**Roboloco im echten Leben**

**Grundton**

Lernszenario

**AUTOR(EN)**

SOU Jane Sandanski-Strumica

**ZUSAMMENFASSUNG**

Verwendung von Grundgeräuschen mit dem Lego Mindstorm EV3 Roboter

**SCHLÜSSELELEMENTE**

| **ÜBERBLICK** | |
| --- | --- |
| **Themen** | Musik , Informatik |
| **Was soll getan werden?** | 1. Die SchülerInnen lernen, wie man grundlegende Geräusche im Lego Mindstorm EV3-Roboter verwendet und wie man Python bei der Verwendung des Roboters einsetzt. |
| **Zielpublikum** | SchülerInnen |
| **Vorbereitungszeit** | 15 min |
| **Unterrichtszeit** | 30 min |
| **Wo Sie diese Lektion herunterladen können (und mehr).** | RoboLocode-Unterrichtsmaterialien:  <https://teducativas.madeira.gov.pt/roboloco>  <https://ev3dev-lang.readthedocs.io/projects/python-ev3dev/en/2.0.0beta1/other.html> |
| **Was Sie brauchen** | Der LEGO Mindstorm EV3 |
| **Verwendete Ressourcen** | Der LEGO Mindstorm EV3 - nur Stein  Klassen [(](https://ev3dev-lang.readthedocs.io/projects/python-ev3dev/en/2.0.0beta1/other.html)https://ev3dev-lang.readthedocs.io/projects/python-ev3dev/en/2.0.0beta1/other.html)  [**https://sites.google.com/site/ev3python/learn\_ev3\_python/loudspeaker\_speech**](https://sites.google.com/site/ev3python/learn_ev3_python/loudspeaker_speech) |

**EINFÜHRUNG IN DEN UNTERRICHT**

**LEHRPLAN**

| **Übung** |
| --- |
| 1. **Anweisungen** 2. **1. Bereiten Sie Dateien im wav-Format vor. Nehmen Sie die Datei sound.zip von der Seite**   [**https://sites.google.com/site/ev3python/learn\_ev3\_python/loudspeaker\_speech**](https://sites.google.com/site/ev3python/learn_ev3_python/loudspeaker_speech) **finden Sie alle in das wav-Format konvertierten LEGO MINDSTORMS EDUCATION Dateien.**   1. **Erstellen Sie ein spezielles Verzeichnis namens "sounds" im Verzeichnis Ihres Roboters und kopieren Sie wav-Dateien hinein (vollständiger Pfad /home/robot/sound).** 2. **3. Erstellen Sie ein Programm, das den Roboter**  * **-eine wav-Datei oder ein beliebiges Tonsignal abspielen und den Satz "Willkommen beim E V 3-Entwicklungsprojekt" vorlesen,** * **-l ein Lied abspielen** |
| **Kurze Hilfe zur Programmierung** |
| **Für die Übung benötigte Befehle/Funktionen**  **Bibliothek:** [**ev3dev**](http://mumin.pl/Probot/_downloads/f1554763194447c0c438f34af286000d/Sound_lib_ev3dev.zip)  **Einen Standard-Piepton abspielen**  beep(<Frequenz in Hz>)  **Einen einzelnen Ton spielen**  tone(Frequenz in Hz,Dauer in Millisekunden)  **Abspielen einer Tonfolge**  tone(ton\_sequenz)  tone\_sequence - Liste von Tupeln.  Jedes Tupel hat die Form (Frequenz in Hz, Dauer in Millisekunden, Wartezeit in Millisekunden)  **Abspielen einer WAV-Tondatei**  play(*wav\_file*)  **Ein Lied abspielen**  play\_song(*song*, *tempo=120*, *delay=0.05*) song - Liste von Tupeln.  Jedes Tupel hat die Form (Notenname, Werte)  Der Notenname und sein Wert in konventioneller Notation (siehe  https://newt.phys.unsw.edu.au/jw/notes.html) für Frequenz und Dauer. Symbolische Noten sind zulässig (z. B. A4, D#3, Gb5). Für die Angabe der Dauer sollte w - ganze Note, h - halbe Note, q - Viertelnote, e - Achtelnote, s - Sechzehntelnote verwendet werden. Triolen sollten zum Beispiel so geschrieben werden: ('D4','e3'), ('D4','e3'), ('D4','e3'). Sprechen eines Textes speak(*text*)  **Beispiel:**  **#!/usr/bin/env python3**  **Sound.speak('Hallo')**  **oder**  **sound=Sound()**  **sound.speak('Hallo')**  **Bibliothek:** [**ev3dev2**](http://mumin.pl/Probot/_downloads/079228b4da187933c84ef53f32edc7eb/Sound_lib_ev3dev2.zip)  **Einen Standard-Piepton abspielen**  beep(<Frequenz in Hz>,play\_type=0)  **Einen einzelnen Ton spielen**  tone(Frequenz in Hz,Dauer in Millisekunden,play\_type=0)  **Abspielen einer WAV-Tondatei**  play\_file(*wav\_file*, *volume=100*, *play\_type=0*)  **Ein Lied abspielen**  play\_song(*song*, *tempo=120*, *delay=0.05*)  Analog wie oben  ***#!/usr/bin/env python3***  **Sprechen Sie einen Text**  **speak(text, speak\_opts='-a 200 -s 130', volume=100, play\_type=0)**  **a = Amplitude (200 max, 100 Standard), s = Geschwindigkeit 80-500, Standard = 175**  **abspielen\_type**  **hat zwei Optionen**  **| Sound.PLAY\_WAIT\_FOR\_COMPLEE=0 (| Standard)**  **Sound.PLAY\_NO\_WAIT\_FOR\_COMPLETE=1**  Das Verhalten, nachdem die Wiedergabe gestartet wurde. Wir verwenden wait() nicht nach diesem Befehl**.**  **Beispiel**  **#!/usr/bin/env python3**  **Sound().speak('Hallo',volume=50)**  **oder**  **sound=Sound()**  **sound.speak('Hallo',volume=50)** |

**BEWERTUNG**

Um die Lektion zu bewerten, sollte den Schülern eine einfache Frage gestellt werden.

Was haben Sie bei dieser Übung gelernt?

**SCHÜLER-FEEDBACK**

Nach dem Ende der Stunde können die Schüler ihr Feedback abgeben.

**LEHRERBEMERKUNGEN**

*Fügen Sie hier Ihre Kommentare und Bewertungen* ***NACH der*** *Durchführung dieser Lektion ein, falls vorhanden.*