Lektion 7: micro:bit mit Python **Übung 3: Lichtmessungen**

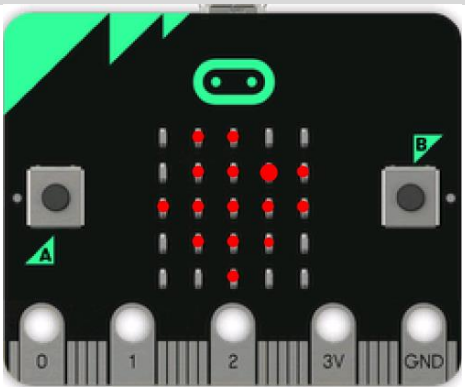
In dieser Lektion verwenden Sie den Lichtsensor auf dem micro:bit und speichern die Daten in einer TI-Nspire™-Liste zur weiteren Analyse.

Lernziele:

- Auslesen und Anzeigen des Lichtsensors auf dem micro:bit
- Daten mittels Python zum TI-Nspire™ CX II-T übertragen
- Untersuchen der gewonnenen Daten vom micro:bit

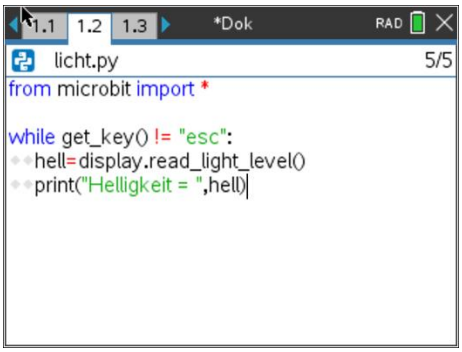
Lehrerhinweis: In dieser Lektion wird das inverse quadratische Gesetz für die Abhängigkeit der Beleuchtungsstärke von der Entfernung von einer (punktförmigen) Lichtquelle angenähert. Statt der Entfernung wird dabei eine gleichmäßige Bewegung weg von der Lichtquelle verwendet, die eine gute Näherung für die Entfernung darstellt.

1. Der micro:bit kann die Umgebungslichtstärke über die Display-LEDs ablesen. Die Display-LEDs können also auch als Eingabegerät verwendet werden!



Hinweis: mehr Information über den Lichtsensor gibt es unter
Sensing changes in light on the micro:bit : Help & Support

2. Starten Sie ein neues Python-Programs in einem neuen Dokument.
   Dazu drücken Sie die **[home]** - Taste und wählen dann **Neu >** und **Add Python >Neu…**
   Das Programm heißt **licht..**
   Um die Werte zu überprüfen, die der Lichtsensor erkennen kann, schreiben Sie das folgende Kurzprogramm mit Hilfe der BBC micro:bit-Menüs:

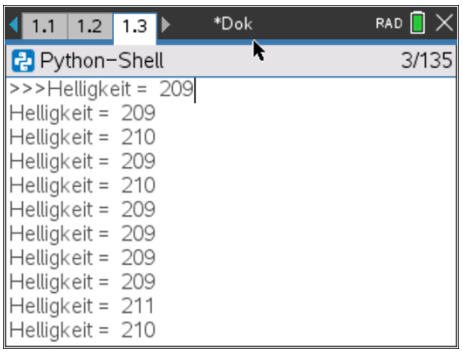



```
from microbit import *

while get_key() != "esc":
    hell = display.read_light_level()
    print("Helligkeit = ",hell)
```

   Man findet **var= .read_light_level()** im **[menu] > BBC micro:bit >Sensors ; hell** ist der selbstgewählte Variablenname.
   Erinnerung: **[menu] > More Modules > BBC micro:bit > Commands> while get_key() != "esc":**
   Denken Sie daran, die letzten beiden Anweisungen so einzurücken, dass sie der while-Schleifentext sind .

3. Führen Sie das Programm aus und richten Sie das Display des micro:bit auf eine Lichtquelle. Es spielt keine Rolle, was auf dem Display angezeigt wird. Bewegen Sie das micro:bit in Richtung und weg vom Licht und beobachten Sie die sich ändernden Werte auf dem TI-Nspire™ CX II-T-Bildschirm. Sie sollten Werte sehen, die zwischen 0 und 255 variieren.
   Wie Sie wahrscheinlich erwarten, nimmt die Helligkeit mit wachsender Entfernung von der Lichtquelle ab. Jetzt fügen Sie dem Programm Code hinzu, um die Lichtpegeldaten zu sammeln und dann ein Streudiagramm von Helligkeit gegen die Zeit zu erstellen.

   Drücken Sie **[esc],** um das Programm zu beenden und zum Editor zurückzukehren.

4. Zunächst müssen zwei **leere** Listen am Programmanfang erstellt werden.:

   zeit = [ ]
   bel = [ ]

   Die eckigen Klammern findet man auf der Tastatur oder in **[menu] > Eingebaute > Listen**

   Ebenfalls vor der Schleife muss noch der Startpunkt der Zeit festgelegt werden:

   t = 0

   Vermeiden Sie es, das Wort "time" als Variable zu verwenden, da es im Zeitmodul verwendet wird.

5. Fügen Sie im Schleifentext nach der **print-Anweisung** eine Anweisung hinzu, um die Timervariable **t** zu erhöhen. Wir verwenden ein Zeitintervall von einer Sekunde zwischen den Lichtmessungen, also verwenden Sie:

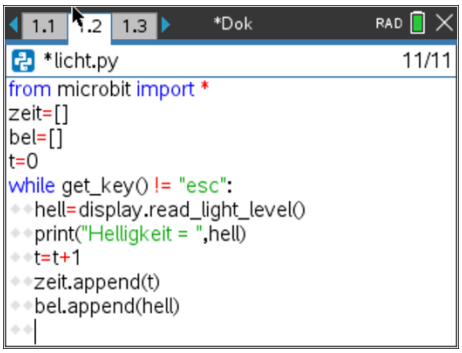
   t = t + 1

6. Ergänzen der Listen **bel** und **zeit** mit ihren aktuellen Werten durch die Anweisung **.append()**:

   ♦♦**times.append(t)**
   ♦♦**brights.append(bright)**

   .**append( )** findet man in **[menu] > Eingebaute > Listen**



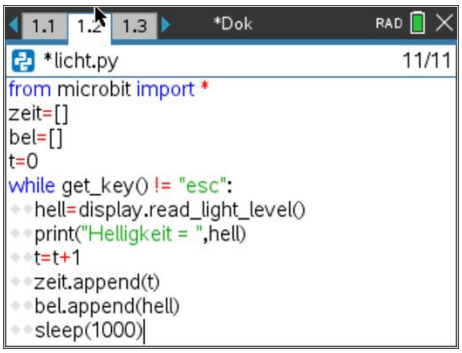
7. Damit im Abstand von 1 s neue Werte ermittelt werden, muss die Programmausführung zwischendurch „schlafen“:

   ♦♦**sleep(1000)**

   sleep( ) findet man in:

   [menu] > BBC micro:bit > Commands

   Denken Sie daran, dass **sleep( )** bei der Verwendung von micro:bit *Millisekunden für sein Argument verwendet 1000 ms = 1 s .*



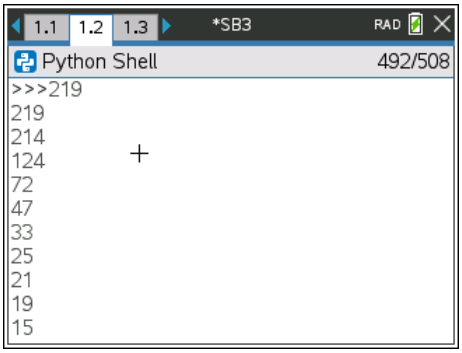
6. Nach dem Schleifenkörper (*kein Einrücken mehr!*), **speichern** Sie die beiden Python-Listen in zwei TI-Nspire™-Listen mit den gleichen Namen in beiden Umgebungen.
   Beginnen Sie am *Anfang* einer neuen Zeile (kein Einzug!).
   Finden Sie **store_list()** in **[menu] > BBC micro:bit > Commands**. Es braucht zwei Argumente: den "TI-Nspire™-Listennamen" in Anführungszeichen und den Python-Listennamen ohne Anführungszeichen. Folgen Sie den QuickInfo-Anweisungen.

   In diesem Programm werden die Python-Listen in TI-Nspire™*-Listen gespeichert, kurz bevor das Programm endet, wenn Sie also* **[esc]** *drücken, um die Schleife zu verlassen*

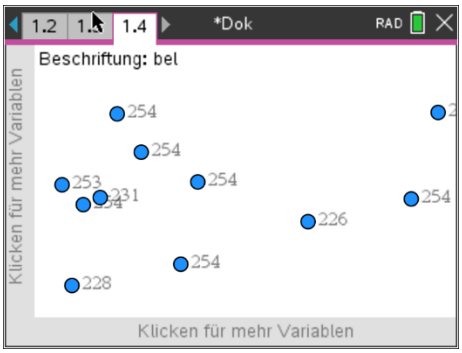
6. Führen Sie das Programm aus. Beginnen Sie mit dem micro:bit in der Nähe Ihrer Lichtquelle. Eine freiliegende Glühbirne oder eine Smartphone-Taschenlampe funktionieren gut. Bewegen Sie das micro:bit langsam aber *stetig* gleichmäßig vom Licht weg, bis die Helligkeit unter 10 liegt.
   Drücken Sie **[esc],** um das Programm zu beenden.

   Wiederholen Sie den Vorgang, bis Sie das Gefühl haben, dass Sie "gute" Daten haben.

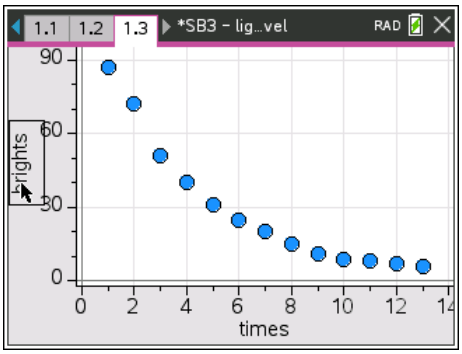
   Tipp: Eine Smartphone-Taschenlampe funktioniert super!



6. Wenn das Programm beendet ist, fügen Sie Ihrem Dokument eine Seite hinzu, indem Sie **[ctrl] [doc]** drücken. Wählen Sie **"Daten & Statistiken hinzufügen".**
Sie sehen eine Seite ähnlich diesem Bild. Eine Ihrer Listen wird mit den auf dem Bildschirm verstreuten Punkten angezeigt. Sie müssen nun die Darstellung organisieren...



7. Klicken Sie unten auf dem Bildschirm auf die Eingabeaufforderung '*Klicken Sie hier, um eine Variable hinzuzufügen*' und wählen Sie aus der Liste die **Zeit** aus. Dies ist die *unabhängige* Variable.
Wählen Sie auf der linken Seite des Bildschirmes die Liste **hell** aus**.** Dies ist die *abhängige* Variable. Jetzt könnte Ihr Streudiagramm wie das nebenstehende Bild aussehen.
Möglicherweise müssen Sie das Programm *mehrmals* ausführen, um eine gute, glatte Darstellung wie die hier gezeigte zu erhalten.
Möglicherweise müssen Sie auch das Anzeigefenster anpassen, wenn Sie andere Daten verwenden.



Sie können auch das Zeitintervall zwischen den Stichproben anpassen, aber stellen Sie sicher, dass Sie sowohl das Argument **sleep(1000)** als auch *den Zählerwert* ***(t = t + 1)*** *bearbeiten.*

Was fällt Ihnen auf?

- *Welches Muster sehen Sie? Welches mathematische Modell passt zu diesem physikalischen Phänomen?*
- *Verwenden Sie die Datenanalysewerkzeuge des* TI-Nspire™ CX II-T*, um ein gutes mathematisches Modell für Ihre Daten zu ermitteln.*

Lehrertipp: Die Lektion untersucht das inverse quadratische Gesetz für die Beleuchtung von einer Lichtquelle, aber das Gesetz ist eine Beziehung zwischen der Entfernung von der Lichtquelle und der Lichtintensität, nicht der Zeit.

Intensität = k / (Entfernung2)**

Eine vorsichtige, konstante Bewegung des micro:bit von der Lichtquelle weg stellt eine lineare Beziehung zwischen Zeit und Entfernung her, so dass das inverse quadratische Verhältnis weiterhin anwendbar ist.

Sie können die Aktivität so ändern, dass eine Entfernungsmessung von der Lichtquelle zum micro:bit verwendet wird, und einen Tastendruck verwenden, um jeden Datenpunkt zu erfassen. Bewegen Sie beispielsweise das micro:bit in jedem Sammelmoment 10cm weiter vom Licht weg. Verwenden Sie **get_key(1),** um das Programm anzuhalten, bis das micro:bit für das nächste Beispiel richtig positioniert ist.