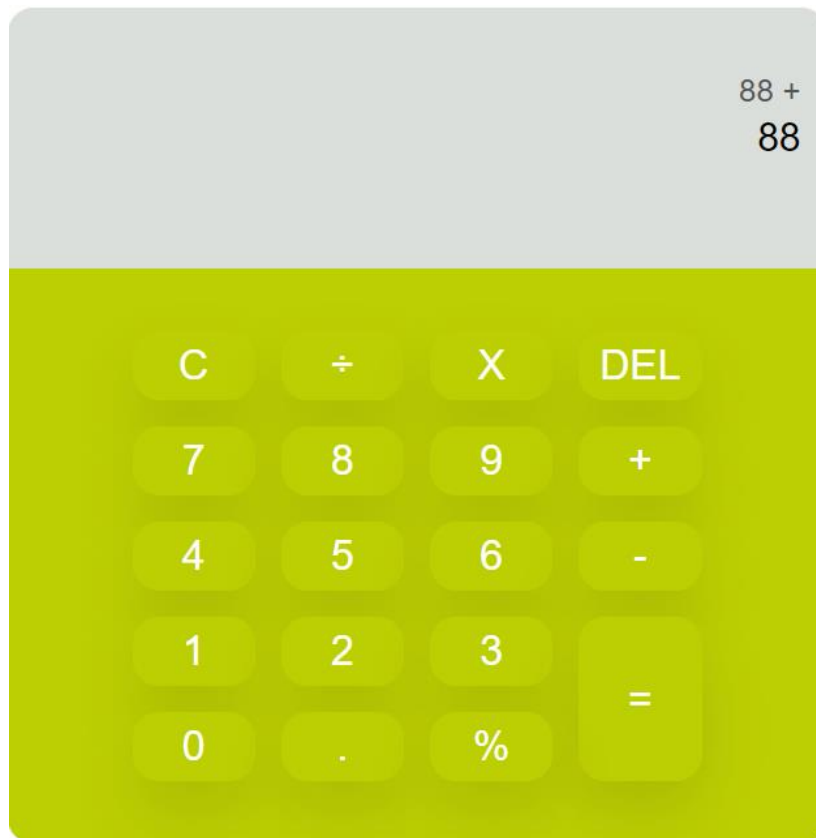


JS-AUFGABE

Eignungstag 2023



Inhaltsverzeichnis

1	Theorie.....	2
1.1	Mathematische Operationen.....	2
1.2	Funktionen.....	2
1.3	Variablen.....	2
2	Aufgabenstellung	3
2.1	Vorbereitung	3
2.2	Aufgabe	5

1 Theorie

1.1 Mathematische Operationen

Mathematische Operationen werden in JavaScript ähnlich wie in der Mathematik durchgeführt. Das jeweilige Operationszeichen steht zwischen zwei Zahlen oder Variablen.

```
// Addition
1 + 1;

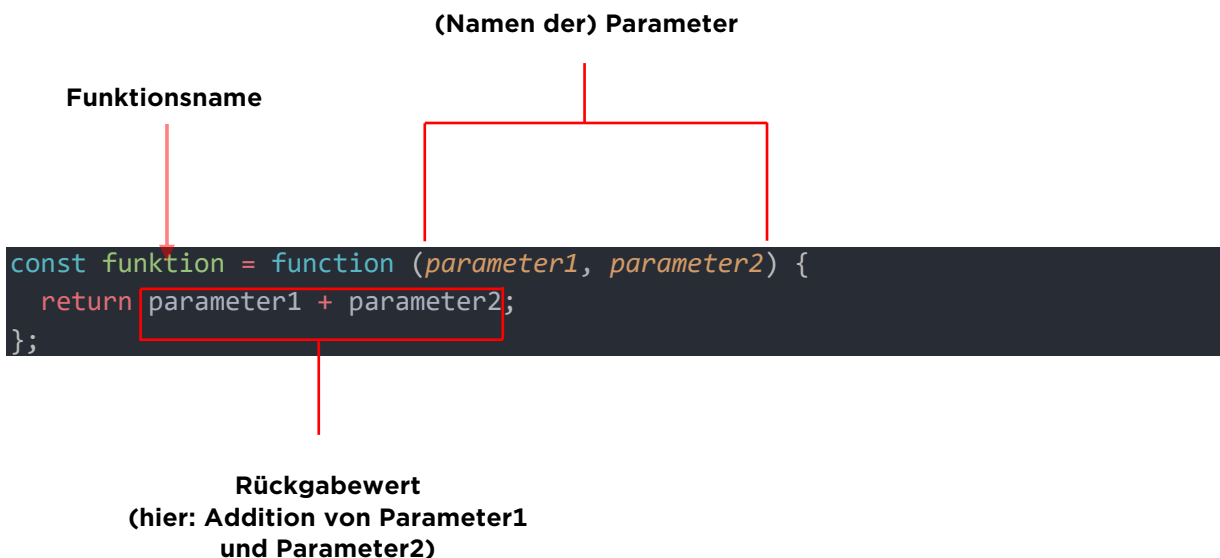
// Subtraktion
1 - 1;

// Multiplikation
2 * 3;

// Division
3 / 2;
```

1.2 Funktionen

Wie in der Mathematik (z.B. $y = f(x)$) gibt es in JavaScript auch Funktionen. Die Idee hinter einer Funktion ist, dass man eine bestimmte Funktionalität (im Code) wiederverwendbar macht. Zum einen eliminiert man Redundanzen (Wiederholungen) im Code, zum zweiten muss man Änderungen nur an einer zentralen Stelle vornehmen. Eine Funktion nimmt Argumente bzw. Parameter an, nutzt diese für irgendetwas, und gibt etwas wieder zurück. Schauen wir uns an, wie eine Funktion aussieht.



1.3 Variablen

Eine Variable hat die gleiche Rolle wie in der Mathematik: Einen Wert zu halten. Du kannst dir eine Variable vorstellen, wie eine angeschriebene Kiste: Jedes mal, wenn du etwas aus der Kiste rausnehmen möchtest, musst du zuerst die richtige Kiste finden (deshalb schreibst du es ja an). So haben in JavaScript Variablen Namen, damit man sie ansprechen kann. Schauen wir uns das doch an.

Variablenname

Variablenwert

```
const variable = 1;
```

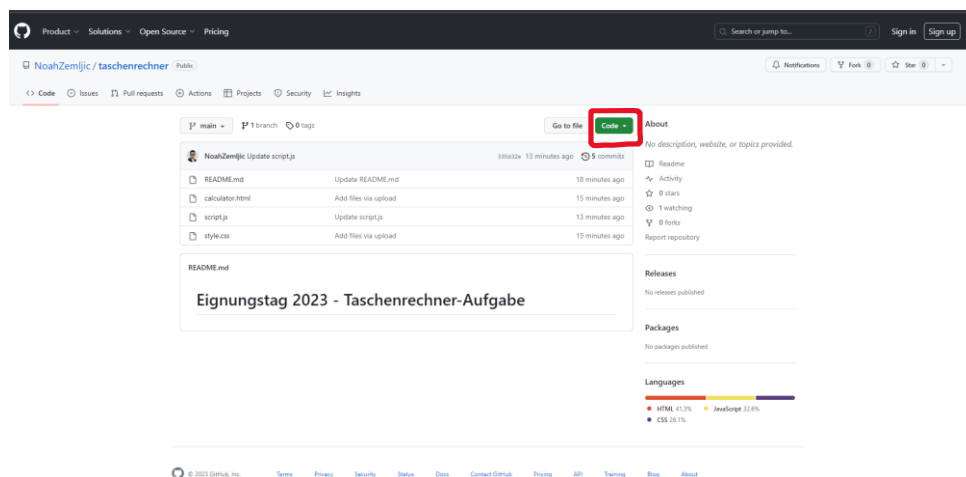
Wenn wir nun die Variable ansprechen möchten, zum Beispiel die Variable mit zwei multiplizieren, würde das so aussehen:

```
variable * 2;
```

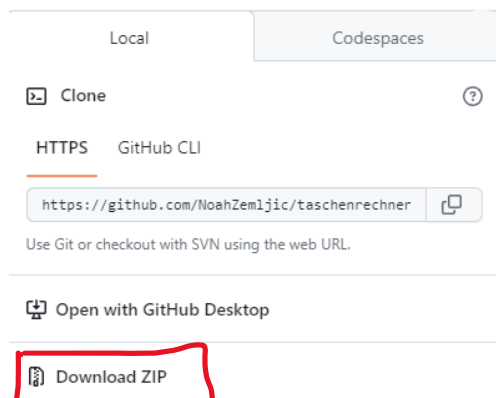
2 Aufgabenstellung

2.1 Vorbereitung

Bevor Du mit der eigentlichen Aufgabe starten kannst, musst du die Aufgabe online herunterladen. Die Aufgabe ist auf GitHub hochgeladen. GitHub ist ein sogenanntes Version Control System (VCS), was nichts anderes macht als uns erlaubt, vorherige Versionen von unserem Code zu holen. Du findest die Aufgabe unter den folgenden Link: <https://github.com/NoahZemljic/taschenrechner>.

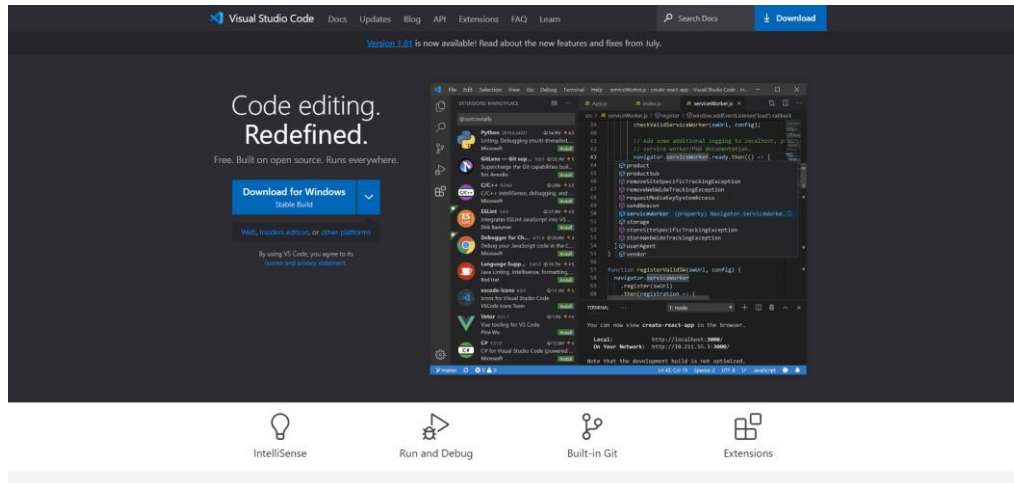


Nun klickst Du auf den grünen Knopf Code und wählst «Download ZIP» aus.



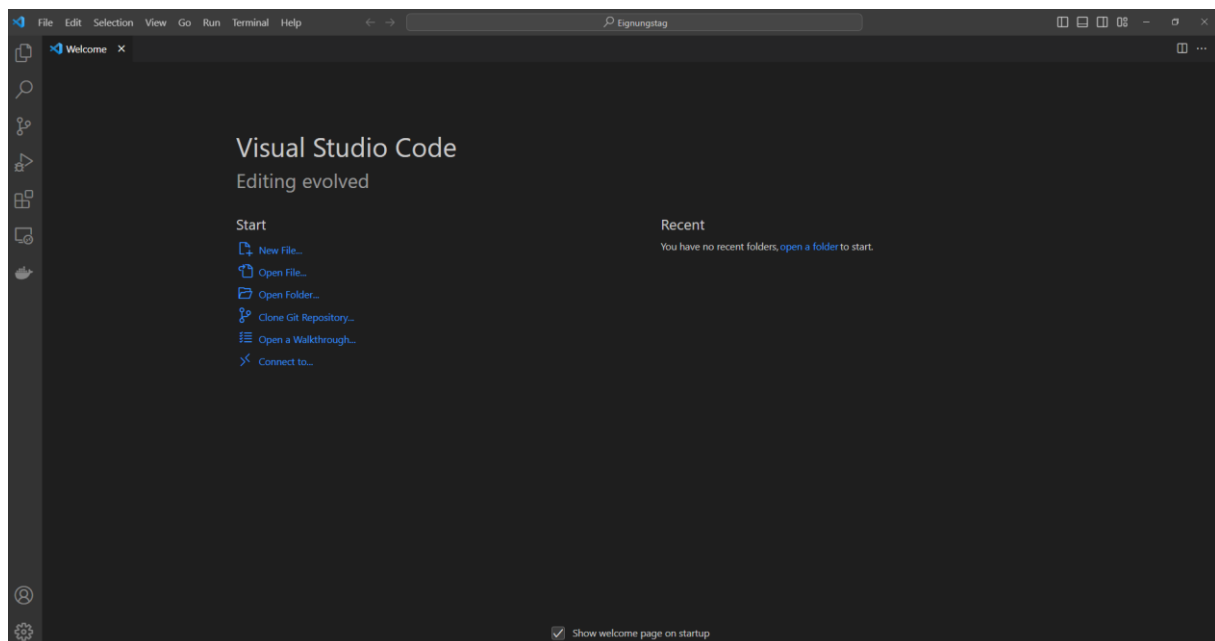
Jetzt kannst Du die ZIP-Datei im Ordner «Downloads» speichern und entpacken.

Super, aber wie kann man jetzt diese Aufgabe öffnen. Dazu brauchen wir einen Code-Editor. Ein Code-Editor ist eine Anwendung, welches uns erlaubt, Quellcode zu bearbeiten. Für diese Aufgabe nutzen wir den bekanntesten Code-Editor der Welt: Visual Studio Code. Anbei der Link: <https://code.visualstudio.com>.



Klicke auf den blauen Knopf «Download for Windows (Stable Build)» und installiere die Anwendung.

Wenn du die neu installierte Anwendung öffnest, sollte dieses Fenster aufkommen.



Nun gehst du auf Files -> Open Folder und wähle den Ordner im Ordner «taschenrechner-main» aus.

2.2 Aufgabe

Nun ist es so weit: Zeit für eine Aufgabe!

Ein Taschenrechner ist vorgegeben (das UI und die Logik). Deine Aufgabe ist es, die Funktionalität in der Datei «script.js» zu implementieren, nämlich die mathematischen Operationen. Achtung: Hier gibt es noch die Operation namens Modulo. Schaue online nach, wie das entsprechende Zeichen in JavaScript aussieht.

```
41  switch (action) {  
42      case '+':  
43          currentCalcEl.textContent = '';  
44          break;  
45      case '-':  
46          currentCalcEl.textContent = '';  
47          break;  
48      case 'X':  
49          currentCalcEl.textContent = '';  
50          break;  
51      case '÷':  
52          currentCalcEl.textContent = '';  
53          break;  
54      case '%':  
55          currentCalcEl.textContent = '';  
56  }
```

Im Screenshot siehst du welchen Abschnitt die Aufgabe betrifft (Zeilen 41 bis 56). Bitte ändere keinen Code ausserhalb von diesem Bereich sowie diese break Anweisungen (Zeilen 44, 47, 50, 53), sonst wird der Taschenrechner nicht mehr funktionieren.

Die zwei Zahlen, mit welchen gerechnet werden muss, sind in den Variablen calc1 und calc2 abgespeichert. Die Funktionalität muss nach dem Gleich (=) kommen. Wichtig zu beachten ist, dass diese Zahlen als "Strings" gespeichert sind. Um mit ihnen rechnen zu können, muss man jedoch sie zu Zahlen konvertieren. Schaue doch nach, wie man das macht.

Um den Taschenrechner überhaupt sehen zu können, musst Du einen Doppelklick auf die Datei index.html machen.