



In dieser Übung werden wir einen automatischen Lichtschalter entwickeln, der auf das Umgebungslicht reagiert. Das Licht wird ausgeschaltet, wenn es dunkel wird und wieder eingeschaltet wenn es hell wird.

#### Lernziele:

- **READ BRIGHTNESS**
- Eine **While**-Schleife verwenden
- **If...Then...Else...End** einsetzen, um die LED abhängig von der Helligkeit ein- oder auszuschalten

Wir werden ein Programm schreiben, das den BRIGHTNESS-Wert misst und ein Licht einschaltet, wenn es „dunkel“ wird. Bei steigender Raumbelichtung geht das Licht wieder aus. Genau so arbeiten viele automatische Lichtschalter und Nachtbeleuchtungen.

Unser Programm wird den Lichtsensor des TI-Innovator™ Hub auslesen und das eingebaute Licht einschalten (**LIGHT ON**), wenn die Helligkeit unter einen bestimmten Wert fällt und wieder ausschalten, wenn sie über diesem Wert ist.

#### Der Beginn des Programms:

1. Beginne mit einem neuen Programm und nenne es *hell2*.
2. Füge **Disp** mit dem Text „Auto-Licht“ in Anführungszeichen an
3. Gib der lokalen Variablen **b** den Anfangswert 2 mit **b:=2**.
4. Füge eine **While...EndWhile**-Schleife mit der Bedingung **b>1** an. (Der Helligkeitswert liegt über „sehr niedrig“).

```

Define hell2()=
Prgm
Local b
Disp "Auto-Licht"
b:=2
While b>1
EndWhile
EndPrgm
  
```

Zur Beendigung der Schleife und des Programms ist der Sensor abzudecken.

5. Trage in den Körper der **While**-Schleife **Send "READ BRIGHTNESS"** und **Get b** aus dem **Hub**-Menü ein, wie rechts gezeigt.
6. Schreibe noch **Disp b** dazu. Dann kannst du auch die Messwerte im *Calculator* ablesen.

```

Define hell2()=
Prgm
Local b
Disp "Auto-Licht"
b:=2
While b>1
Send "READ BRIGHTNESS "
Get b
Disp b
EndWhile
EndPrgm
  
```

#### If-Anweisungen

Unsere **If**-Anweisung wird zwei Anweisungsblöcke haben: einen für den Fall, dass die Bedingung wahr ist und einen für den anderen Fall.

Die Struktur dieser *mehrzeiligen* Anweisung ist:

```

If <Bedingung> Then
  <mache dies, wenn Bedingung wahr (erfüllt) ist>
Else
  <mache dies, wenn Bedingung falsch (nicht erfüllt) ist>
EndIf
  
```

Du kannst überall weitere Leerzeilen einfügen (drücke einfach **enter**).

```

Define hell2()=
Send "READ BRIGHTNESS "
Get b
Disp b
If Then
Else
EndIf
EndWhile
EndPrgm
  
```

**Hinweis:** Beachte, dass **Else** und **EndIf** in eigenen Zeilen stehen.



## 10 Minuten Coding

### TI-NSPIRE™ CX MIT DEM TI-INNOVATOR™ HUB

#### Nun legen wir die Bedingung fest ...

Der Helligkeitswert wird in der Variablen **b** gespeichert und liegt zwischen 0 und 100.

Was ist ein guter Wert für „dunkel“? Wir wählen 25, aber du kannst jeden Wert zwischen 0 und 100 wählen.

Du kannst das Programm verbessern, indem du diesen „Einschaltwert“ über eine **Request**-Anweisung ermittelst. **Request** aber muss vor der **While**-Schleife stehen.

Hole das ‘<’ (kleiner)-Zeichen über die **ctrl** **=** Tasten.

7. Dann schalte **LIGHT ON** oder **OFF** in den **Then** und **Else**-Teilen wie gezeigt.
8. Speichere mit **ctrl** **B** das Programm.
9. Verbinde den TI-Innovator™ Hub mit dem Rechner und starte das Programm.
10. Verändere das Licht, das auf den Sensor fällt und beobachte, wie sich das LIGHT (die rote LED am Hub) ein- und ausschaltet.

Es könnte hilfreich sein, die Anweisung **Disp b** nach dem **Get b** ins Programm aufzunehmen um den Wert von **b** im *Calculator* zu zeigen und zusätzliche **Disp**-Anweisungen in die **Then** und **Else**-Blöcke einzubauen, um anzuzeigen, ob die LED „ON“ oder „OFF“ ist.

Um den Schleifendurchlauf (und das Programm) zu beenden muss man den Sensor ganz abdecken, so dass der Messwert unter 1 fällt.

## LEKTION 3: ÜBUNG 2

### LEHRERINFORMATION

```
* hell2
Send "READ BRIGHTNESS "
Get b
Disp b
If b < 25 Then
Else
EndIf
EndWhile
```

```
"hell2" erfolg. gespeichert
While b > 1
Send "READ BRIGHTNES
Get b
Disp b
If b < 25 Then
Send "SET LIGHT ON"
Else
Send "SET LIGHT OFF"
EndIf
EndWhile
```