dass nicht die Zahl 8 addiert wird, sondern die aktuelle Position der Zahl im Array. (Die erste Zahl also +0 die zweite +1 die dritte +2 usw.)

8 Selektieren und Entfernen eines Elementes

Schreiben Sie JavaScript Code in der `index.html` Datei, welcher das `h1`-Element aus dem DOM-Entfernt, sie können dazu die Methode `remove` ohne Parameter auf dem Element aufrufen.

Verwenden Sie dazu `document.getElementById`.

9 Entfernen aus einer Liste

Unterhalb des `h1` finden Sie eine Liste von Elementen, entfernen Sie das dritte `li` Element aus dem DOM. Verwenden Sie dazu die folgenden Selektoren der Reihe nach: `querySelector`, `querySelectorAll` und `getElementById`.

10 Prüfen des Inhalts

Selektieren Sie via JavaScript alle Elemente mit der Klasse `item`. Schreiben Sie nun eine Schleife über alle Elemente und entfernen Sie jedes Element, welches den Inhalt `Element 5` hat. Passen Sie das DOM danach so an, dass mehrere Elemente der Liste den Inhalt `Element 5` haben und prüfen Sie ob alle ordnungsgemäss entfernt werden. (Reminder: Auf jedem Element existiert das Property `innerHTML`, damit können Inhalt eines Elementes abgefragt und auch neu gesetzt werden.)

11 Anpassen des Inhaltes

Erweitern Sie die Schleife der vorherigen Aufgabe so, dass zusätzlich das zweite Element der Liste den neuen Inhalt `Ich wurde bearbeitet` hat.

12 Anpassen der CSS-Klassen

Selektieren Sie den Text im Footer und geben Sie ihm zusätzlich zu der Klasse `text` noch die Klasse `blink`. Sie können dafür die Methode `add` auf dem Objekt `classList` nutzen welches sie auf der Node vorfinden.

13 Umschalten von CSS-Klassen

Passen Sie das HTML so an, dass im Footer noch ein zweiter Paragraph mit der Klasse `text` existiert. Geben Sie ihm ausserdem bereits die Klasse `blink`.

Passen Sie ihren JavaScript-Code so an, dass alle Paragraphen im Footer, welche die Klasse `Blink` haben, diese verlieren und die Paragraphen welche sie nicht haben diese erhalten.

Das Ganze sollte auch funktionieren, wenn die Reihenfolge der Texte im Footer geändert wird. Nutzen Sie dafür die `contains`-Methode auf der Klassenliste der Elemente.

14 CSS-Styling anpassen

Um nicht nur über CSS-Klassen das Styling der Elemente anzupassen, können wir CSS-Properties auch direkt via JavaScript anpassen/definieren. Dafür gibt es auf dem Objekt `style` jedes CSS-Property als Attribut welches ausgelesen und gesetzt werden kann. Der einzige Unterschied besteht darin, dass die Properties in JavaScript in `camelCase` definiert sind und nicht wie in CSS in `kebab-case`.

Fügen Sie im Footer einen Link hinzu, den Inhalt und das Ziel definieren Sie selbst. Schreiben Sie nun einen JavaScript code, welcher die Farbe auf den Wert `#333333` setzt. Setzen Sie zusätzlich die Schriftgrösse auf 8 Pixel.