Lenguaje de Arduino

La programación de Arduino está basada en el Lenguaje C/C++ y se puede dividir en tres partes principales: Estructura, Datos (Variables y constantes) y Funciones.

Structure

* [setup](https://www.arduino.cc/en/Reference/Setup)()
* [loop](https://www.arduino.cc/en/Reference/Loop)()

Control Structures

* [if](https://www.arduino.cc/en/Reference/If)
* [if...else](https://www.arduino.cc/en/Reference/Else)
* [for](https://www.arduino.cc/en/Reference/For)
* [switch case](https://www.arduino.cc/en/Reference/SwitchCase)
* [while](https://www.arduino.cc/en/Reference/While)
* [do... while](https://www.arduino.cc/en/Reference/DoWhile)
* [break](https://www.arduino.cc/en/Reference/Break)
* [continue](https://www.arduino.cc/en/Reference/Continue)
* [return](https://www.arduino.cc/en/Reference/Return)
* [goto](https://www.arduino.cc/en/Reference/Goto)

Further Syntax

* [;](https://www.arduino.cc/en/Reference/SemiColon) (semicolon)
* [{}](https://www.arduino.cc/en/Reference/Braces) (curly braces)
* [//](https://www.arduino.cc/en/Reference/Comments) (single line comment)
* [/\* \*/](https://www.arduino.cc/en/Reference/Comments) (multi-line comment)
* [#define](https://www.arduino.cc/en/Reference/Define)
* [#include](https://www.arduino.cc/en/Reference/Include)

Arithmetic Operators

* [=](https://www.arduino.cc/en/Reference/Assignment) (assignment operator)
* [+](https://www.arduino.cc/en/Reference/Arithmetic) (addition)
* [-](https://www.arduino.cc/en/Reference/Arithmetic) (subtraction)
* [\*](https://www.arduino.cc/en/Reference/Arithmetic) (multiplication)
* [/](https://www.arduino.cc/en/Reference/Arithmetic) (division)
* [%](https://www.arduino.cc/en/Reference/Modulo) (modulo)

Comparison Operators

* [==](https://www.arduino.cc/en/Reference/If) (equal to)
* [!=](https://www.arduino.cc/en/Reference/If) (not equal to)
* [<](https://www.arduino.cc/en/Reference/If) (less than)
* [>](https://www.arduino.cc/en/Reference/If) (greater than)
* [<=](https://www.arduino.cc/en/Reference/If) (less than or equal to)
* [>=](https://www.arduino.cc/en/Reference/If) (greater than or equal to)

Boolean Operators

* [&&](https://www.arduino.cc/en/Reference/Boolean) (and)
* [||](https://www.arduino.cc/en/Reference/Boolean) (or)
* [!](https://www.arduino.cc/en/Reference/Boolean) (not)

Pointer Access Operators

* [\* dereference operator](https://www.arduino.cc/en/Reference/Pointer)
* [& reference operator](https://www.arduino.cc/en/Reference/Pointer)

Bitwise Operators

* [&](https://www.arduino.cc/en/Reference/BitwiseAnd) (bitwise and)
* [|](https://www.arduino.cc/en/Reference/BitwiseAnd) (bitwise or)
* [^](https://www.arduino.cc/en/Reference/BitwiseAnd) (bitwise xor)
* [~](https://www.arduino.cc/en/Reference/BitwiseXorNot) (bitwise not)
* [<<](https://www.arduino.cc/en/Reference/Bitshift) (bitshift left)
* [>>](https://www.arduino.cc/en/Reference/Bitshift) (bitshift right)

Compound Operators

* [++](https://www.arduino.cc/en/Reference/Increment) (increment)
* [--](https://www.arduino.cc/en/Reference/Increment) (decrement)
* [+=](https://www.arduino.cc/en/Reference/IncrementCompound) (compound addition)
* [-=](https://www.arduino.cc/en/Reference/IncrementCompound) (compound subtraction)
* [\*=](https://www.arduino.cc/en/Reference/IncrementCompound) (compound multiplication)
* [/=](https://www.arduino.cc/en/Reference/IncrementCompound) (compound division)
* [%=](https://www.arduino.cc/en/Reference/IncrementCompound) (compound modulo)
* [&=](https://www.arduino.cc/en/Reference/BitwiseCompoundAnd) (compound bitwise and)
* [|=](https://www.arduino.cc/en/Reference/BitwiseCompoundOr) (compound bitwise or)

Variables

Constants

* [HIGH](https://www.arduino.cc/en/Reference/Constants) | [LOW](https://www.arduino.cc/en/Reference/Constants)
* [INPUT](https://www.arduino.cc/en/Reference/Constants) | [OUTPUT](https://www.arduino.cc/en/Reference/Constants) | [INPUT\_PULLUP](https://www.arduino.cc/en/Reference/Constants)
* [LED\_BUILTIN](https://www.arduino.cc/en/Reference/Constants)
* [true](https://www.arduino.cc/en/Reference/Constants) | [false](https://www.arduino.cc/en/Reference/Constants)
* [integer constants](https://www.arduino.cc/en/Reference/IntegerConstants)
* [floating point constants](https://www.arduino.cc/en/Reference/Fpconstants)

Data Types

* [void](https://www.arduino.cc/en/Reference/Void)
* [boolean](https://www.arduino.cc/en/Reference/BooleanVariables)
* [char](https://www.arduino.cc/en/Reference/Char)
* [unsigned char](https://www.arduino.cc/en/Reference/UnsignedChar)
* [byte](https://www.arduino.cc/en/Reference/Byte)
* [int](https://www.arduino.cc/en/Reference/Int)
* [unsigned int](https://www.arduino.cc/en/Reference/UnsignedInt)
* [word](https://www.arduino.cc/en/Reference/Word)
* [long](https://www.arduino.cc/en/Reference/Long)
* [unsigned long](https://www.arduino.cc/en/Reference/UnsignedLong)
* [short](https://www.arduino.cc/en/Reference/Short)
* [float](https://www.arduino.cc/en/Reference/Float)
* [double](https://www.arduino.cc/en/Reference/Double)
* [string](https://www.arduino.cc/en/Reference/String) - char array
* [String](https://www.arduino.cc/en/Reference/StringObject) - object
* [array](https://www.arduino.cc/en/Reference/Array)

Conversion

* [char()](https://www.arduino.cc/en/Reference/CharCast)
* [byte()](https://www.arduino.cc/en/Reference/ByteCast)
* [int()](https://www.arduino.cc/en/Reference/IntCast)
* [word()](https://www.arduino.cc/en/Reference/WordCast)
* [long()](https://www.arduino.cc/en/Reference/LongCast)
* [float()](https://www.arduino.cc/en/Reference/FloatCast)

Variable Scope & Qualifiers

* [variable scope](https://www.arduino.cc/en/Reference/Scope)
* [static](https://www.arduino.cc/en/Reference/Static)
* [volatile](https://www.arduino.cc/en/Reference/Volatile)
* [const](https://www.arduino.cc/en/Reference/Const)

Utilities

* [sizeof](https://www.arduino.cc/en/Reference/Sizeof)()
* [PROGMEM](https://www.arduino.cc/en/Reference/PROGMEM)

[Functions](http://arduino.cc/en/Reference/FunctionDeclaration)

Digital I/O

* [pinMode](https://www.arduino.cc/en/Reference/PinMode)()
* [digitalWrite](https://www.arduino.cc/en/Reference/DigitalWrite)()
* [digitalRead](https://www.arduino.cc/en/Reference/DigitalRead)()

Analog I/O

* [analogReference](https://www.arduino.cc/en/Reference/AnalogReference)()
* [analogRead](https://www.arduino.cc/en/Reference/AnalogRead)()
* [analogWrite](https://www.arduino.cc/en/Reference/AnalogWrite)() - *PWM*

Due & Zero only

* [analogReadResolution](https://www.arduino.cc/en/Reference/AnalogReadResolution)()
* [analogWriteResolution](https://www.arduino.cc/en/Reference/AnalogWriteResolution)()

Advanced I/O

* [tone](https://www.arduino.cc/en/Reference/Tone)()
* [noTone](https://www.arduino.cc/en/Reference/NoTone)()
* [shiftOut](https://www.arduino.cc/en/Reference/ShiftOut)()
* [shiftIn](https://www.arduino.cc/en/Reference/ShiftIn)()
* [pulseIn](https://www.arduino.cc/en/Reference/PulseIn)()

Time

* [millis](https://www.arduino.cc/en/Reference/Millis)()
* [micros](https://www.arduino.cc/en/Reference/Micros)()
* [delay](https://www.arduino.cc/en/Reference/Delay)()
* [delayMicroseconds](https://www.arduino.cc/en/Reference/DelayMicroseconds)()

Math

* [min](https://www.arduino.cc/en/Reference/Min)()
* [max](https://www.arduino.cc/en/Reference/Max)()
* [abs](https://www.arduino.cc/en/Reference/Abs)()
* [constrain](https://www.arduino.cc/en/Reference/Constrain)()
* [map](https://www.arduino.cc/en/Reference/Map)()
* [pow](https://www.arduino.cc/en/Reference/Pow)()
* [sqrt](https://www.arduino.cc/en/Reference/Sqrt)()

Trigonometry

* [sin](https://www.arduino.cc/en/Reference/Sin)()
* [cos](https://www.arduino.cc/en/Reference/Cos)()
* [tan](https://www.arduino.cc/en/Reference/Tan)()

Characters

* [isAlphaNumeric](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isAlpha](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isAscii](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isWhitespace](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isControl](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isDigit](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isGraph](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isLowerCase](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isPrintable](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isPunct](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isSpace](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isUpperCase](https://www.arduino.cc/en/Reference/CharacterAnalysis)()
* [isHexadecimalDigit](https://www.arduino.cc/en/Reference/CharacterAnalysis)()

Random Numbers

* [randomSeed](https://www.arduino.cc/en/Reference/RandomSeed)()
* [random](https://www.arduino.cc/en/Reference/Random)()

Bits and Bytes

* [lowByte](https://www.arduino.cc/en/Reference/LowByte)()
* [highByte](https://www.arduino.cc/en/Reference/HighByte)()
* [bitRead](https://www.arduino.cc/en/Reference/BitRead)()
* [bitWrite](https://www.arduino.cc/en/Reference/BitWrite)()
* [bitSet](https://www.arduino.cc/en/Reference/BitSet)()
* [bitClear](https://www.arduino.cc/en/Reference/BitClear)()
* [bit](https://www.arduino.cc/en/Reference/Bit)()

External Interrupts

* [attachInterrupt](https://www.arduino.cc/en/Reference/AttachInterrupt)()
* [detachInterrupt](https://www.arduino.cc/en/Reference/DetachInterrupt)()

Interrupts

* [interrupts](https://www.arduino.cc/en/Reference/Interrupts)()
* [noInterrupts](https://www.arduino.cc/en/Reference/NoInterrupts)()

Communication

* [Serial](https://www.arduino.cc/en/Reference/Serial)
* [Stream](https://www.arduino.cc/en/Reference/Stream)

USB (32u4 based boards and Due/Zero only)

* [Keyboard](https://www.arduino.cc/en/Reference/MouseKeyboard)
* [Mouse](https://www.arduino.cc/en/Reference/MouseKeyboard)

**INSTRUCCIONES IMPORTANTES**

## **pinMode()**

#### Description

Configures the specified pin to behave either as an input or an output. See the description of [digital pins](https://www.arduino.cc/en/Tutorial/DigitalPins) for details on the functionality of the pins.

As of Arduino 1.0.1, it is possible to enable the internal pullup resistors with the mode INPUT\_PULLUP. Additionally, the INPUT mode explicitly disables the internal pullups.

#### Syntax

pinMode(pin, mode)

#### Parameters

pin: the number of the pin whose mode you wish to set

mode: [INPUT](https://www.arduino.cc/en/Reference/Constants), [OUTPUT](https://www.arduino.cc/en/Reference/Constants), or [INPUT\_PULLUP](https://www.arduino.cc/en/Reference/Constants). (see the [digital pins](https://www.arduino.cc/en/Tutorial/DigitalPins) page for a more complete description of the functionality.)

#### Returns

None

## **digitalRead()**

#### Description

Reads the value from a specified digital pin, either [HIGH](https://www.arduino.cc/en/Reference/Constants) or [LOW](https://www.arduino.cc/en/Reference/Constants).

#### Syntax

digitalRead(pin)

#### Parameters

pin: the number of the digital pin you want to read (int)

#### Returns

[HIGH](https://www.arduino.cc/en/Reference/Constants) or [LOW](https://www.arduino.cc/en/Reference/Constants)

## **digitalWrite()**

#### Description

Write a [HIGH](https://www.arduino.cc/en/Reference/Constants) or a [LOW](https://www.arduino.cc/en/Reference/Constants) value to a digital pin.

If the pin has been configured as an OUTPUT with [pinMode](https://www.arduino.cc/en/Reference/PinMode)(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

If the pin is configured as an INPUT, digitalWrite() will enable (HIGH) or disable (LOW) the internal pullup on the input pin. It is recommended to set the [pinMode](https://www.arduino.cc/en/Reference/PinMode)() to [INPUT\_PULLUP](https://www.arduino.cc/en/Reference/Constants) to enable the internal pull-up resistor. See the [digital pins tutorial](https://www.arduino.cc/en/Tutorial/DigitalPins) for more information.

NOTE: If you do not set the pinMode() to OUTPUT, and connect an LED to a pin, when calling digitalWrite(HIGH), the LED may appear dim. Without explicitly setting pinMode(), digitalWrite() will have enabled the internal pull-up resistor, which acts like a large current-limiting resistor.

#### Syntax

digitalWrite(pin, value)

#### Parameters

pin: the pin number

value: [HIGH](https://www.arduino.cc/en/Reference/Constants) or [LOW](https://www.arduino.cc/en/Reference/Constants)

#### Returns

none

EJEMPLO:

int ledPin = 13;                 *// LED connected to digital pin 13*  
  
void **setup**()  
{  
  pinMode(ledPin, OUTPUT);      *// sets the digital pin as output*  
}  
  
void **loop**()  
{  
  digitalWrite(ledPin, HIGH);   *// sets the LED on*  
  delay(1000);                  *// waits for a second*  
  digitalWrite(ledPin, LOW);    *// sets the LED off*  
  delay(1000);                  *// waits for a second*  
}