Comunidad de computrónica

Instituto Tecnológico de Costa Rica



**Manual de iniciación a Arduino**

Daniel Amador Salas

# 

[**Historia de arduino**](#_4gk9pc2rec18) **3**

[**¿Qué es Arduino?**](#_4y1xay5ua89l) **3**

[**Conceptos Básicos**](#_t8q80ruvc4qc) **3**

[Hardware](#_h3vim9waeykr) 3

[¿Qué es un microcontrolador?](#_8a1k0llrd7bg) 3

[Entradas/Salidas](#_5ok0vjh7jt83) 3

[Digital/Analógico](#_pyzi5d46mpux) 4

[Software](#_f11sydyfyhd5) 4

[Estructura](#_8awiwxy733i2) 4

[Interface de desarrollo integrado (IDE)](#_uiwj2ndxce5c) 5

[Práctica](#_t39inv4ccqas) 6

[Ejercicio #1](#_la4w58ef0mnu) 6

[Ejercicio #2](#_9xb1usgcb9qf) 8

[Ejercicio #3](#_vdpdrrcuw25s) 9

[Ejercicio #4](#_r8t9xdsq196m) 11

# 

# Historia de arduino

Arduino fue inventado en 2015 por Massimo Banzi estudiante del instituto IVRAE. En principio se creó por la necesidad de aprendizaje de los estudiantes de computación y electrónica del mismo instituto, ya que adquirir una placa de microcontroladores era bastante caro. Su nombre proviene del bar “Bar di Re Arduino” donde Massimo pasaba algunas horas.

# ¿Qué es Arduino?

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Está pensado para artistas, diseñadores, como hobby y para cualquiera interesado en crear objetos o entornos interactivos.

Arduino al ser una comunidad open-source ofrece varias ventajas como:

* Multitud de shields y versiones de placas.
* Una amplia comunidad de usuarios.
* Variedad de precios accesibles.
* La posibilidad de desarrollar nuestros propios prototipos.

# Conceptos Básicos

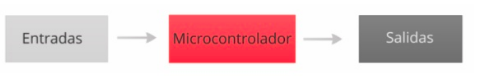
## Hardware

### ¿Qué es un microcontrolador?

Es un circuito integrado programable capaz de realizar operaciones matemáticas a gran velocidad.

### Entradas/Salidas

Dentro de la placa arduino el microcontrolador se encarga de procesar la información de las entradas produciendo cambios en las salidas.



### Digital/Analógico

* **Digital:** las señales digitales toman valores entre 0 y 1 / apagado o encendido / 0v o 5v.
* **Analógico:** las señales analógicas toman valores infinitos entre su valor mínimo y su valor máximo.

#### 

## Software

### Estructura

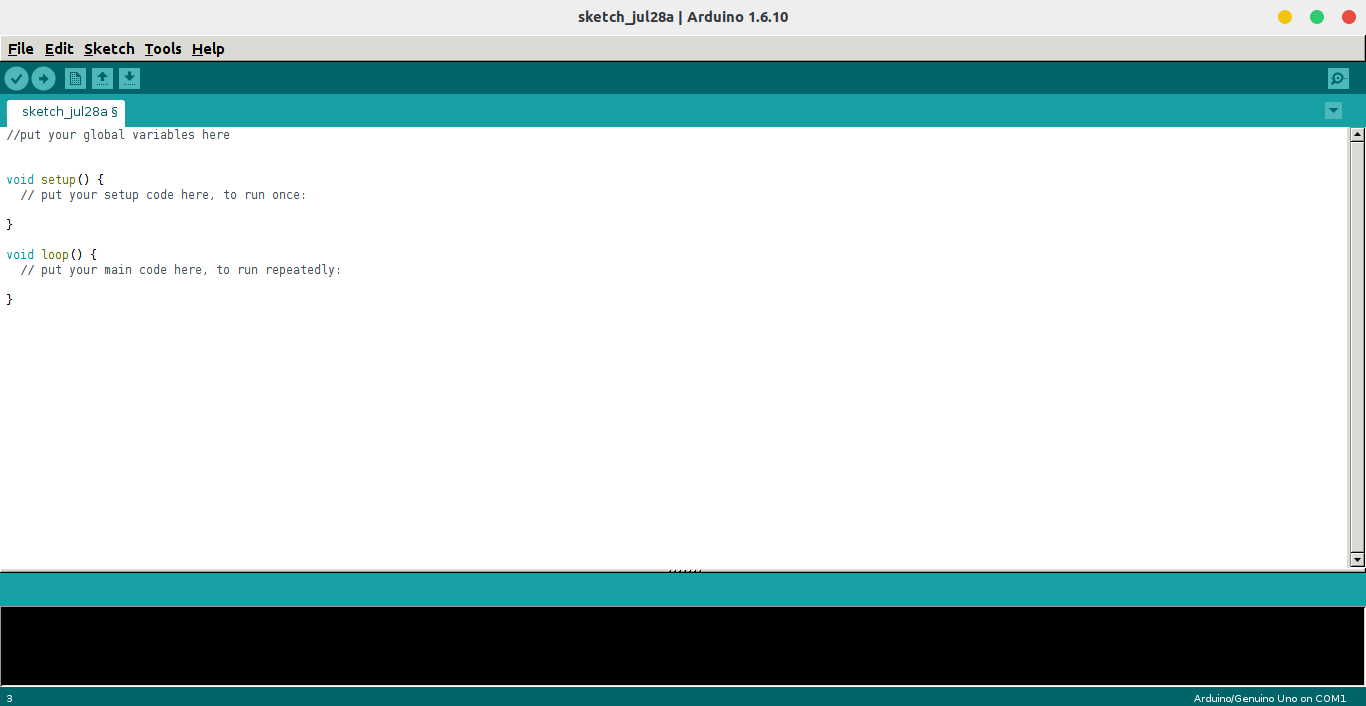
El entorno de desarrollo integrado(IDE) de Arduino está compuesto por tres partes principales.

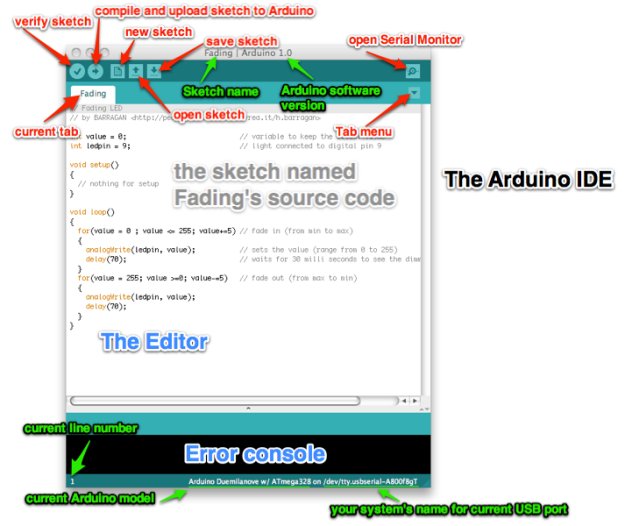
En la primera parte del IDE fuera de los dos bloques principales de código se encuentran las variables globales.

Luego tenemos el setup en este bloque se realizan las declaraciones de roles de cada uno de los pines del Arduino, es decir si es de entrada o salida.

El bloque loop es donde se realiza todo el trabajo de programación del arduino y aquí es donde se configuran la recepción de datos, el comportamiento de los sensores y demás componentes que utilicemos.

#### Interface de desarrollo integrado (IDE)



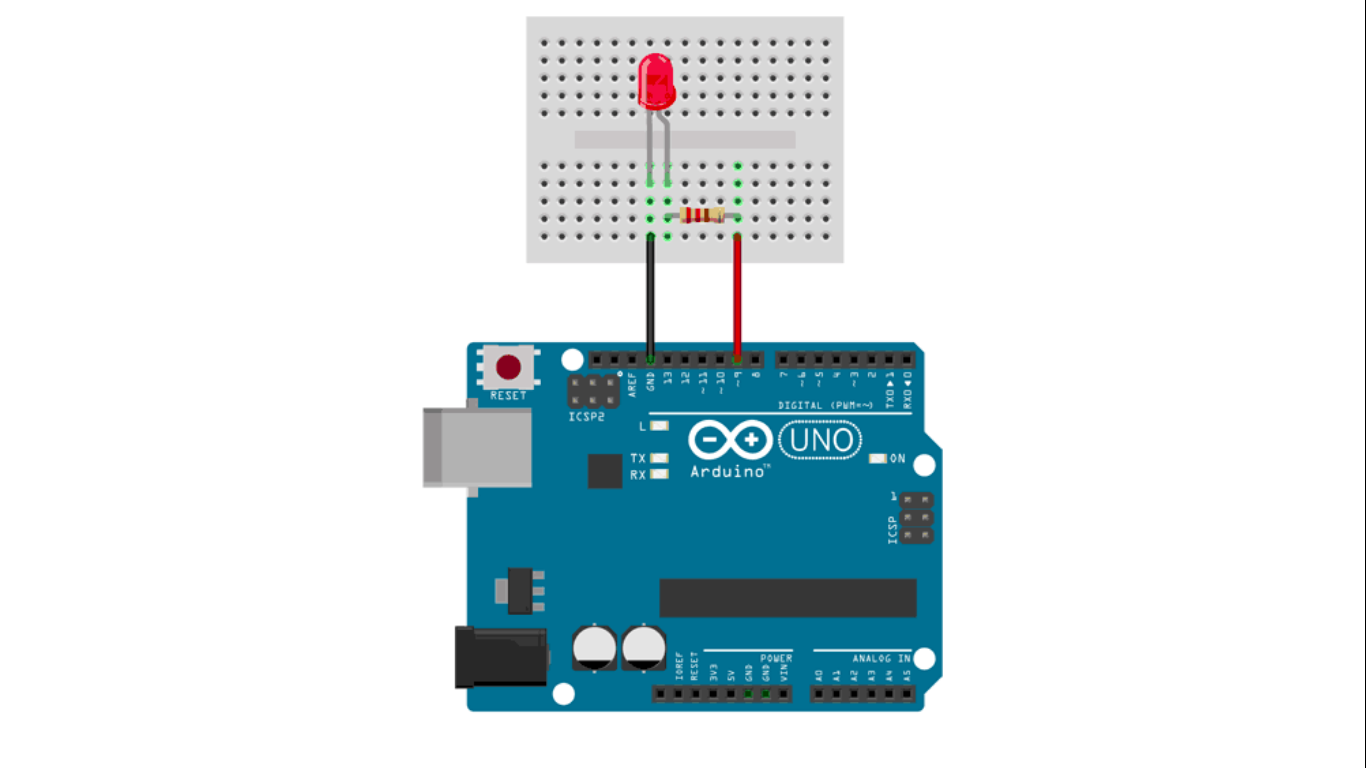
En la siguiente imagen se muestran las partes del arduino. 

## Práctica

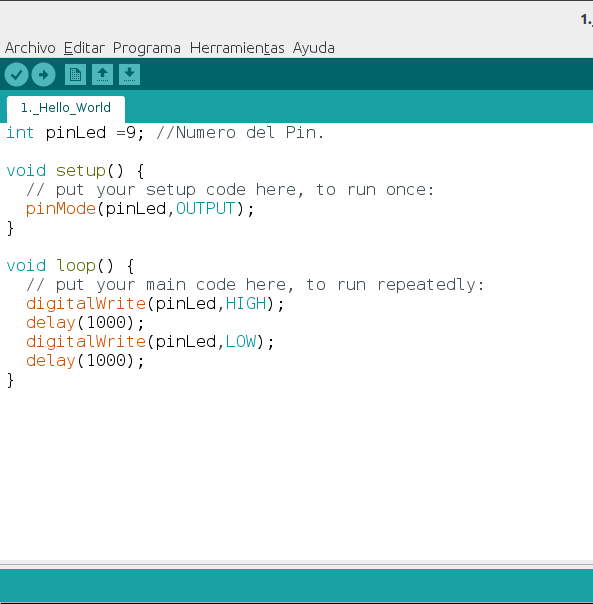
### Ejercicio #1

Este primer ejercicio es la típica introducción a la manipulación de componentes en arduino. En el vamos a conectar y configurar un led para que se encienda y se apague. Los materiales y instrucciones de ensamblaje son los siguientes.





El código de configuración para este ejercicio es el siguiente:



#### 

#### 

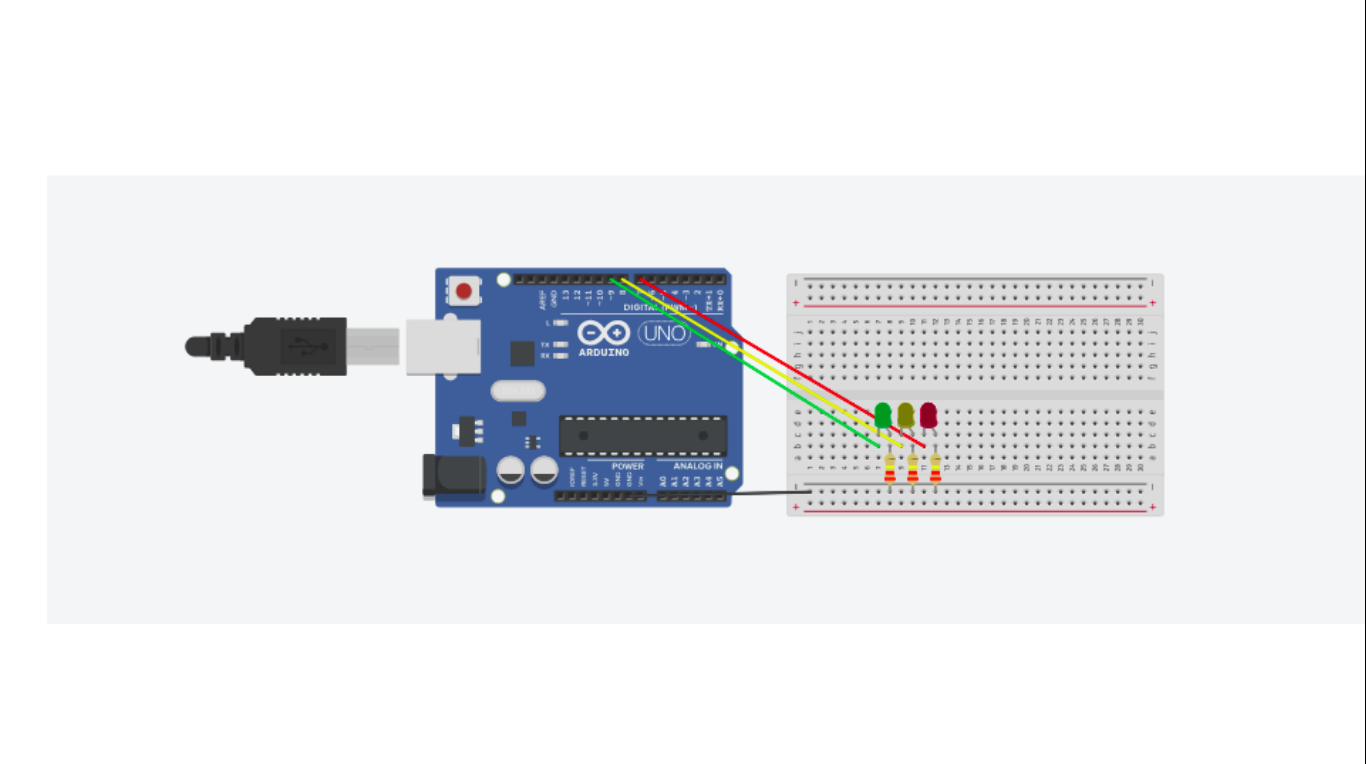
#### 

#### 

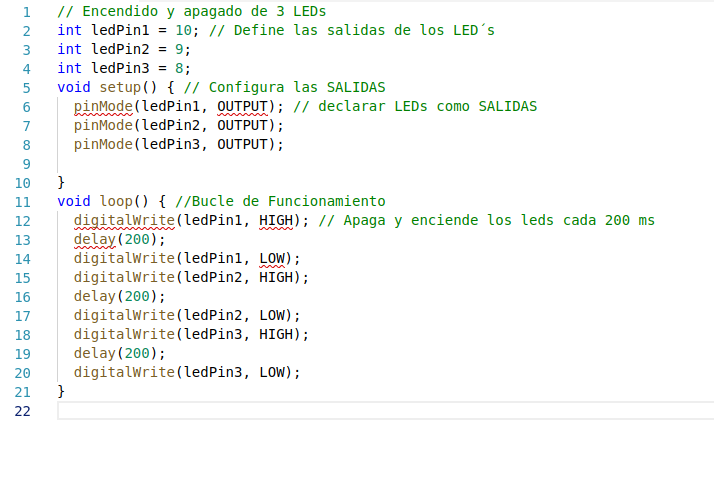
### Ejercicio #2

En este segundo ejercicio configuraremos una secuencia de leds para que actué como un semáforo. Para esto vamos a seguir las siguientes instrucciones.



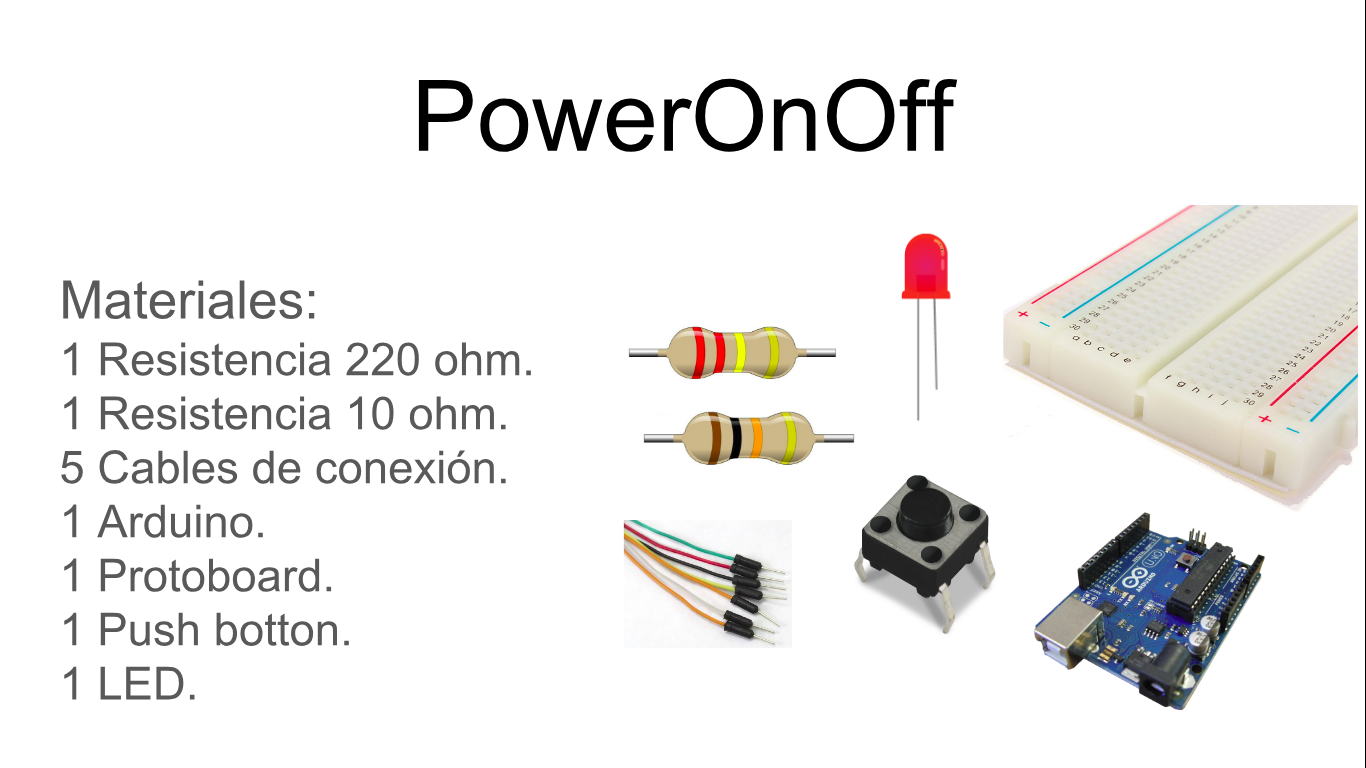


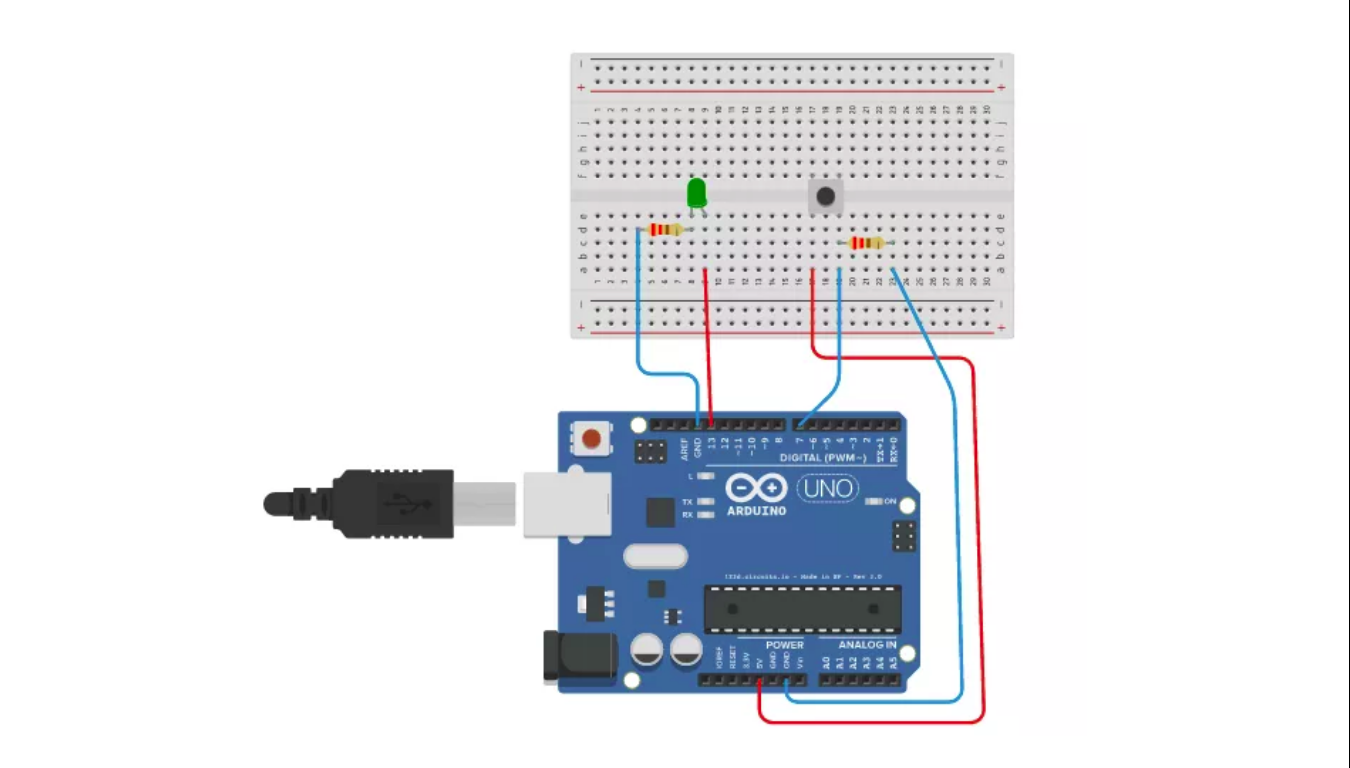
El código de configuración del este ejercicio es el siguiente.



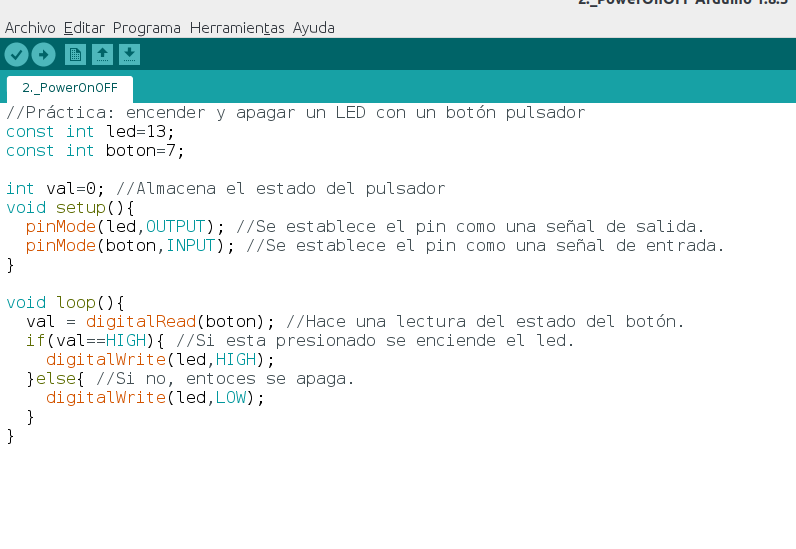
### Ejercicio #3

En este tercer ejercicio vamos a configurar un botón para que encienda y apague un led simulando una lámpara. Las instrucciones de ensamblaje y configuración son las siguiente.

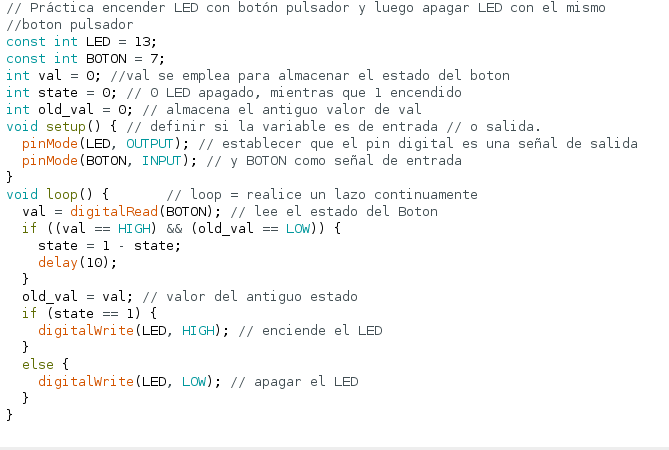




Las instrucciones de configuración son las siguientes. Con esta configuración si mantenemos pulsado el botón el led permanecerá encendido y soltamos se apagará inmediatamente.



Si queremos configurarlo para que funcione simulando una lámpara podemos configurarlo de la siguiente manera.

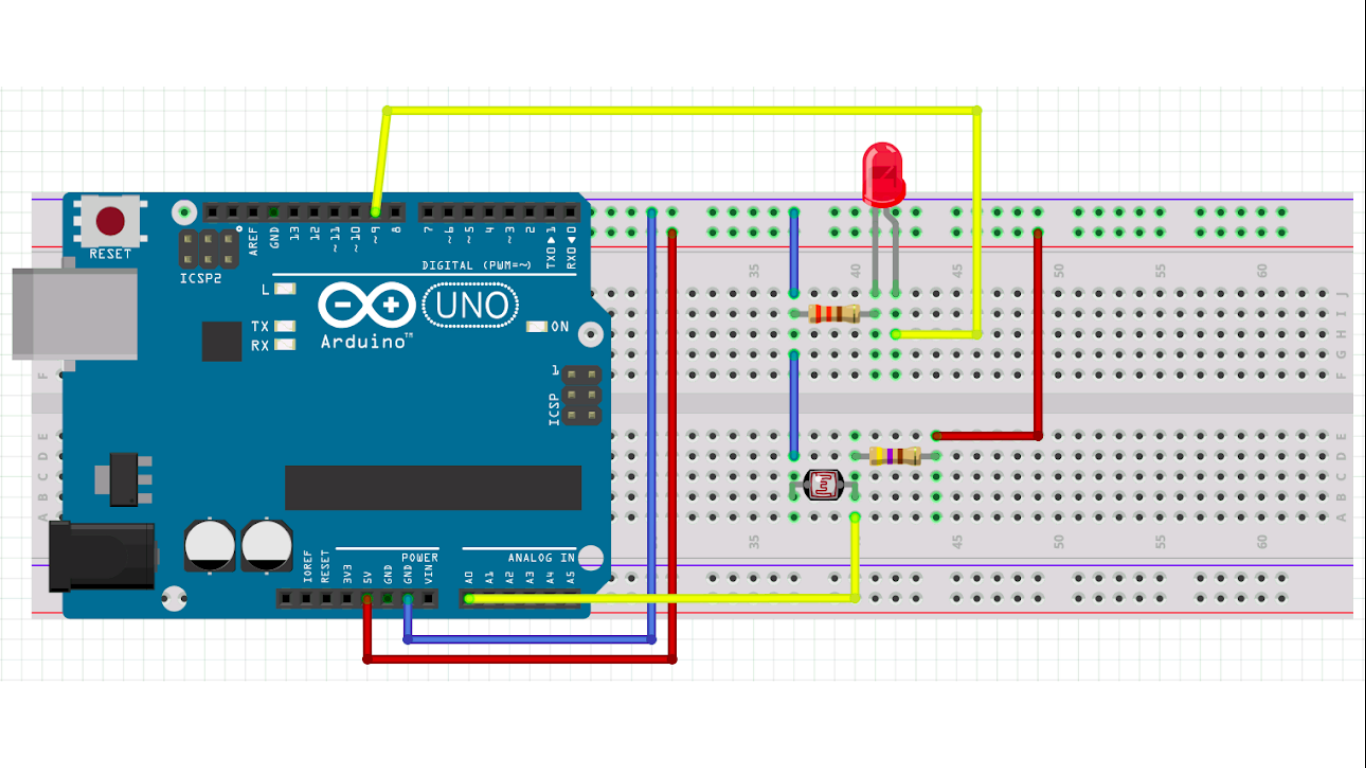


#### 

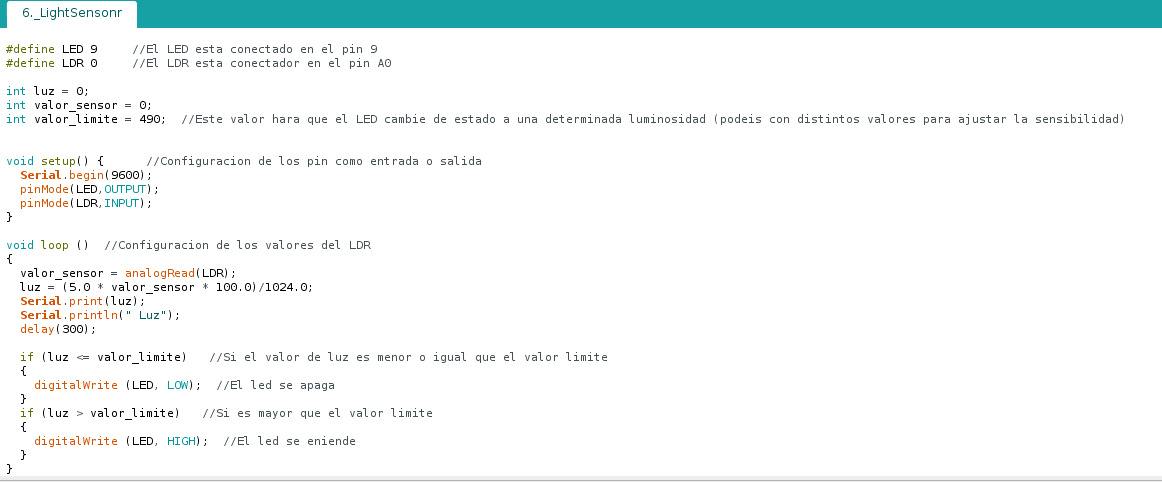
### Ejercicio #4

En este cuarto ejercicio vamos a configurar un sensor de luminiscencia para encender un led. El objetivo es configurarlo para cuando haya poca luz se encienda el led y cuando este iluminado se apague. Las instrucciones de ensamblaje y configuración son las siguientes.





El código de configuración de este ejercicio es el siguiente



#### 