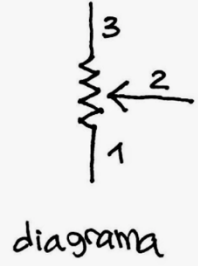
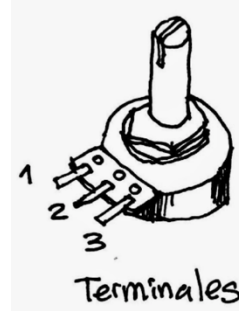
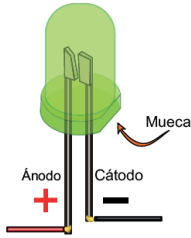


## PRACTICA 2

### MATERIALES:



- 1 Placa de desarrollo NodeMCU y su cable USB
- 1 Potenciómetro
- 2 LED
- 2 Resistencia de 56 ohmios (opcionales)\*

### Condicionales, if y bloques

La estructura if nos permite ejecutar fragmentos de códigos o bloques dependiendo de si se cumple o no una condición. La estructura básica de un if es la siguiente:

#### Fragmento de código

```
if (condicion) {  
    instrucciones a ejecutar.  
}
```

Las instrucciones que se encuentran dentro del if (las que están dentro de las llaves “{ }”), solo se van a ejecutar si la condición se cumple, es decir si la condición que está dentro de los paréntesis tiene resultado verdadero.

Las condiciones son preguntas por valores de verdad. Por ejemplo, podemos preguntar si un número es mayor que otro, si es menor, si es igual o si es distinto, etc. Incluso podemos combinar condiciones usando operadores lógicos (AND y OR). Las siguientes son ejemplos de condiciones:

Expresión	Qué devuelve	Qué significa
8>0	verdadero	8 es mayor a 0
10<11	verdadero	10 es menor a 11
8>10	falso	8 es mayor a 10
10>=0	verdadero	10 es mayor o igual a 0
5==0	falso	5 es igual a 0
5!=0	verdadero	5 es distinto de 0
digitalRead(8) == LOW	Depende del valor en el pin 8	Si el resultado de la función read es LOW devolverá verdadero, sino falso

Para combinar condiciones se puede utilizar el siguiente ejemplo. Se utiliza la lógica AND para realizar una instrucción si cumple la condición de tener un número mayor que 0 y menor que 5. También se restringir una instrucción si cumple la lógica OR o incluso utilizando la lógica NOT.

Logica AND:

```
if (x > 0 && x < 5)    // cierto sólo si las dos expresiones
                        // son ciertas
```

Logica OR:

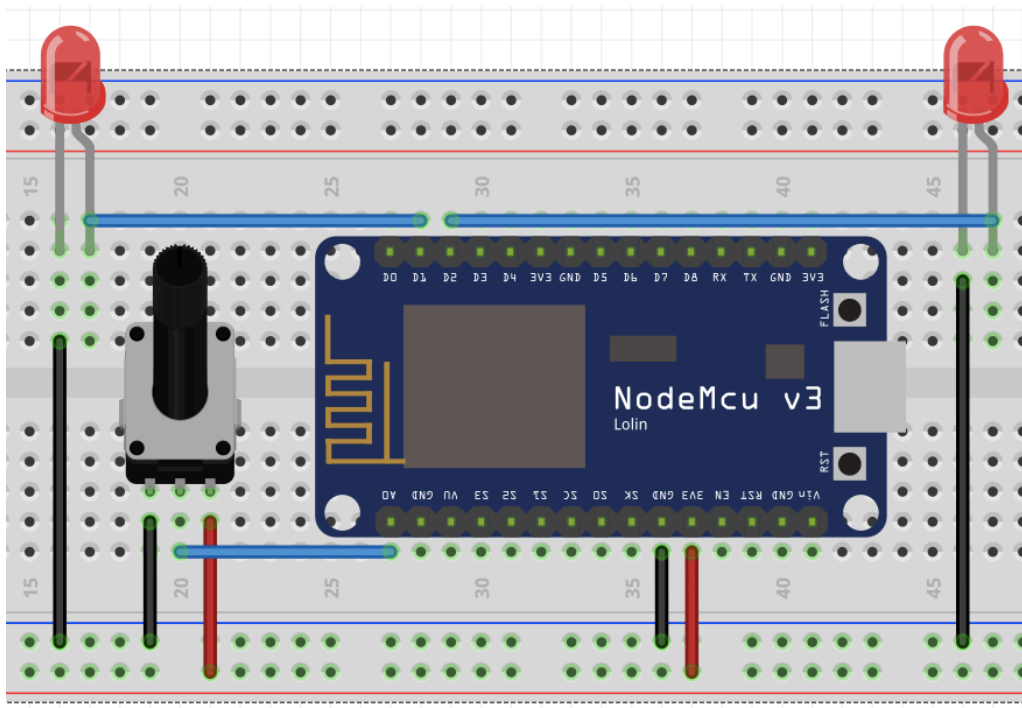
```
if (x > 0 || y > 0)    // cierto si una cualquiera de las
                        // expresiones es cierta
```

Logica NOT:

```
if (!x > 0)            // cierto solo si la expresión es
                        // falsa
```

Al conectar los leds tendrás que tener un especial cuidado ya que las patas de los mismos deben estar correctamente conectadas. Recuerda que las salidas son de 3V y a diferencia del Arduino que tiene 5V en las salidas y por lo tanto puedes colocar resistencias de solo 56 Ohmios para limitar la corriente.

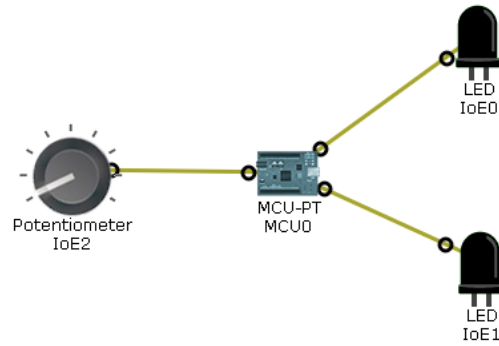
Recuerda que el ánodo o positivo generalmente tiene la pata más larga y va conectado a la señal de salida del NODEMCU en pines GPIO programados como salida.



**Ejercicio 1:** Crear un código de tal forma que al rotar el potenciómetro los primeros grados encienda un led y al continuar rotando el potenciómetro encienda el otro led. Los led's me indicarán el nivel de rotación del potenciómetro.

El NodeMCU puede tener un voltaje máximo en el convertidor de señales analógicas a digitales (para tener una lectura de 1023) de 3.3V. Esto implica que dentro del código tendremos un valor entre 0 y 1023 que se mapea con el voltaje entre 0 y 3,3V.





Ejercicio 2: En una empresa existe un sensor de luz que la hace encender si esta baja de cierto nivel. También hay un interruptor para encender la luz. El sensor de temperatura será simulado por el potenciómetro y la luz por el led.

A través de un led y el NodeMCU implementar el circuito que encienda la luz en caso de que se cumpla cualquier condición, que disminuya la tensión en el puerto análogo o que esté en alto un puerto GPIO.

### TRABAJO PARA LA CASA:

Realice una simulación en el Packet Tracer de tal forma que el led se encienda para temperaturas elevadas.

The screenshot shows a Packet Tracer workspace with a circuit diagram on the left and a code editor on the right.

**Circuit Diagram:** A Temperature Sensor (IoE4) is connected to an MCU-PT MCU0. A Heating Element (IoE7) is connected to a Potentiometer (IoE8), which is also connected to the MCU0. The MCU0 is connected to an LED (IoE0).

**Code Editor:** The code is titled "Blink (JavaScript) - main.js". It contains the following JavaScript code:

```

1 function setup() {
2   pinMode(0, INPUT);
3   pinMode(1, OUTPUT);
4 }
5
6 function loop() {
7   var val = digitalRead (0);
8
9   if (val > 800) {
10    digitalWrite(1, HIGH);
11  }
12  else {
13    digitalWrite(1, LOW);
14  }
15  Serial.println(val);
16  delay (1000);
17 }
18

```

The console output shows the following messages:

```

566
558
550
542
Blink (JavaScript) stopped.

```