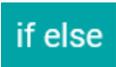


Grundlagen zur Programmierung des micro:bit mit Python III

Verzweigung (aus Logik)

Eines der wichtigsten Programmierkonzepte ist die **Verzweigung**. Damit kann man **Fallunterscheidungen** machen. Das hatten wir von der Idee schon früher (bei der Ereignissteuerung), dort konnten wir aber nur aus vorgegebenen Ereignissen auswählen. Nun können wir selbst bestimmen, was für uns ein Ereignis ist, bei dem etwas passieren soll. Weiters können diese Verzweigungen in anderen Ereignissen („Drücke Knopf A“, „Wenn geschüttelt“,...) vorkommen.

Um diese Fallunterscheidung (spezielles Ereignis oder nicht) zu machen, benötigen wir folgenden Befehl aus  :

Block	Beschreibung
 <p>if True: pass else: pass</p>	<p>Das ist eine Fallunterscheidung (Verzweigung); statt True kommt in deinem Programm eine Frage (ob ein Ereignis zutrifft), stimmt diese, wird der Code hier ausgeführt, stimmt diese nicht, dann wird dieser Code ausgeführt</p>
	<p>Statt dem True kann man mathematische Vergleiche hinschreiben. Diese Vergleiche haben entweder den Wert ‚wahr‘ oder ‚falsch‘, je nachdem ob die beiden Werte gleich sind. Damit das Programm auch weiß, dass du hier einen Vergleich machst und nicht eine Wertzuweisung, benutzt man zweimal das „=“;</p> <p>Auch andere Vergleiche (kleiner <, kleiner gleich <=, usw.) sind möglich.</p>

Will man nun den micro:bit zu einem Mini-Orakel machen, das einem bei Tastendruck zufällig ein Häkchen oder ein X anzeigt (für Ja bzw. Nein), so kann man das nun machen. Bisher konnten wir nur die Zufallszahl ausgeben und mussten uns merken, dass z.B. 0 Nein und 1 Ja bedeutet. Nun können wir Wörter oder auch Symbole anzeigen abhängig von der Zufallszahl.

Der auf der nächsten Seite abgebildete Code tut genau das: Es wird solange ein Fragezeichen angezeigt, bis am micro:bit die Tasten A und B gedrückt werden. Passiert das, dann wird eine Zufallszahl zwischen 0 und 1 ausgewählt und gleich überprüft, ob diese gleich 1 ist. Ist das der Fall (Zufallszahl=1), dann wird ein Häkchen angezeigt, ansonsten (Zufallszahl=0) ein X.

def on_forever():

```
basic.show_leds("""
.###.
...#.
..#..
.....
..#..
""")
```

Zeige ein Fragezeichen, solange nichts Anderes (A+B-Knopf gedrückt) passiert.

basic.forever(on_forever)

Wird A+B am micro:bit gedrückt, mache den Code darunter.

Wähle eine zufällige Zahl zwischen 0 und 1 aus.

def on_button_pressed_ab():

```
if randint(0, 1) == 1:
    basic.show_icon(IconNames.YES)
else:
    basic.show_icon(IconNames.NO)
```

Überprüfe, ob die gewählte Zufallszahl gleich 1 ist.

Wenn die Zufallszahl = 1 (die Frage ist wahr), dann zeige das Häkchen an.

input.on_button_pressed(Button.AB, on_button_pressed_ab)

Ist die Zufallszahl nicht = 1 (also hier 0), dann zeige das X an.



Du kannst deine Programme mit dem neuen Wissen über Verzweigungen erweitern oder ganz neue 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!