

WEB-TECHNOLOGIEN

ÜBUNG 7: JAVASCRIPT-BASICS

AUFGABE 1: JS CODING KATA

Unter einer Kata versteht man eine kleine, abgeschlossene Programmierübung, die die Programmierfertigkeiten "schärfen" soll. Der Begriff entstammt der Clean-Code-Bewegung (z.B. [Coding Dojo](#)).

Lösen Sie folgende Kata mit Hilfe von JavaScript (mit Stift und Papier!):

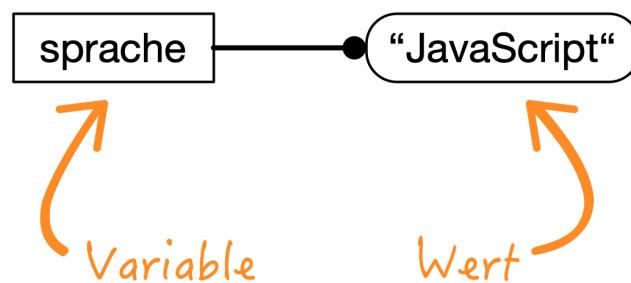
Schreiben Sie ein Programm, welches alle Zahlen von 1 bis 100 ausgibt. Für Vielfache von 3 soll das Programm statt der Zahl die Zeichenkette "Tic" ausgeben, für Vielfache von 5 die Zeichenkette "Tac", und für solche Zahlen, die ein Vielfaches von 3 und 5 sind, die Zeichenkette "TicTac".

AUFGABE 2: VARIABLEN UND WERTE

Betrachten Sie unten stehenden JavaScript-Code. Welche Werte haben die vier deklarierten Variablen nach Ausführung des Codes? Skizzieren Sie die Antwort mit Hilfe der "Kabeldarstellung" aus der Vorlesung (siehe unten).

```
let isBad = false;  
let isGood = !isBad;  
let isSomething = isBad || isGood;  
let isAmbivalent = isBad && isGood;  
isBad = true;
```

"Kabeldarstellung" aus der Vorlesung:



AUFGABE 3: SICHTBARKEIT

Betrachten Sie unten stehenden JavaScript-Code. Welche Ausgabe produzieren die mit Kommentaren markierten Zeilen?

```
var number1 = 23;
let number2 = 42;

console.log(number1, number2); // Ausgabe 1

if (number1 < number2) {
    var number1 = 42;
    let number2 = 23;

    function func(number) {
        var number1 = number;
        let number2 = 23;

        console.log(number1, number2);
    }

    console.log(number1, number2); // Ausgabe 2
    func(number2); // Ausgabe 3
    console.log(number1, number2); // Ausgabe 4
}

console.log(number1, number2); // Ausgabe 5
```

AUFGABE 4: FUNKTIONEN

Die Funktion `rechne` in unten stehendem JavaScript-Code stellt einen äußerst simplen Taschenrechner dar. Ergänzen Sie die drei fehlenden Funktionen, welche mögliche Operationen des Taschenrechners darstellen (siehe mit Kommentaren markierte Zeilen):

1. `addiere`: Schreiben Sie diese Funktion als *Funktionsanweisung*.
2. `subtrahiere`: Schreiben Sie diese Funktion als *Funktionsausdruck*.
3. `multipliziere`: Schreiben Sie diese Funktion als *Arrow Function*.

```
function rechne(operand1, operand2, operation) {
  let ergebnis = operation(operand1, operand2);
  return ergebnis;
}

// Hier Code hinzufügen!

console.log(rechne(1, 5, addiere)); // 1
console.log(rechne(10, 5, subtrahiere)); // 2
console.log(rechne(7, 7, multipliziere)); // 3
```