Unit 2: For Loops — Skill Builder 1: Blink the LIGHT

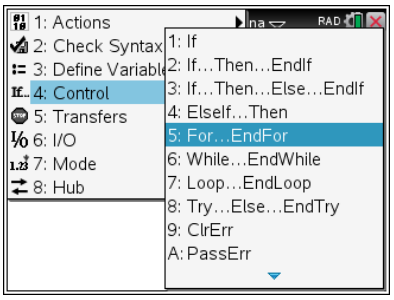
In this first lesson for Unit 2, you will learn about the **For** loop in the handheld through a program that makes the LED light blink while displaying information on the handheld screen.

Objectives:

- Learn about the **For** loop
- Make the light blink
- Use the **DispAt** statement for text and variables

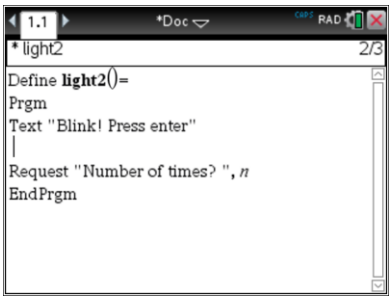
Let's write a program to make the LED light blink a certain number of times. With other input statements (or arguments) you can also control the amount of time that the light is on and off.

This program introduces you to the **For…EndFor** loop.



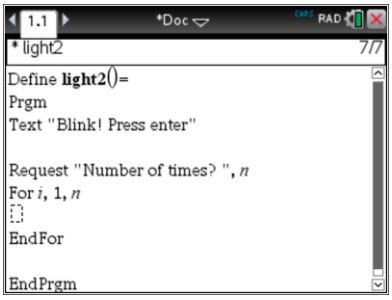
Setting up the Program

1. Start a new program, and name it light2.
2. Add **Text** by selecting **menu > I/O** and selecting **Text**.
3. Within quotation marks, add the text "*Blink! Press enter"* as shown.
4. Add **Request** by selecting **menu > I/O > Request.**
5. Within quotation marks, add the text "*Number of times?"* as shown.
6. Add a comma and the variable *n*.



Adding the For Loop

1. Add the **For** structure by selecting **menu > Control > For…EndFor**.
   - Both the **For , ,** statement and the **EndFor** statements are pasted into your program with a line between them to add the loop body.
2. Add the rest of the **For** statement's components, ***i***, **1**, ***n*** between the commas provided.
   - This statement means "For **i** going from **1** to **n** by ones."

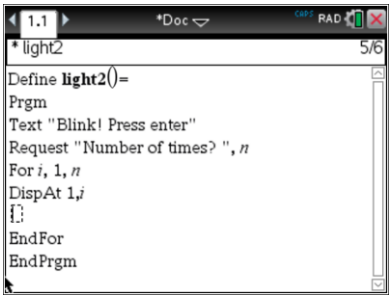


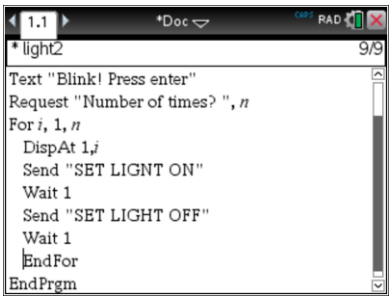
3. Press **enter** a few times in the loop body to create blank lines that we'll fill in next.
   - Don't worry about how many blank lines to enter. You can always add more lines if you need them, and blank lines have no effect when you run the program.
   - The block of statements between For and EndFor is called the 'loop body'. It is this section of code that will be processed *n* times thanks to the work done by the **For** loop.

We want the light to blink ON and OFF *n* times. We also want the handheld to display the number of the blink.

Start the loop body with **DispAt 1, *i*** which is the loop control variable.

DispAt *displays the expression following the comma at the line number in front of the comma. The output area of the program is below the program name and there are 8 lines available for* **DispAt** *to use.*





Now add statements to turn the light ON and OFF as shown.

1. Add **Send "SET LIGHT** from the **Hub** menu.
2. Add the word **ON** from the **Hub > Settings** menu (or just type it).
3. Add **Wait** (in seconds) from the **Hub** menu so that the handheld waits before sending the next command.
   - In our program, we use 1 second, but you can use any values you like, including decimals.
4. Add another **Send** statement to turn the light OFF, and then add another **Wait** statement.
5. Run the program (**ctrl+R**). You should see the light blink and the counter increase on the Calculator app.

Notice the indenting of the loop body? This helps to make programs more readable but has no effect on the running of the program.

Here's a challenge: Add **Request** statements at the top of the program (before the **For** statement) to set the timing for each of the two **Wait** values (the ON time and the OFF time). Use those variables instead of numbers in the Wait statements.

Run the program again. Observe the blinking and the values displayed on the handheld screen.