
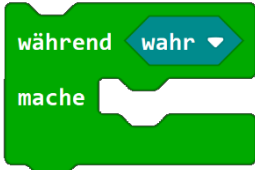


Grundlagen zur Programmierung des micro:bit VI

Wiederholungen (aus Schleifen)

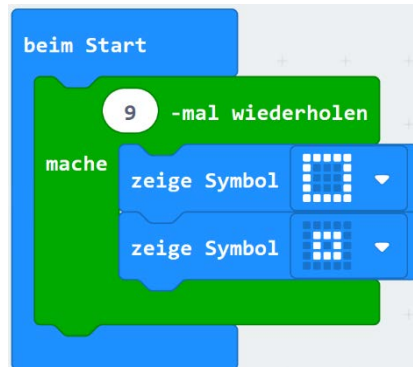
Du kennst dich nun schon sehr gut mit Verzweigungen und Variablen aus und hast auch viele coole und lustige Dinge damit gemacht. Es gibt aber noch mehr: Wenn wir z.B. einen Codeblock 5x wiederholen wollen ehe etwas anderes passiert (z.B. 5x blinken bevor das Programm startet), ist das ziemlich unpraktisch. Zurzeit müssten wir die gleichen Blöcke fünfmal untereinander reihen. Dafür gibt es **Wiederholungsblöcke**, die ihren Inhalt mehrmals wiederholen – entweder bis ein bestimmtes Ereignis eintritt (while) oder du gibst eine fixe Anzahl ein (Zählschleife bzw. for). Diese Wiederholungsblöcke nennt man in der Informatik „**Schleifen**“. Auch der „**Dauerhaft**“-Block ist eine **Schleife**, die durchgehend wiederholt wird.

Um Schleifen zu programmieren, benötigen wir folgende Blöcke aus :

Block	Beschreibung
	<p>wiederholt die Blöcke, die innerhalb stehen, so oft wie angegeben (hier 4 mal); (je nach Zahl, ändert sich die Anzahl der Wiederholungen)</p> <p>nennt man Zählschleife</p>
	<p>wiederholt die Blöcke, die innerhalb stehen, so lange, bis die Variable ‚Index‘ von 0 startend, den hinteren Wert (hier 3) erreicht hat (hier wird der Inhalt also 4 Mal ausgeführt)</p> <p>nach jeder Ausführung der Blöcke innen (wenn unten angelangt), wird ‚Index‘ automatisch um 1 erhöht</p> <p>nennt man FOR-Schleife</p>
	<p>wiederholt die Blöcke, die innerhalb stehen, solange das Fragekästchen ‚wahr‘ ist (wie bei den Verzweigungen); wenn die Antwort ‚falsch‘ ist, werden die Wiederholungen beendet</p> <p>nennt man WHILE-Schleife</p>

Zählschleife

Mit der Zählschleife könnte man z.B. ein Quadrat genau 9 mal blinken lassen. Den Code dafür siehst du unten. Die Blöcke innerhalb des grünen Blocks werden genau neunmal ausgeführt, danach beginnt die Ausführung des dauerhaft-Blocks, sofern vorhanden.



Beispielcode VI.1 – Zählschleife

For-Schleife

Die For-Schleife funktioniert gleich wie die Zählschleife. Der Vorteil hier ist, dass man die aktuelle Wiederholungsnummer (Index) ausgeben kann. Bei dieser Schleifenart gibt es einen Startwert (immer 0) und einen angegebenen Wert (im Beispiel unten 8), welche die Anzahl der Wiederholungen bestimmen. Zuerst hat die Variable ‚Index‘ den Wert 0, nach jeder Ausführung der inneren Blöcke, wird diese automatisch um 1 geändert. Index hat also die Werte 0, 1, 2 usw. – bis zur angegebenen Zahl. Bei der letzten Wiederholung ist die Variable ‚Index‘ in unserem Beispiel 8. Es gibt hier insgesamt 9 Wiederholungen (0-8).

Mit der For-Schleife können wir das Programm von oben, wo die Rechtecke 9-mal blinken so erweitern, dass jedes Mal auch die aktuelle Wiederholungszahl ausgegeben wird. Man weiß, also wie oft das Rechteck noch blinken wird.



Beispielcode VI.2 – For-Schleife

While-Schleife

Die While-Schleife könnte man auch als „so lange bis“ Schleife verstehen. Es wird also der Code innerhalb so lange ausgeführt, bis das Fragekästchen ‚falsch‘ ist.

Mit der While-Schleife kann man den micro:bit z.B. am Beginn so lange blinken lassen, bis er geschüttelt wird. Wird er geschüttelt, wird das Blinken abgebrochen und das „normale“ Programm (im dauerhaft) startet. Damit könnte man z.B. ein Spiel erst mit Schütteln starten – nicht sofort, wenn der micro:bit Strom bekommt.

Wir benötigen eine Variable (hier heißt sie „Start“). Am Beginn setzen wir diese auf 0.

Wird der micro:bit geschüttelt, ändert sich „Start“ von 0 auf 1.

Überprüfe nun mit dem ‚Frage-Kästchen‘, ob „Start“ den Wert 0 hat. Und solange das gilt, wiederhole die Blöcke innerhalb der Schleife.

Wenn sich „Start“ auf 1 ändert, wird die Schleife abgebrochen und der dauerhaft-Block wird ausgeführt.

Beispielcode VI.3 – While-Schleife



Du kannst nun neue Arten von Programmen schreiben, schau dir dazu das zugehörige **Arbeitsblatt** an! Vergiss nicht dein Programm im Anschluss herunterzuladen und auf den micro:bit zu verschieben um es auszuprobieren!