int

[Data Types]

Description

Integers are your primary data-type for number storage.

On the Arduino Uno (and other ATmega based boards) an int stores a 16-bit (2-byte) value. This yields a range of -32,768 to 32,767 (minimum value of -2^15 and a maximum value of (2^15) - 1). On the Arduino Due and SAMD based boards (like MKR1000 and Zero), an int stores a 32-bit (4-byte) value. This yields a range of -2,147,483,648 to 2,147,483,647 (minimum value of -2^31 and a maximum value of (2^31) - 1).

int’s store negative numbers with a technique called ([2’s complement math](http://en.wikipedia.org/wiki/2%27s_complement)). The highest bit, sometimes referred to as the "sign" bit, flags the number as a negative number. The rest of the bits are inverted and 1 is added.

The Arduino takes care of dealing with negative numbers for you, so that arithmetic operations work transparently in the expected manner. There can be an unexpected complication in dealing with the [bitshift right operator](https://www.arduino.cc/reference/en/language/structure/bitwise-operators/bitshiftright) (>>) however.

Syntax

int var = val;

Parameters

var: variable name.  
val: the value you assign to that variable.

Example Code

This code creates an integer called 'countUp', which is initially set as the number 0 (zero). The variable goes up by 1 (one) each loop, being displayed on the serial monitor.

int countUp = 0; //creates a variable integer called 'countUp'

void setup() {

Serial.begin(9600); // use the serial port to print the number

}

void loop() {

countUp++; //Adds 1 to the countUp int on every loop

Serial.println(countUp); // prints out the current state of countUp

delay(1000);

}

Notes and Warnings

When signed variables are made to exceed their maximum or minimum capacity they *overflow*. The result of an overflow is unpredictable so this should be avoided. A typical symptom of an overflow is the variable "rolling over" from its maximum capacity to its minimum or vice versa, but this is not always the case. If you want this behavior, use [unsigned int](https://www.arduino.cc/reference/en/language/variables/data-types/unsignedint).