



### Unit 3: BRIGHTNESS, IF, and WHILE

### Skill Builder 2: BRIGHTNESS & LIGHT with IF, WHILE

In this lesson, we'll develop an automatic light switch that responds to the ambient light, turning on when darkness falls and turning off when the amount of light rises.

#### Objectives:

- **READ BRIGHTNESS**
- Use a **While** loop
- Use **If...Then...Else...End** to turn the light on or off depending on BRIGHTNESS.

Let's now write a program that detects the BRIGHTNESS value and turns a light on when it gets 'dark'. When the room lighting gets brighter, the light goes off. This is exactly how many automatic light switches and night lights work.

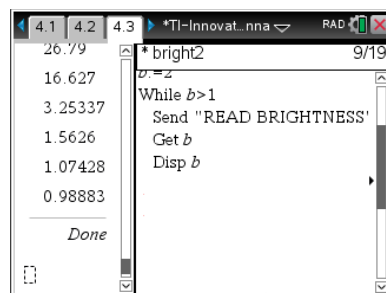
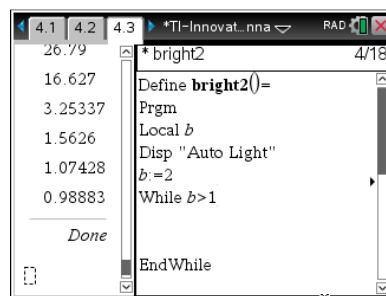
Our program will read the light sensor on the TI-Innovator™ Hub and turn the onboard **LIGHT ON** whenever the brightness value falls below a certain level and turn it off when the brightness is above that level.

#### Setting up the Program:

1. Start a new program, and call it BRIGHT2.
2. Add **Disp**, a set of quotations marks, and the text Auto Light.
3. Initialize the variable **b** by adding the statement **2→b**.
4. Add a **While...EndWhile** loop with the condition **b>1**. (The brightness value is more than 'very low'.)

To terminate the loop and the program, cover the light sensor.

5. In the **While** loop body, add **Send "READ BRIGHTNESS"** and **Get b** from the **HUB** menu as shown.
6. Add **Disp b** to see the reading.



#### If Statements

Our **If** statement will have two 'blocks' of code: one for when the condition is true and another for when the condition is false.

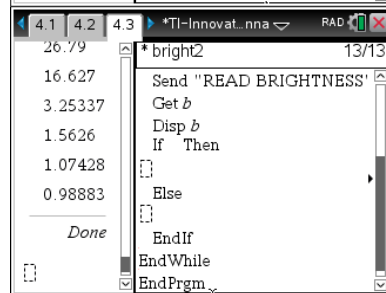
The structure of the *multi-line* statement is

**If** <condition> **Then**  
    <do this when true>

**Else**  
    <do this when false>

**EndIf**

You can add more blank lines anywhere by pressing [enter].





## 10 Minutes of Code

### TI-NSPIRE CX WITH THE TI-INNOVATOR™ HUB

#### Now for writing the condition...

The brightness value is stored in the variable **b** and ranges from 0 to 100.

What is a good 'dark' value? We chose 25, but you can change it to any value between 0 and 100.

You could improve the program by using a **Request** statement for this 'trigger' value. Just be sure to use the **Request** statement before the **While** loop starts.

Get the '<' (less than) operator from ctrl-= key

7. Finally, turn the **LIGHT ON** or **OFF** in the **Then** and **Else** blocks as shown.
8. Press **ctrl-B** to store the program.
9. Run the program with the TI-Innovator Hub attached.
10. Control the light hitting the sensor, and watch the LIGHT (the red LED on the Hub) turn on or off.

It might help to add a **Disp b** statement *after* the **Get b** statement to display the value of **b** on the calculator app and add statements in the **Then** and **Else** blocks to show whether the light is "ON" or "OFF".

To stop the loop (and the program), cover the light sensor completely so that the reading falls below 2.

## UNIT 3: SKILL BUILDER 2

### STUDENT ACTIVITY

```
4.1 4.2 4.3 *TI-Innovat... RAD 8/13
26.79
16.627
3.25337
1.5626
1.07428
0.98883
Done
Send "READ BRIGHTNESS"
Get b
Disp b
If b < 25 Then
Else
EndIf
EndWhile
EndPrgm
```

```
4.1 4.2 4.3 *TI-Innovat... RAD 12/13
26.79
16.627
3.25337
1.5626
1.07428
0.98883
Done
Send "READ BRIGHTNESS"
Get b
Disp b
If b < 25 Then
Send "SET LIGHT ON"
Else
Send "SET LIGHT OFF"
EndIf
EndWhile
EndPrgm
```