

INTERMEDIATE PROGRAMMING LESSON



EV3 CLASSROOM: VARIABLES

By Sanjay and Arvind Seshan



EV3 CLASSROOM LESSON
BY EV3LESSONS.COM

Objectives

1. Learn about different types of variables
2. Learn how to read and write to variables

Prerequisites: Color Sensor, and Display Blocks, Wait blocks

Variables

What is a variable? Ans. A variable stores a value that you can use later in your program. Think of it like a notepad or a box that holds a value for you.

You can name the variable whatever you want

You can define the type of variable:

- Variable (Holds a number)
- List (Holds a set of numbers ... 1,2,3,10,55)

You can either....

- Write – put a value into the variable
- Read – retrieve the last value written to the variable

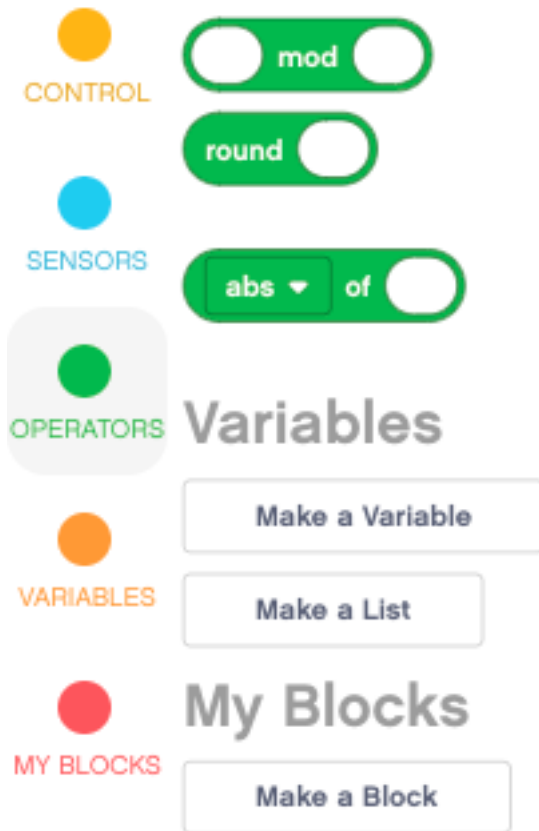
Why Variables?

Variables are an easy way to transfer data across code

You can also use variables to transfer data into a My Block without an input (eg. A variable for wheel size in Move Inches – You probably do not want this to be an input since it rarely changes. You may also use the value in other locations and want to change it just in one spot.)

List variables can store multiple data items and make it easy to process all of them. We will cover list variables in a separate lesson in the advanced section.

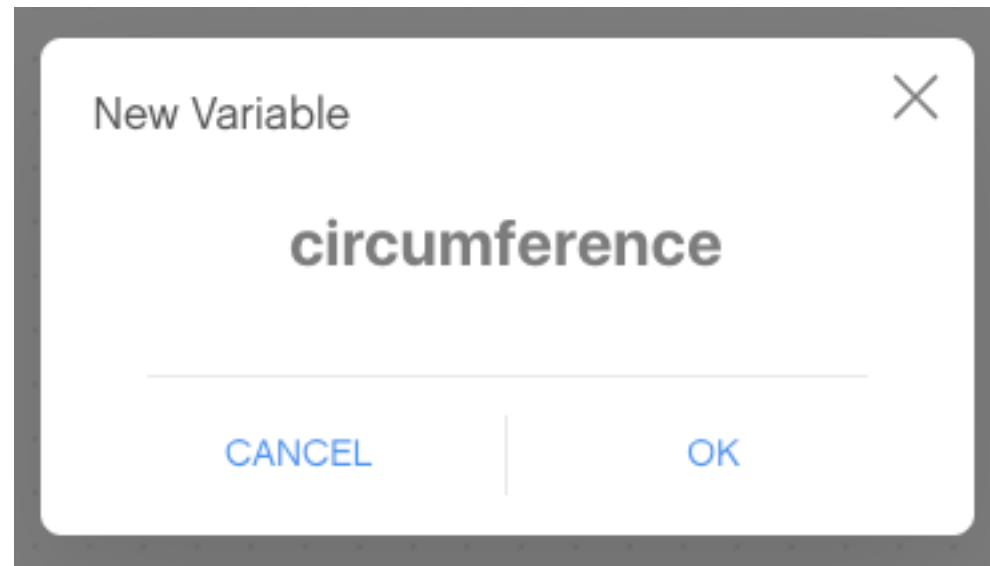
Creating a Variable



To create a variable, scroll down to the Variables section

Select Make a Variable and Name it.

In the example below, a variable called “circumference” has been made



Writing to a Variable

Once you have created the variable, it will appear in the menu bar.

Variables

Make a Variable

circumference

set circumference to 0

change circumference by 1

Make a List

My Blocks

Make a Block

In the example below, Circumference is set to the circumference of the EV3 Educator robot wheel in centimeters.

Circumference = Pi X Wheel Diameter

Circumference = 3.14 X 5.6

This can be calculated using a Math Block

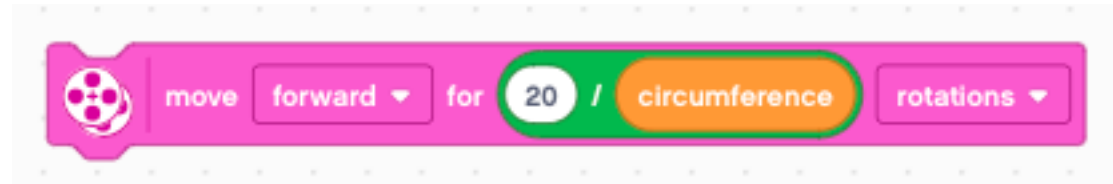
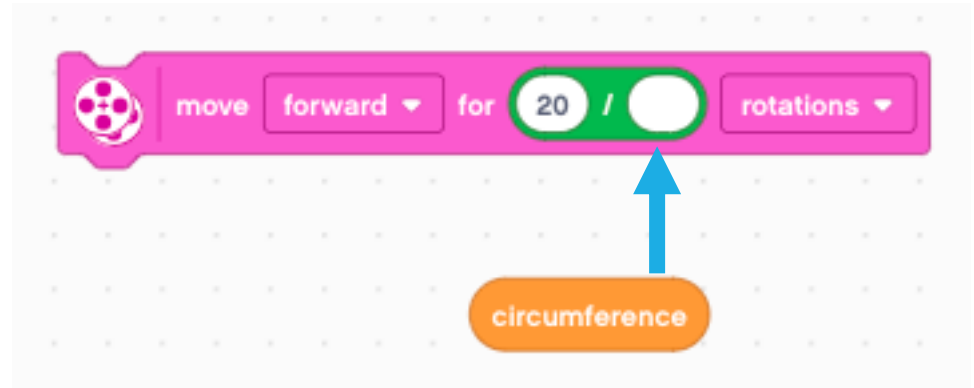
set circumference to 3.14 * 5.6

Reading a Variable

The variable can now be used in any block with an oval shape operator where you would normally type in a value

In the example on the right, the circumference is used to move the robot forward 20 centimeters (20 CM/Centimeters in a Circumference)

For example, if the circumference was 10CM, the robot would have to move 2 rotations to move 20CM.

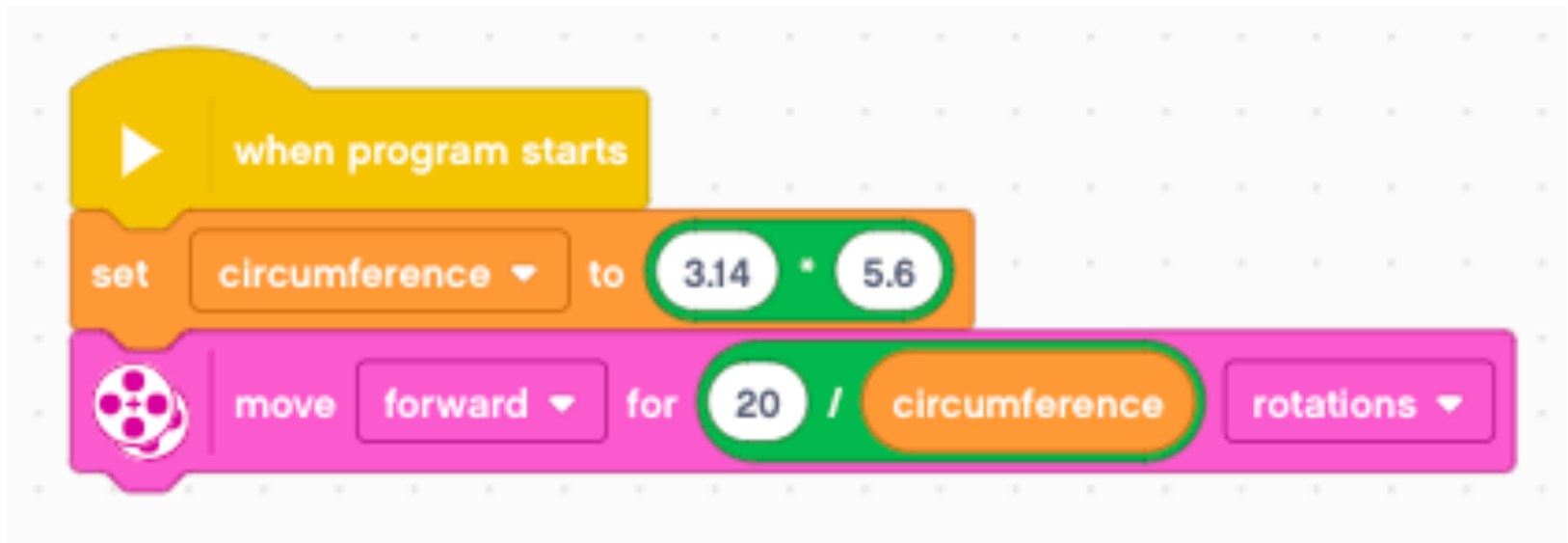


Putting it all together

In this example, the program moves 20CM

First set the variable “circumference” before using it in the program

Use the variable in the movement block



Changing Variables

Once you have created the variable, it will appear in the menu bar.

Variables

Make a Variable

circumference

set circumference to 0

change circumference by 1

Make a List

My Blocks

Make a Block

In the example below, counter is initialized to 1. The change by 2 will add 2 to the counter.

The display block will show a 3 on the screen since $1 + 2 = 3$

Note that you can change by a negative number as well – this will subtract from the variable.

The image shows a Scratch code editor with three blocks on a grid background. The first block is an orange 'set' block with 'counter' in the dropdown and '1' in the input field. The second block is an orange 'change' block with 'counter' in the dropdown and '2' in the input field. The third block is a purple 'write' block with 'counter' in the dropdown, '50' in the first input field, '50' in the second input field, and 'large black' in the font dropdown.

Challenges

Challenge 1:

- Can you make a program that displays the number of times

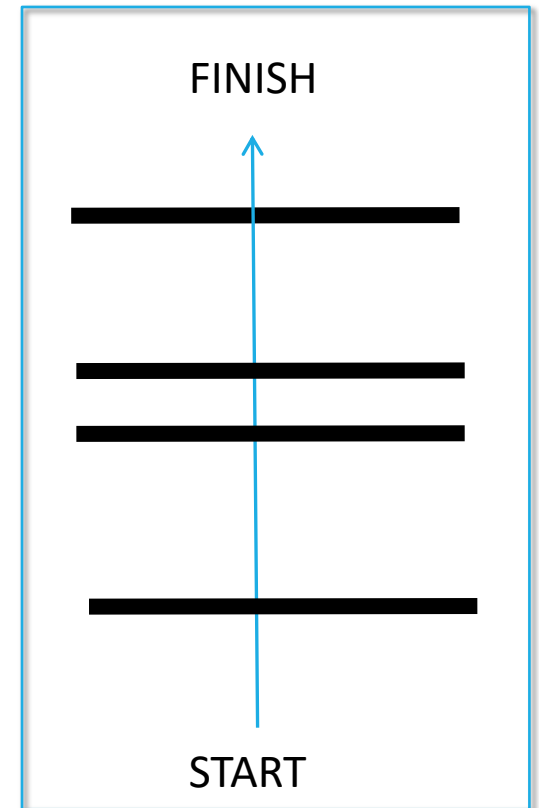
Challenge 2:

- Can you make a program that displays the number of times that you have clicked the up button?

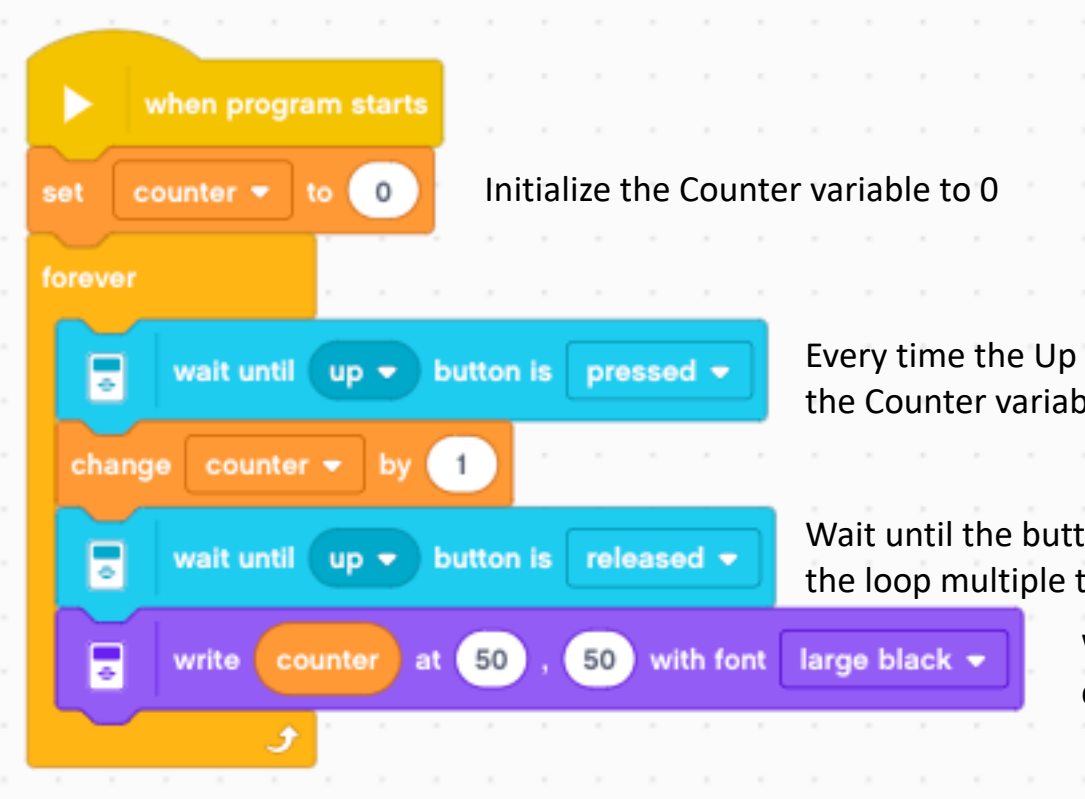
Challenge 3:

- Can you write a program that counts the number of black lines you have crossed?

Challenge 2



Challenge 1 Solution: Count Clicks



The image shows a Scratch script on a grid background. It starts with a yellow 'when program starts' block. Below it is an orange 'set counter to 0' block. A yellow 'forever' loop block contains four blocks: a light blue 'wait until up button is pressed' block, an orange 'change counter by 1' block, another light blue 'wait until up button is released' block, and a purple 'write counter at 50, 50 with font large black' block. The script ends with a white arrow icon at the bottom of the loop.

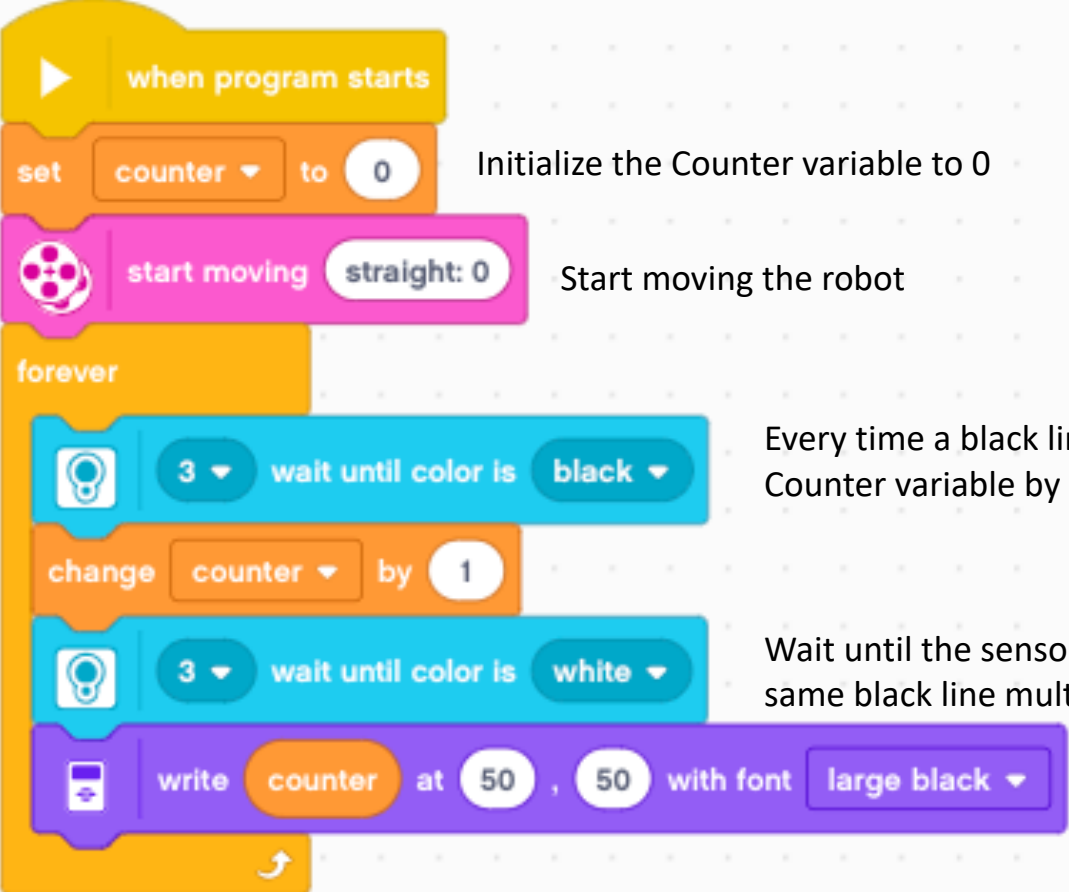
Initialize the Counter variable to 0

Every time the Up button is pressed, increase the Counter variable by one

Wait until the button is released otherwise it will go through the loop multiple times each time you press the button

Write that Counter Variable to the screen to display

Challenge 2 Solution: Count the Lines



The image shows a Scratch script on a grid background. It starts with a yellow 'when program starts' block. This is followed by an orange 'set counter to 0' block. Next is a pink 'start moving straight: 0' block. A large orange 'forever' loop block contains four sub-blocks: a light blue 'wait until color is black' block with a '3' in a dropdown, an orange 'change counter by 1' block, another light blue 'wait until color is white' block with a '3' in a dropdown, and a purple 'write counter at 50, 50 with font large black' block. A white arrow at the bottom of the loop indicates it repeats.

Initialize the Counter variable to 0

Start moving the robot

Every time a black line is seen, increase the Counter variable by one

Wait until the sensor sees white, otherwise you will count the same black line multiple times.

Write that Counter Variable to the screen to display

Non-numeric Variables

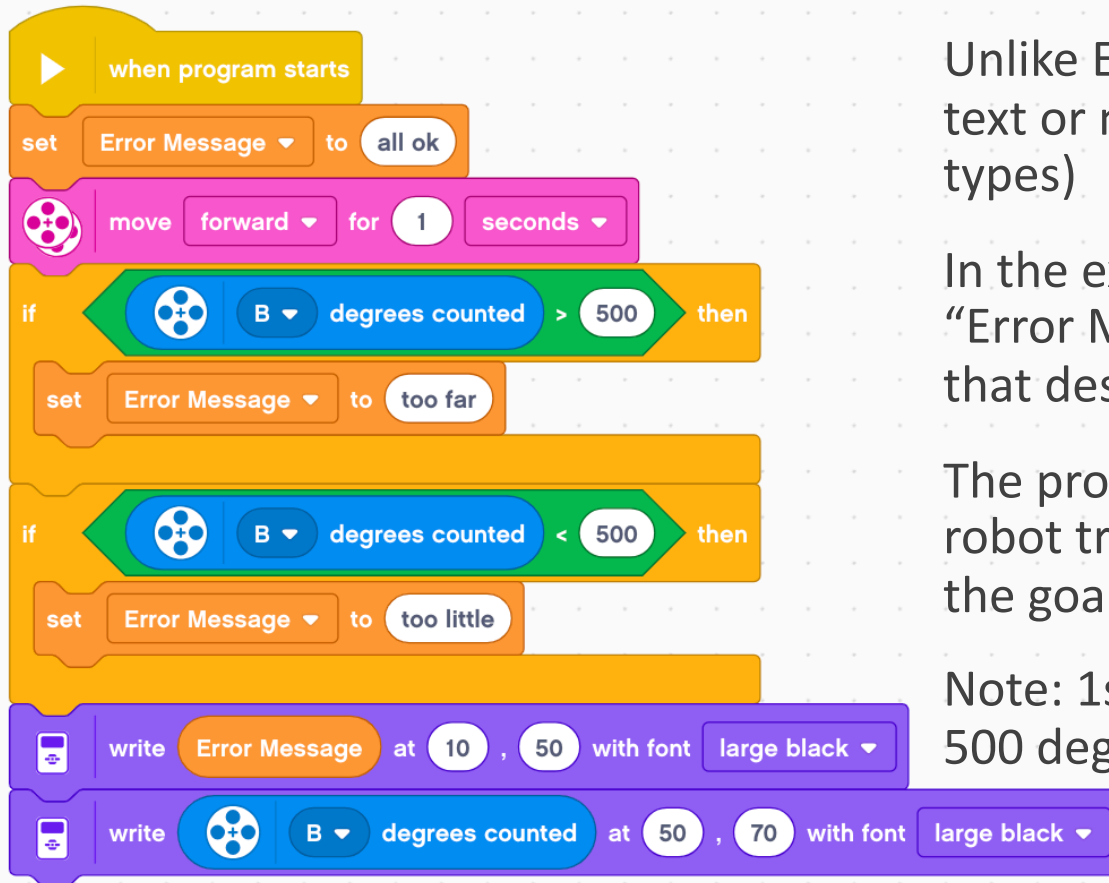
Variables can also store text

Unlike EV3-G, any variable can store text or numbers (there are no variable types)

In the example on the left, we use the “Error Message” variable to store text that describes what went wrong

The program lets the user know if the robot travelled too far or too little if the goal was to move 500 degrees.

Note: 1sec at 50 % speed should move 500 degrees



Discussion and Next Steps

We use variables in the following lessons:

- Advanced: Menu System
- Advanced: Parallel Beam Synchronization

Credits

- This tutorial was created by Sanjay Seshan and Arvind Seshan
- More lessons at www.ev3lessons.com



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).