# EV3 Python


- Einführung
- Motoren

# Einführung

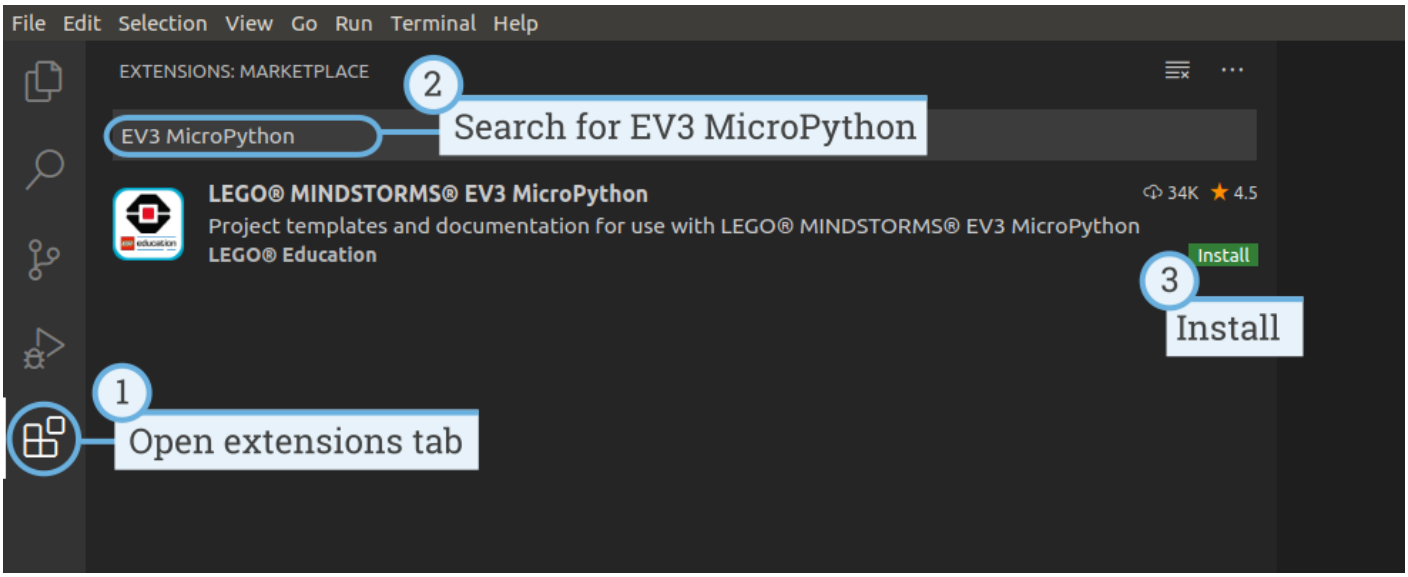
Wir programmieren den Lego-Roboter (auch EV3 genannt) in Micropython. Micropython ist eine spezielle Version der Programmiersprache Python, die speziell für den EV3 angepasst ist und dadurch etas schneller läuft. Es sind alle normalen Python-Befehle verfügbar, aber eventuell funktionieren nicht alle Bibliotheken. Eine genaue Dokumentation ist hier verfügbar (auf Englisch):

https://pybricks.com/ev3-micropython/

# Einrichtung der Programmierumgebung

Für die Programmierung verwenden wir ein Programm das **Visual Studio Code** heißt. Das Programm sollte auf allen Rechnern für die Robotics bereits vorinstallierte sein. Alternativ kann man sich das Programm auch hier runterladen: https://code.visualstudio.com.

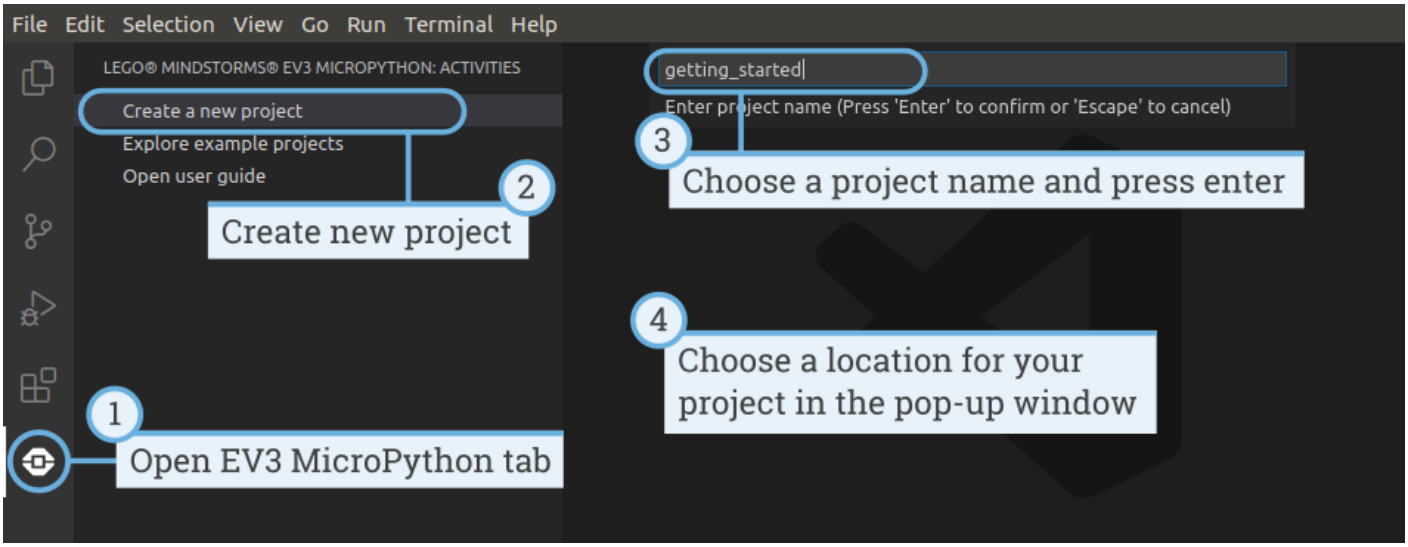
Um Programme für den EV3 zu schreiben und diese auf den Roboter zu Laden wird noch eine spezielle **Erweiterung** gebraucht. Diese kann wie folgt installiert werden:

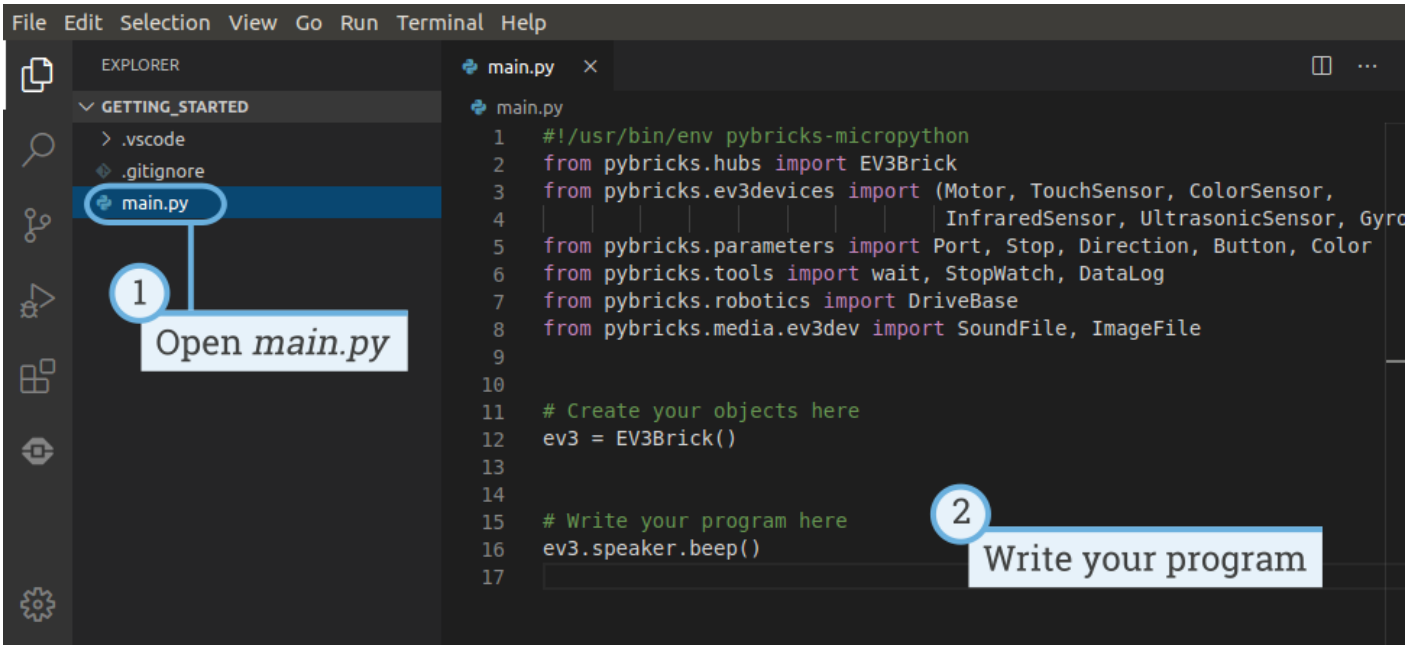



# Ein neues Projekt anlegen

Ist die **Erweiterung** installiert findet man auf der linken Seite ein Symbol das Symbol vom EV3. Mit einem Klick auf das Symbol ändert sich die linke Seite und zeigt drei Optionen an (auf Englisch, das

ist die Standard-Sprache in der Programmierung). Mit einem Klick auf `Create a new project` öffnet sich oben eine Leiste in die man den Projektnamen eingibt (z.Bsp. "robot_team1"). Mit einem Klick auf Enter wird der Name bestätigt. Dadurch öffnet sich ein extra Fenster, über das man auswählt wo das Programm gespeichert werden soll. Nun einfach einen Ort auf dem Computer aussuchen, den man später auch wieder findet. Nach Auswahl des Ortes aktualisiert sich die gesamte Oberfläche und man sieht auf der linken Seite jetzt eine Liste von Dateien, die automatisch erzeugt wurden. Wir fahren fort in dem wir die `main.py` öffnen.




# Code-Beispiel

Beim Anlegen des Projektes wurde bereits etwas Code automatisch für uns generiert:

```
#!/usr/bin/env pybricks-micropython
from pybricks.hubs import EV3Brick
```

```
from pybricks.ev3devices import (Motor, TouchSensor, ColorSensor,
                                 InfraredSensor, UltrasonicSensor, GyroSensor)
from pybricks.parameters import Port, Stop, Direction, Button, Color
from pybricks.tools import wait, StopWatch, DataLog
from pybricks.robotics import DriveBase
from pybricks.media.ev3dev import SoundFile, ImageFile
```

Dieser Code bleibt in jedem Programm gleich. Er importiert weiteren Code, der nicht von uns geschrieben wurde, aber uns bei der Programmierung hilft. Wie das genau funktioniert ist aktuell nicht wichtig, wir behalten diesen Code einfach.

```
# Create your objects here
ev3 = EV3Brick()
```

Hier sagen wir dem Computer (der Roboter ist auch nur ein Computer in klein) was wir alles benutzen wollen. Aktuell benutzen wir nur den Brick.

```
# Write your program here.
ev3.speaker.beep()
```

Hier benutzen wir den Brick von oben (wichtig: wir müssen den gleichen Namen `ev3` verwenden!!) und sagen, dass wir den Lautsprecher (englisch "speaker) einmal piepsen lassen wollen (englisch "beep").

Als nächsten bringen wir dem Roboter bei zu fahren □□

# Motoren

Wir haben unser erste Projekt erstellt und den Roboter piepsen lassen. Jetzt wollen wir den Roboter fahren lassen. Alles was wir dazu brauchen ist ein Roboter mit zwei Motoren (ob Räder oder Ketten ist hier erstmal egal).

# Motoren definieren

Damit wir die Motoren verwenden können, müssen wir dem Computer erst sagen an welche Ports diese angeschlossen werden. Das machen wir wie folgt:

```
# Create your own objects here.
ev3 = EV3Brick()
motor_left = Motor(Port.A)
motor_right = Motor(Port.B)
```

Wir haben zwei Motoren definiert die wir verwenden wollen: `motor_left` (englisch "left" = "links") und `motor_right` (englisch "right" = "rechts). Außerdem haben wir definiert dass der linke Motor an Port A angeschlossen ist, und der rechte Motor an Port B. Ist der Roboter anders aufgebaut, dann müssen wir andere Ports verwenden.

Um die Räder jetzt zu verwenden müssen wir noch eine kleine Änderung machen: Wir müssen oben in der Datei noch ein `import time`

```
...
from pybricks.tools import wait, StopWatch, DataLog
from pybricks.robotics import DriveBase
from pybricks.media.ev3dev import SoundFile, ImageFile
import time
```

# Motoren steuern

Jetzt sind wir startklar!

```
# Write your program here.
ev3.speaker.beep()
```

```
motor_left.run(200)
motor_right.run(-200)
time.sleep(2.5)
motor_left.stop()
motor_right.stop()
```

Diese Programm lässt den linken motor vorwärts drehen, den rechten Motor rückwärts drehen, wartet 2.5s und stoppt dann beide Motoren wieder.

Fragen:

- Warum lassen wir einen Motor vorwärts und den anderen Motor rückwärts drehen?
- Was passiert wenn wir die Zahlen verändern? Was wenn es unterschiedliche große Zahlen sind?
- Wie dreht der Roboter, und wie fährt er gerade aus?

# Im Viereck fahren