

Übungen zu Boolean und Number

Aufgabe 1

Erstelle ein Boolean-Objekt mit dem Standardwert und ein weiteres mit dem Wert `true`. Verwende `toString()` und `valueOf()`, um die Werte in beiden Objekten in eine Zeichenkette und einen primitiven Wert zu konvertieren.

Aufgabe 2

Konvertiere die folgenden Werte in primitive boolesche Werte, ohne den `new` Operator zu verwenden: `"true"`, `0`, `undefined`, `1`, `NaN`.

Aufgabe 3

Verwende `Boolean` und teste, ob eine leere Liste `[]`, ein leeres Objekt `{}` und eine nicht definierte Variable als "truthy" oder "falsy" gelten.

Aufgabe 4

Erstelle ein Number-Objekt mit dem Wert `100` und verwende die Methoden `toString()` und `toFixed()`, um das Ergebnis in verschiedenen Formaten darzustellen: Hexadezimal, Binär und Dezimal mit 2 Nachkommastellen.

Aufgabe 5

Verwende die statischen Methoden von `Number`, um zu überprüfen, ob die folgenden Werte Ganzzahlen sind, sicher gespeichert werden können und endliche Zahlen sind: `2e100`, `-9999999999999999`, `0.5`, `Infinity`.

Aufgabe 6

Schreibe eine Funktion, die eine Zeichenkette mit einer Zahl wie `"120px"` entgegennimmt und diese mit `parseInt()` und `parseFloat()` in eine Zahl umwandelt.

Aufgabe 7

Verwende `toLocaleString()`, um die Zahl `123456.789` in den folgenden Regionen darzustellen: `de-DE`, `en-US`, `fr-FR`, und als Währung in Euro.

Aufgabe 8

Runde die Zahl `1.255` mit `toFixed(2)` und erkläre, warum das Ergebnis eventuell nicht genau das ist, was du erwartest.

Aufgabe 9

Schreibe eine Funktion, die eine Liste von Zahlen akzeptiert und für jede Zahl überprüft, ob sie eine Ganzzahl, eine sichere Ganzzahl und eine endliche Zahl ist. Gib die Ergebnisse für jede Zahl in einem strukturierten Format zurück.

Aufgabe 10

Schreibe ein Programm, das zwei Gleitkommazahlen `x = 0.3` und `y = 0.6` addiert und das Ergebnis auf 1 Dezimalstelle mit `toFixed()` rundet. Erkläre, warum das nicht exakt `0.9` ist und wie du das Problem beheben kannst.