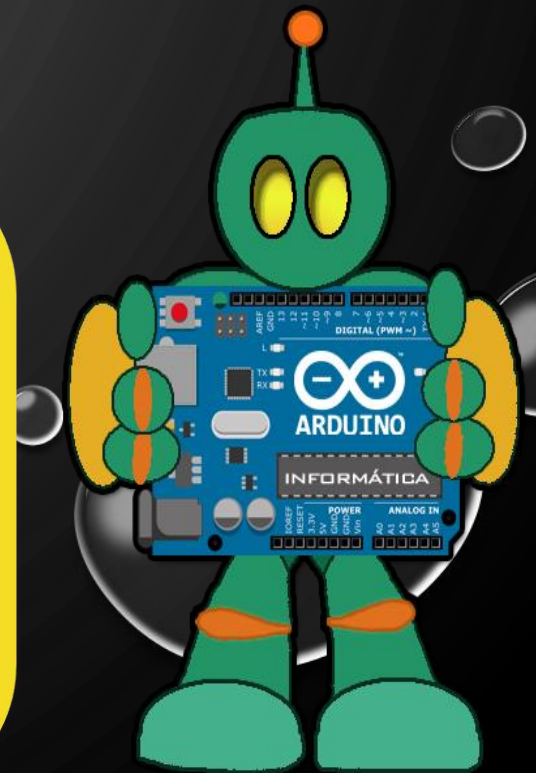
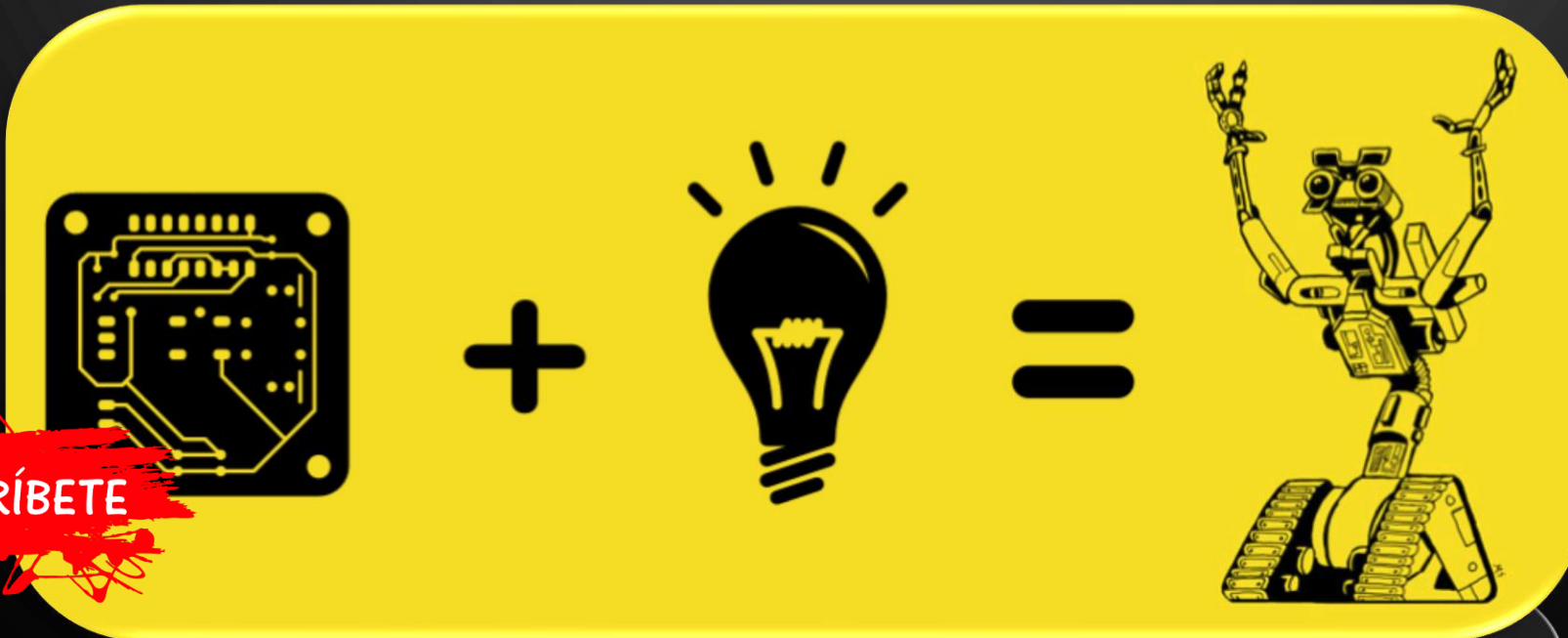
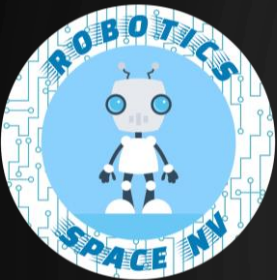


# COMUNIDAD ARDUINO OPEN SOURCE

## CLASE 6 – NODEBOTS



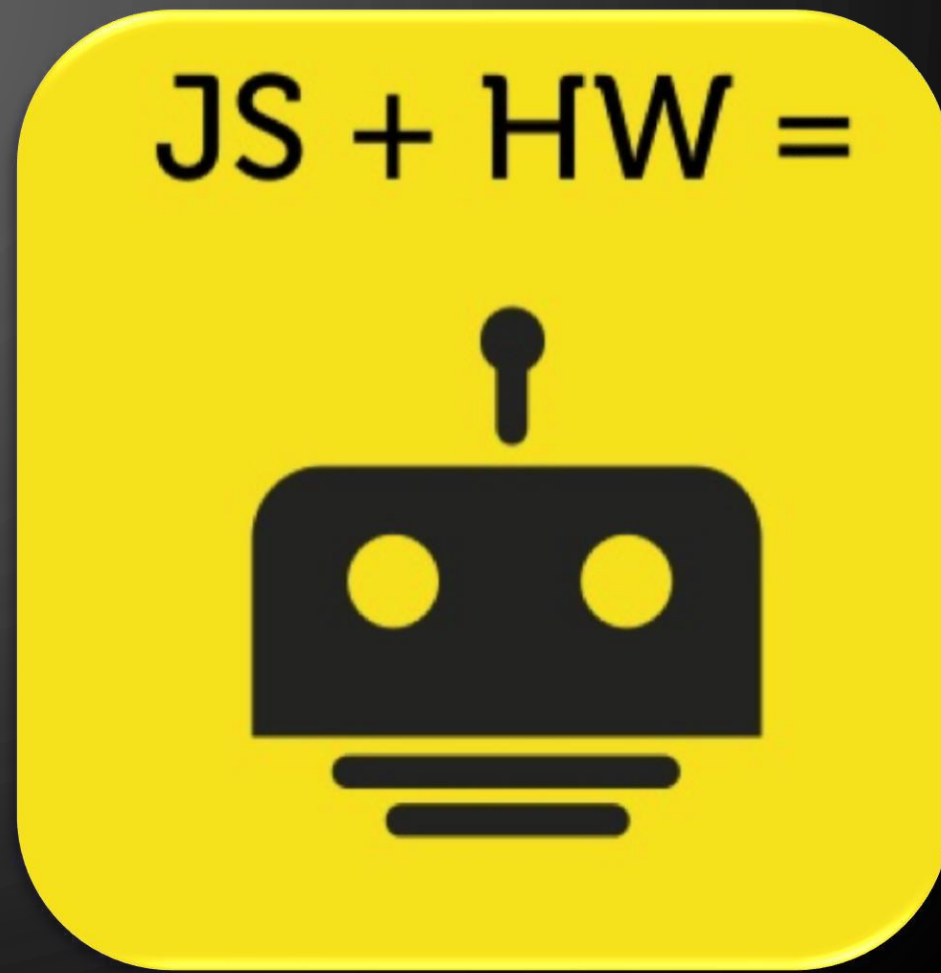
SUSCRÍBETE



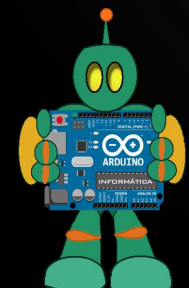
# NODEBOTS

Es una nueva herramienta para la programación de robots.

Tiene por objetivo programar robots a través de JavaScript con soporte a través de Node.JS y J5



TUTOR: NAGIB LUIS VALLEJOS M.





# NODE.JS

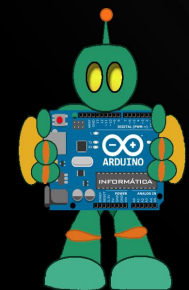
Node.js® es un entorno de ejecución para javascript construido con el **motor de javascript** V8. Está diseñado para crear aplicaciones network escalables y es multiplataforma.

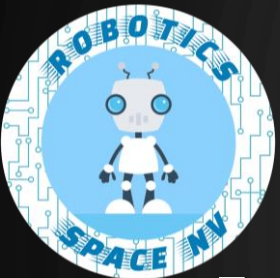
Pueden acceder a el desde:

<https://nodejs.org/es/>



TUTOR: NAGIB LUIS VALLEJOS M.





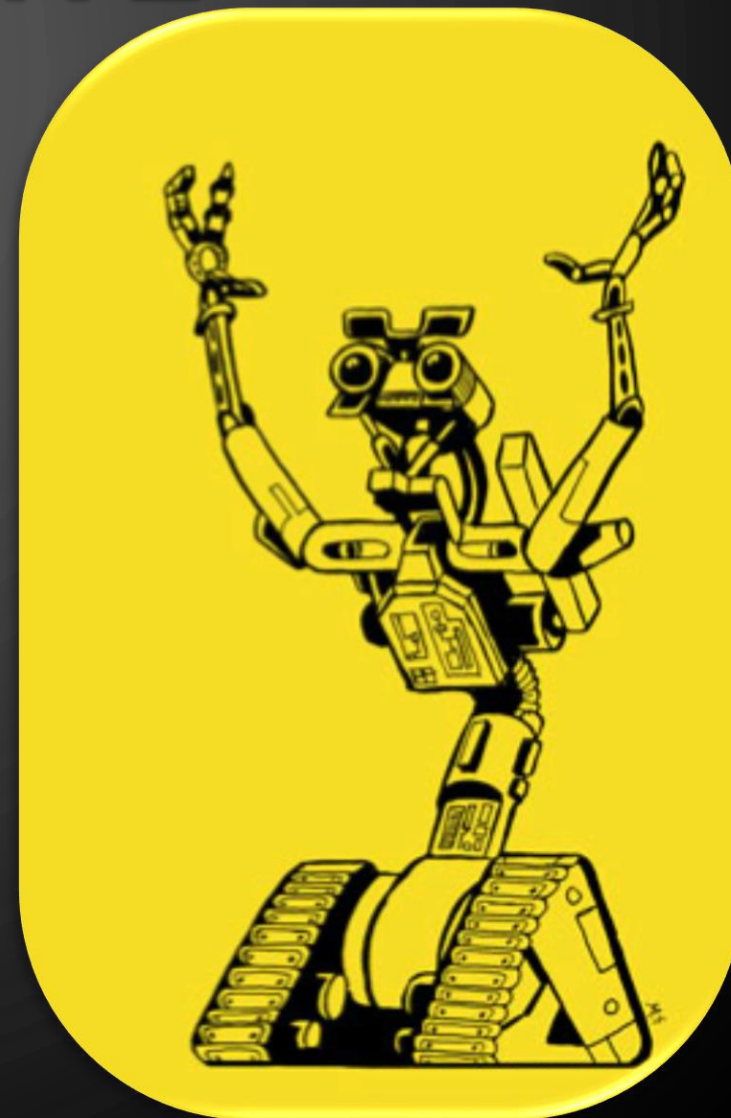
# JOHNNY-FIVE

Es un framework creado el año 2012 por Bocoup que permite programar robots a través de JavaScript.

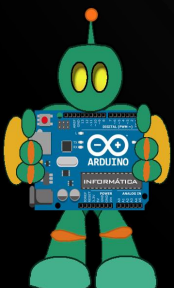
Fue creado para integrarse al desarrollo del internet de las cosas (IoT) junto a microcontroladores y microprocesadores

Página oficial:

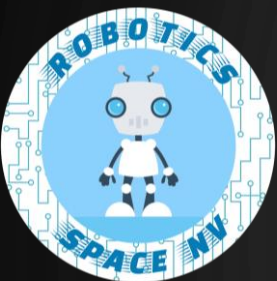
[Http://johnny-five.io/](http://johnny-five.io/)



TUTOR: NAGIB LUIS VALLEJOS M.



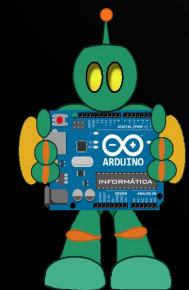


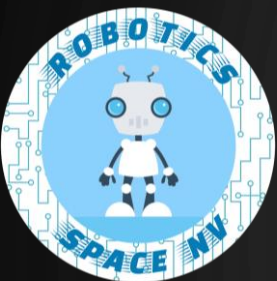


# ESTRUCTURA DE UN NODEBOT



TUTOR: NAGIB LUIS VALLEJOS M.

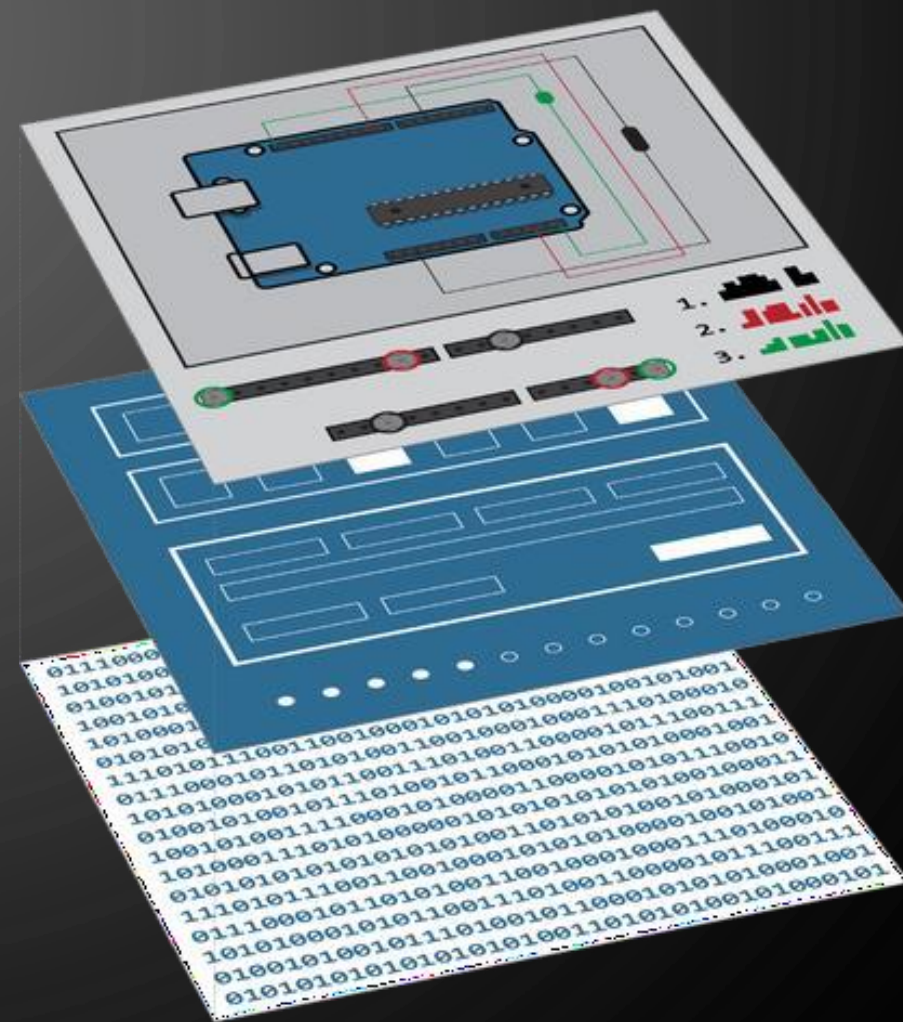




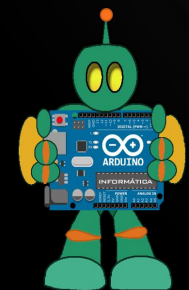
# FIRMATA

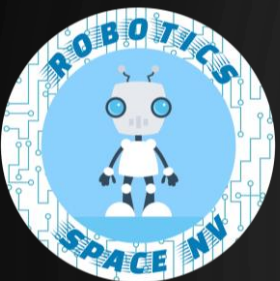
Es un protocolo que sirve para comunicarse con arduino desde cualquier software o cualquier ordenador conectado.

Tiene por objetivo hacer que arduino se vuelva en una extensión de nuestro entorno de desarrollo, lo cual es conveniente ya que nos evitaremos tener que programar en diferentes lenguajes de programación.



**TUTOR: NAGIB LUIS VALLEJOS M.**



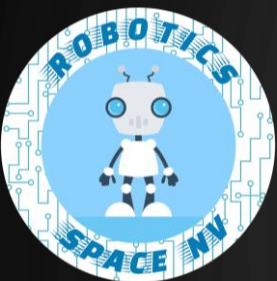


# FIRMATA - COMPATIBILIDAD

Actualmente firmata trabaja con diferentes tipos de microcontroladores, siendo arduino el microcontrolador más completo para el manejo de diferentes lenguajes



TUTOR: NAGIB LUIS VALLEJOS M.



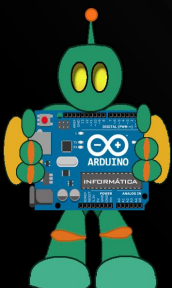
# J5 - PLACAS COMPATIBLES

**Johnny-Five** es compatible con diferentes tipos de microcontroladores y microprocesadores, entre las placas arduino compatibles tenemos:

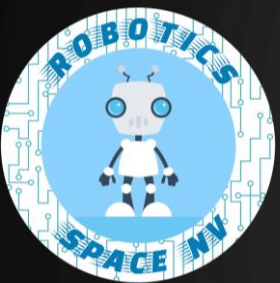
- Arduino uno
- Arduino Leonardo
- Arduino Mega
- Arduino Fio
- Arduino Micro
- Arduino Mini
- Arduino pro mini
- Arduino nano
- Intel Edison arduino



**TUTOR: NAGIB LUIS VALLEJOS M.**

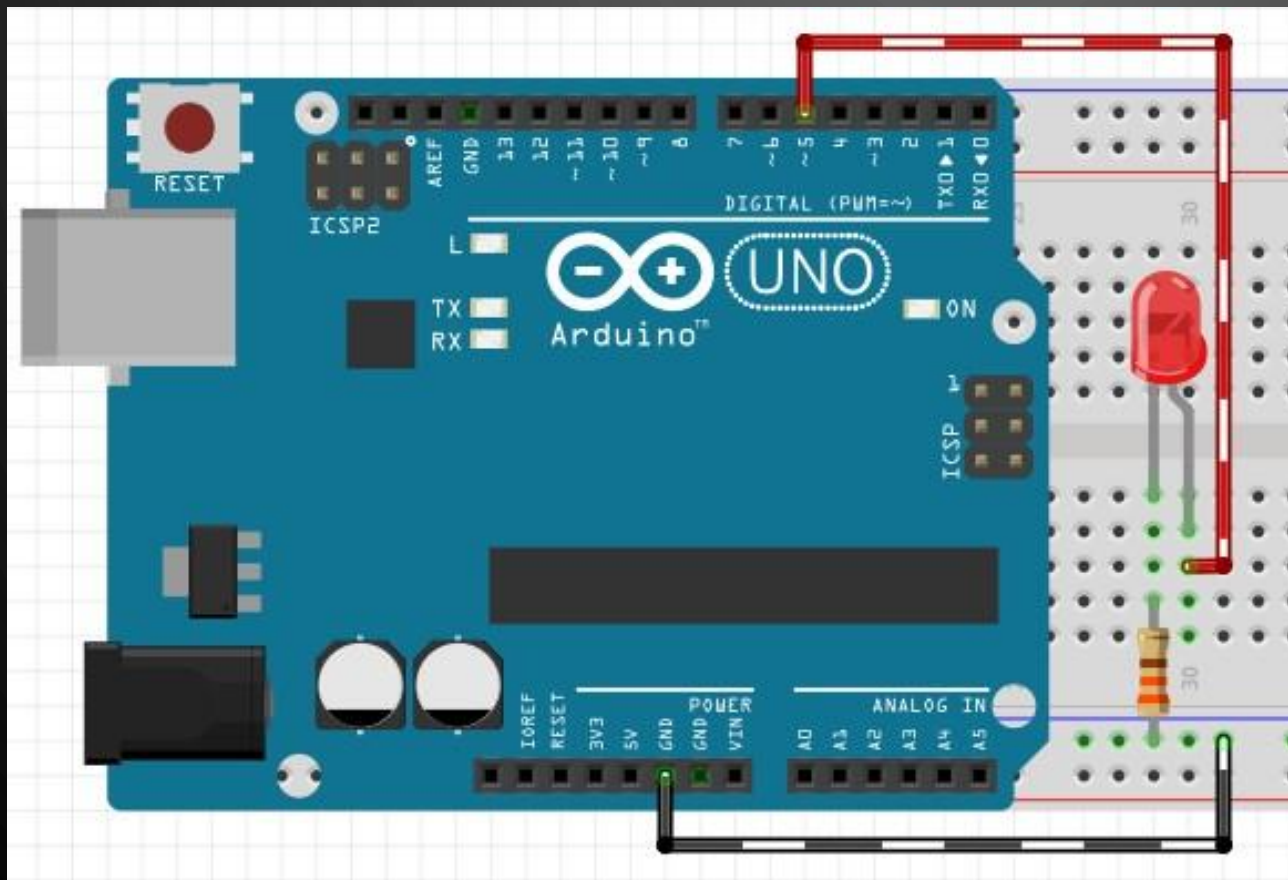




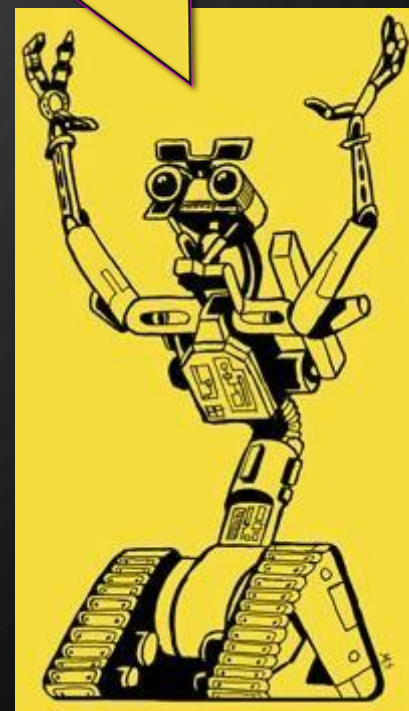


# EJERCICIO 1 - CIRCUITO

Encender y apagar un LED a razón de un segundo

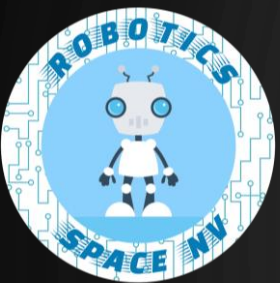


Hola mundo!



TUTOR: NAGIB LUIS VALLEJOS M.

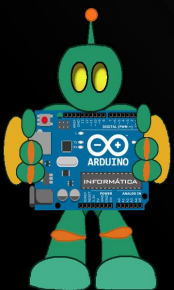




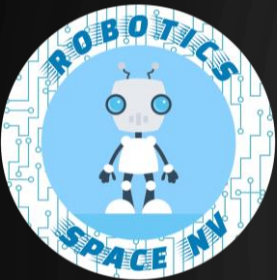
# EJERCICIO 1 - CREACIÓN DEL NODEBOT

Pasos a seguir:

1. Creamos el servidor
2. Nos dirigimos a la carpeta creada desde CMD
3. Instalamos J5: **npm install Johnny-Five**
4. Subimos el programa **standardFirmata** desde el IDE de arduino
5. Programamos en JavaScript
6. Ejecutamos el programa en CMD con el sufijo **node** por delante.



TUTOR: NAGIB LUIS VALLEJOS M.



# 1. CREAMOS EL SERVIDOR

En el escritorio creamos una carpeta a la cual la llamaremos: *nodebots*



## 2. Direccionamiento de la carpeta por consola

Ejecutamos CMD y nos direccionamos a la carpeta creada:

```

C:\> Símbolo del sistema

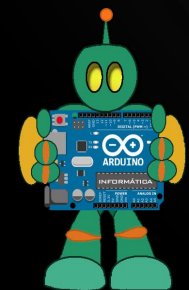
Microsoft Windows [Versión 10.0.19041.572]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

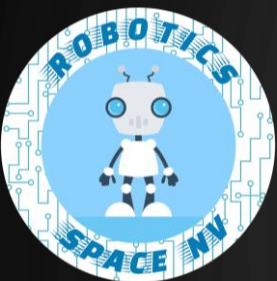
C:\Users\HP>cd desktop

C:\Users\HP\Desktop>cd nodebots

C:\Users\HP\Desktop\nodebots>
```

**TUTOR: NAGIB LUIS VALLEJOS M.**





# 3. INSTALACIÓN DE JOHNNY-FIVE

Para instalar Johnny-five, escribimos el comando:

***npm install johnny-five***

```
C:\Users\HP\Desktop\nodebots>npm install johnny-five

> @serialport/bindings@8.0.8 install C:\Users\HP\Desktop\nodebots\node_modules\@serialport\bindings
> prebuild-install --tag-prefix @serialport/bindings@ || node-gyp rebuild

> serialport@8.0.8 postinstall C:\Users\HP\Desktop\nodebots\node_modules\serialport
> node thank-you.js

Thank you for using serialport!
If you rely on this package, please consider supporting our open collective:
> https://opencollective.com/serialport/donate

npm WARN saveError ENOENT: no such file or directory, open 'C:\Users\HP\Desktop\nodebots\package.json'
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN enoent ENOENT: no such file or directory, open 'C:\Users\HP\Desktop\nodebots\package.json'
npm WARN nodebots No description
npm WARN nodebots No repository field.
npm WARN nodebots No README data
npm WARN nodebots No license field.

+ johnny-five@2.0.0
added 92 packages from 56 contributors and audited 92 packages in 40.365s

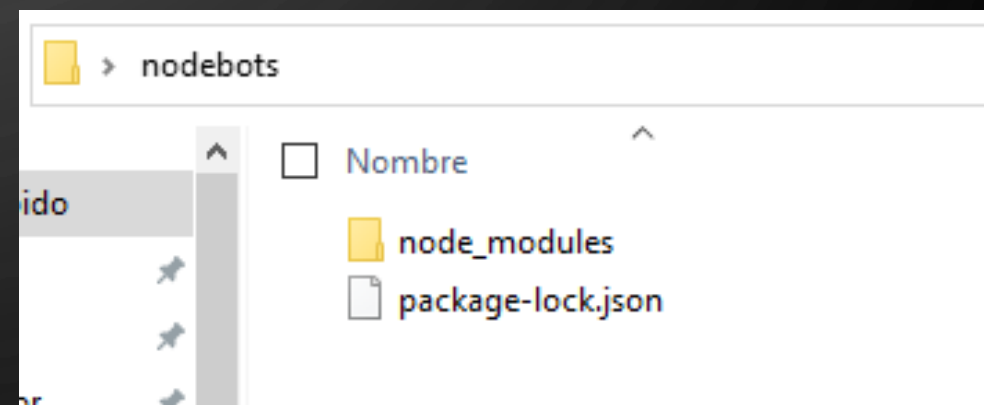
2 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

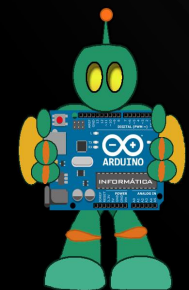
Abrimos la carpeta llamada **nodebots** y si la instalación fue exitosa aparecerá la carpeta de módulos de nodejs.

Esta carpeta contendrá los módulos de:

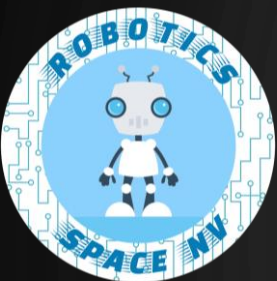
Johnny-five, serial-port, firmata entre otros.



**TUTOR: NAGIB LUIS VALLEJOS M.**





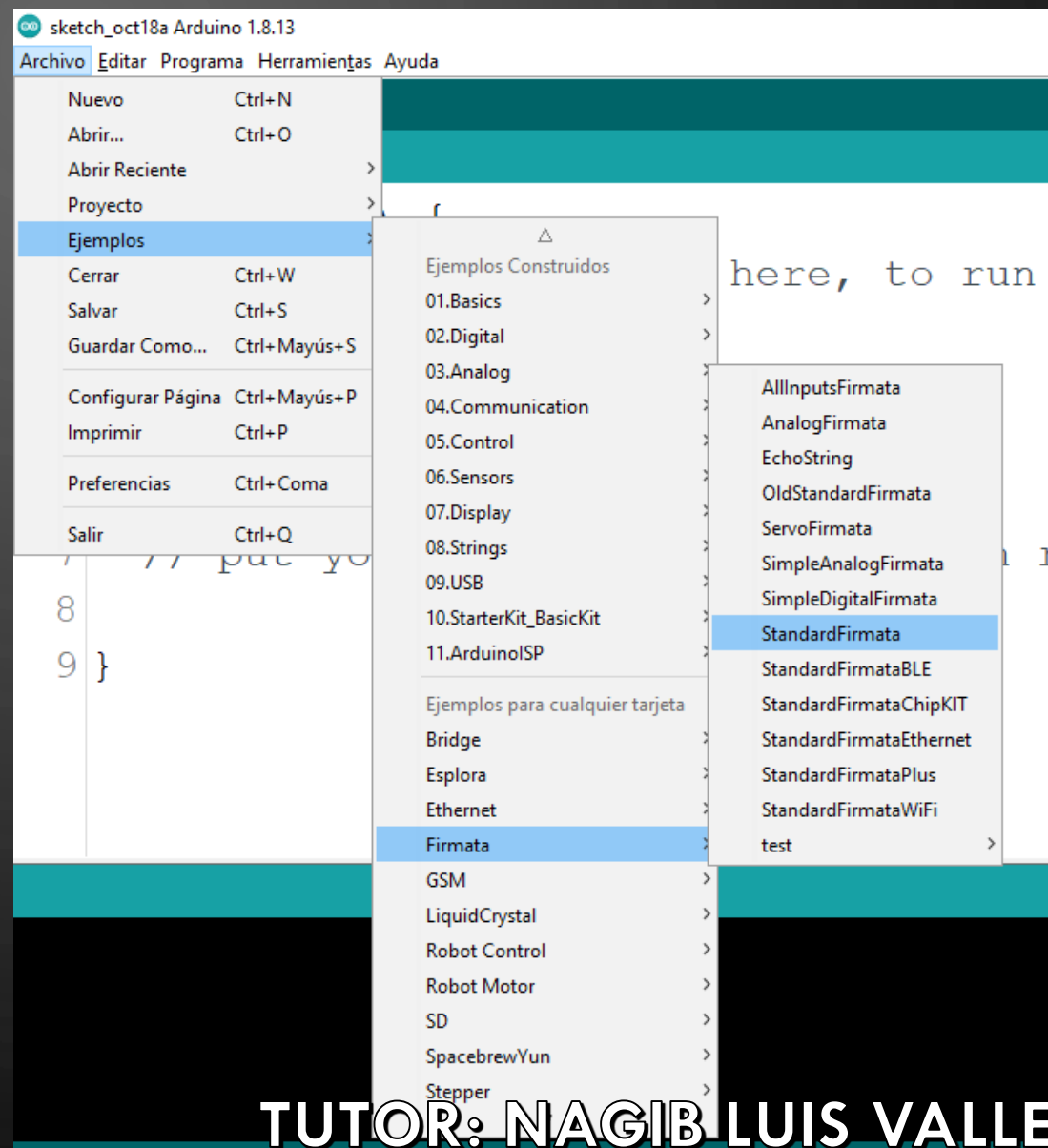


# 4. STANDARDFIRMATA - ARDUINO

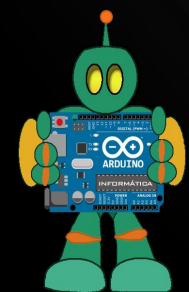
Abrimos el IDE de arduino y cargamos el programa StandardFirmata, para ello vamos a la siguiente dirección:

**Archivo -> Ejemplos -> Firmata -> StandardFirmata**

Seleccionamos la placa de trabajo, el puerto de conexión y subimos el programa.



**TUTOR: NAGIB LUIS VALLEJOS M.**





# EJERCICIO 1 - SOLUCIÓN

Encender y apagar un LED a razón de un segundo

JS ej1.js

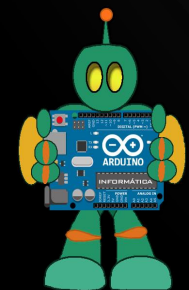
X

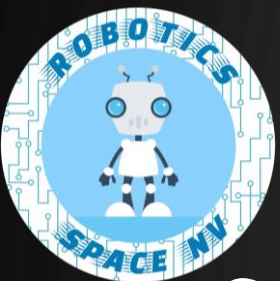
JS ej1.js > ...

```
1  var jf= require("johnny-five");
2  var tarjeta= new jf.Board();
3  tarjeta.on("ready",encender);
4
5  function encender(){
6      var led=new jf.Led(5);
7      led.blink(1000);
8  }
```

Existen varios métodos que tiene la función Led, entre ellas tenemos:  
*On, off, toggle, intensity, fadeIn, fadeOut, etc.*

TUTOR: NAGIB LUIS VALLEJOS M.





# EJERCICIO 1 - FUNCIONAMIENTO

Con la placa conectada ingresamos a CMD y nos dirigimos a la dirección donde guardamos **ej1.js** y lo ejecutamos con el comando: **node ej1.js**

Para terminar la conexión, presionamos **Ctrl+C**.

```
C:\WINDOWS\system32\cmd.exe - node ej1.js
Microsoft Windows [Versión 10.0.19041.572]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

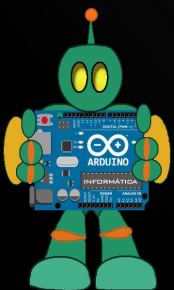
C:\Users\HP>cd desktop/nodebots

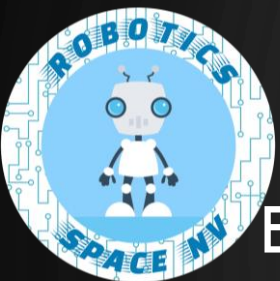
C:\Users\HP\Desktop\nodebots>node ej1.js
1602995219122 Available COM5
1602995219163 Connected COM5
1602995222970 Repl Initialized
>>
```

```
>>
(To exit, press ^C again or ^D or type .exit)
>>
1602995290193 Board Closing.

C:\Users\HP\Desktop\nodebots>
```

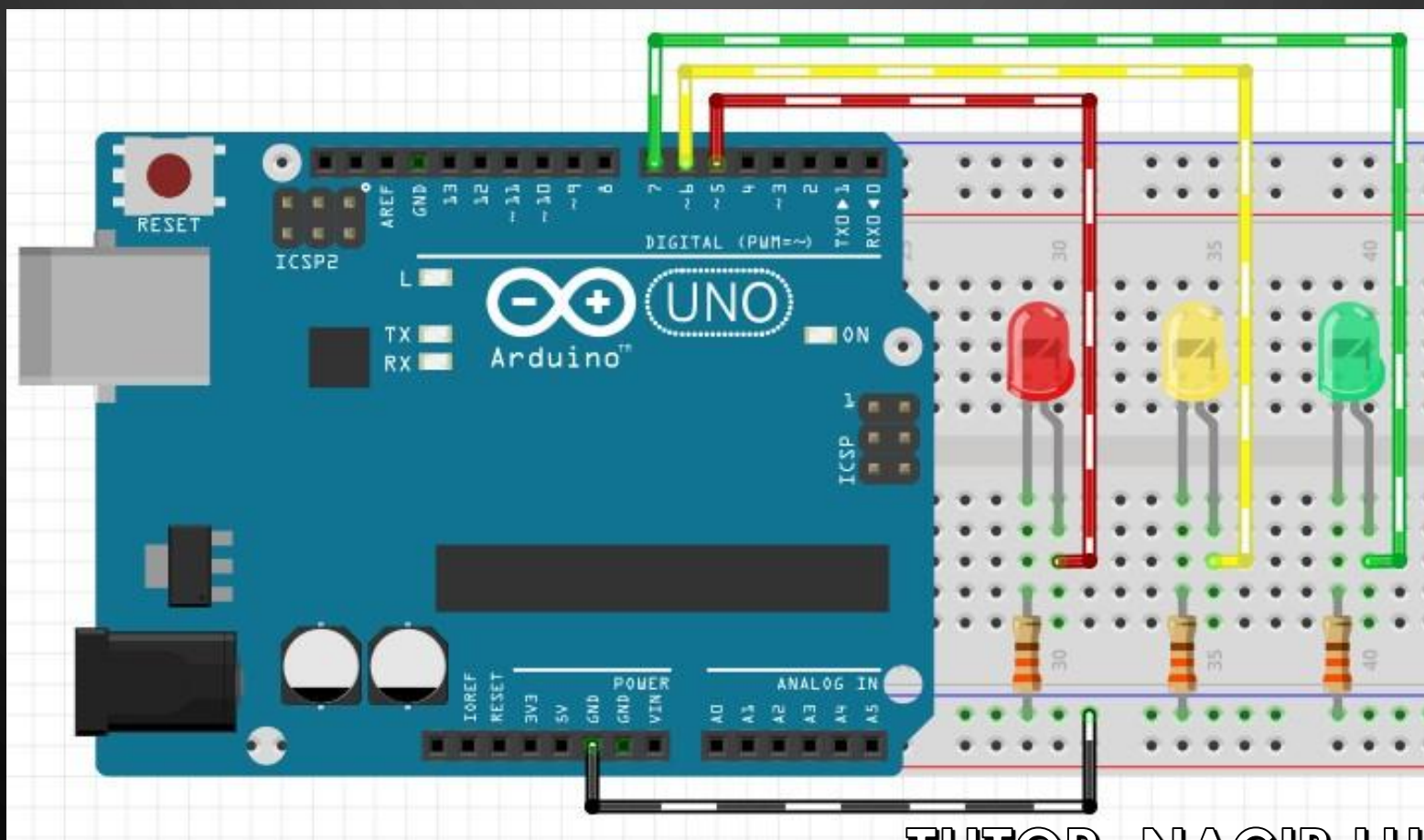
TUTOR: NAGIB LUIS VALLEJOS M.



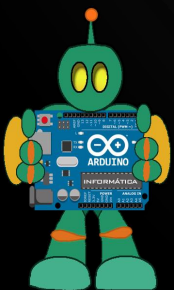


# EJERCICIO 2 - CIRCUITO

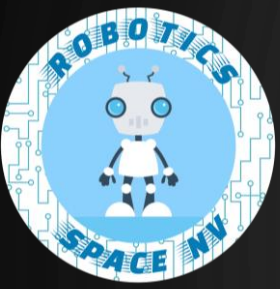
Encender y apagar 3 LED's de manera simultanea. El LED rojo se enciende a razón de 1 segundo, el amarillo a  $\frac{1}{2}$  segundo y el verde a  $\frac{1}{4}$  segundo.



TUTOR: NAGIB LUIS VALLEJOS M.







# EJERCICIO 2 - SOLUCIÓN

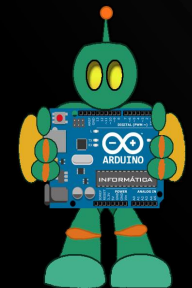
JS ej2.js

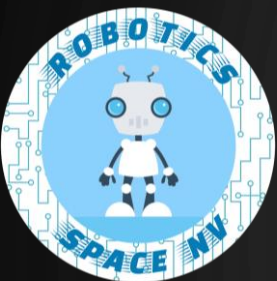


JS ej2.js > ...

```
1  var jf= require("johnny-five");
2  var tarjeta= new jf.Board();
3  tarjeta.on("ready",encender);
4
5  function encender(){
6      var rojo=new jf.Led(5);
7      var amarillo=new jf.Led(6);
8      var verde=new jf.Led(7);
9      rojo.blink(1000);
10     amarillo.blink(500);
11     verde.blink(250);
12 }
```

TUTOR: NAGIB LUIS VALLEJOS M.





# SENSORES Y ACTUADORES

**Johnny-Five** nos permite obtener datos de sensores analógicos y digitales y para ello requiere dos parámetros específicos para poder programarlo con JS.

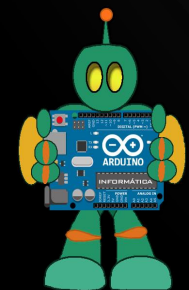
Sintaxis:

**Var x={pin, frecuencia}**

Posteriormente creamos un objeto de tipo Sensor en el cual almacenamos el parámetro de la variable x.



TUTOR: NAGIB LUIS VALLEJOS M.



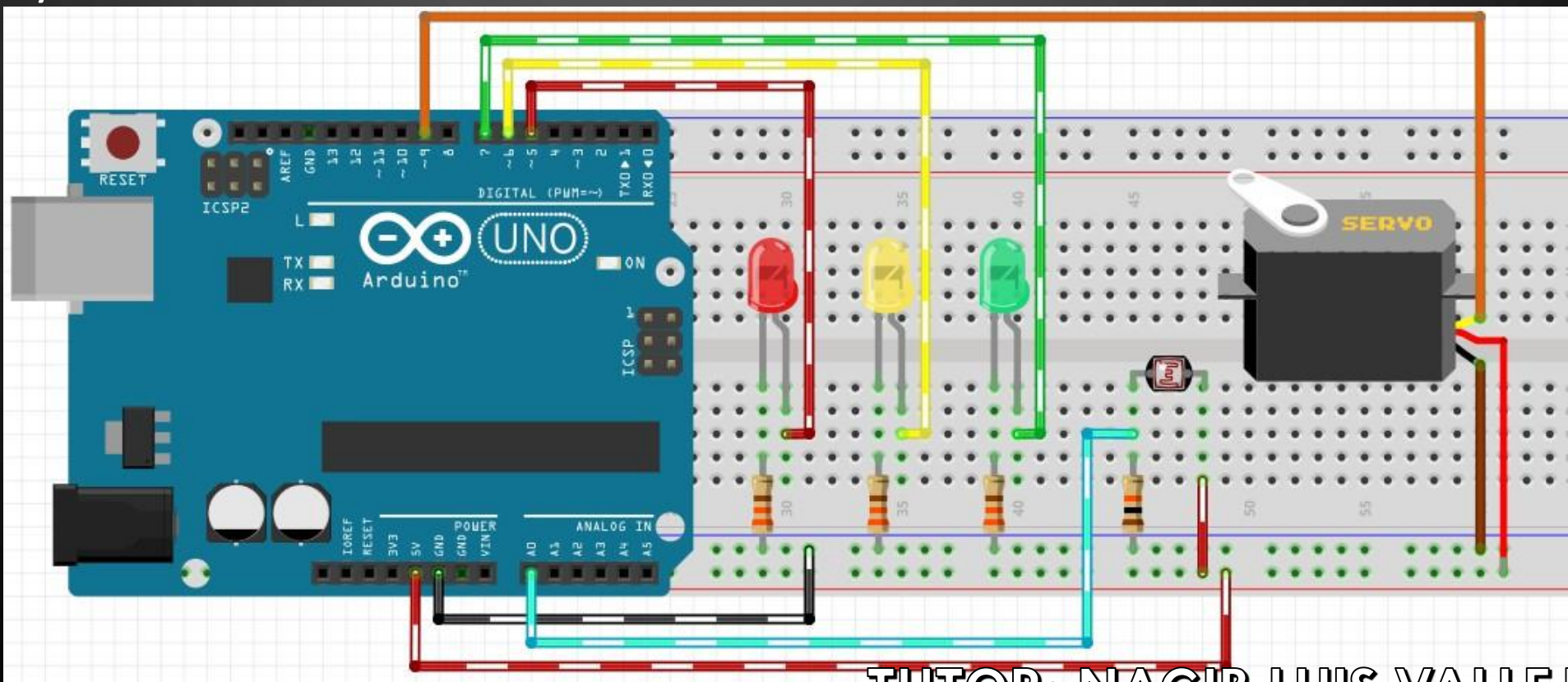




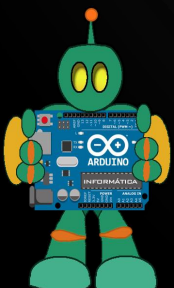
# EJERCICIO 3 - CIRCUITO

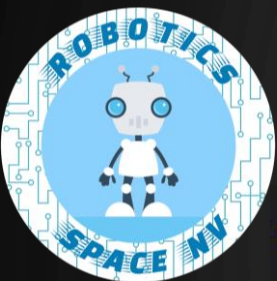
Si el valor censado por el LDR  $> 800$ , mover el servo entre  $60^\circ$  y  $120^\circ$ , de lo contrario se mantiene en  $160^\circ$ .

Si está en  $60^\circ$  se enciende solo el LED rojo, si está en  $120^\circ$  se enciende solo el LED verde y si está en  $160^\circ$  se enciende solo el LED amarillo



TUTOR: NAGIB LUIS VALLEJOS M.





# EJERCICIO 3 - SOLUCIÓN

JS ej3.js

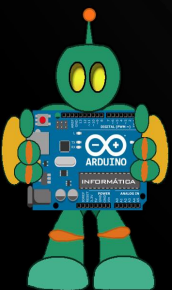


JS ej3.js > ondear

```
1  var jf= require("johnny-five");
2  var tarjeta= new jf.Board();
3  var ldr,servo,sw=0;
4  tarjeta.on("ready",mover);
5
6  function mover(){
7      var configuracion={pin:"A0",freq:50}
8      ldr = new jf.Sensor(configuracion);
9      servo= new jf.Servo(9);
10     servo.to(0);
11     ondear();
12 }
13 //recursividad
14 function ondear(){
15     var luz=ldr.value;
16     var rojo=new jf.Led(5);
17     var amarillo=new jf.Led(6);
18     var verde=new jf.Led(7);
19     console.log("Nivel de luminosidad:"+luz);
20     if(luz>800){//sw=0
21         amarillo.off();
```

```
22         if(sw==1){
23             sw=0;
24             servo.to(60);
25             rojo.on();
26             verde.off();
27         }
28         else{
29             sw=1;
30             servo.to(120);
31             rojo.off();
32             verde.on();
33         }
34     }
35     else{
36         servo.to(160);
37         amarillo.on();
38         rojo.off();
39         verde.off();
40     }
41     setTimeout(ondear,1000);
42 }
```

TUTOR: NAGIB LUIS VALLEJOS M.





# CONTACTOS



SUSCRÍBETE



(+591) 63096640



robotics.space.nv@gmail.com



fb.me/RoboticsSpaceNV



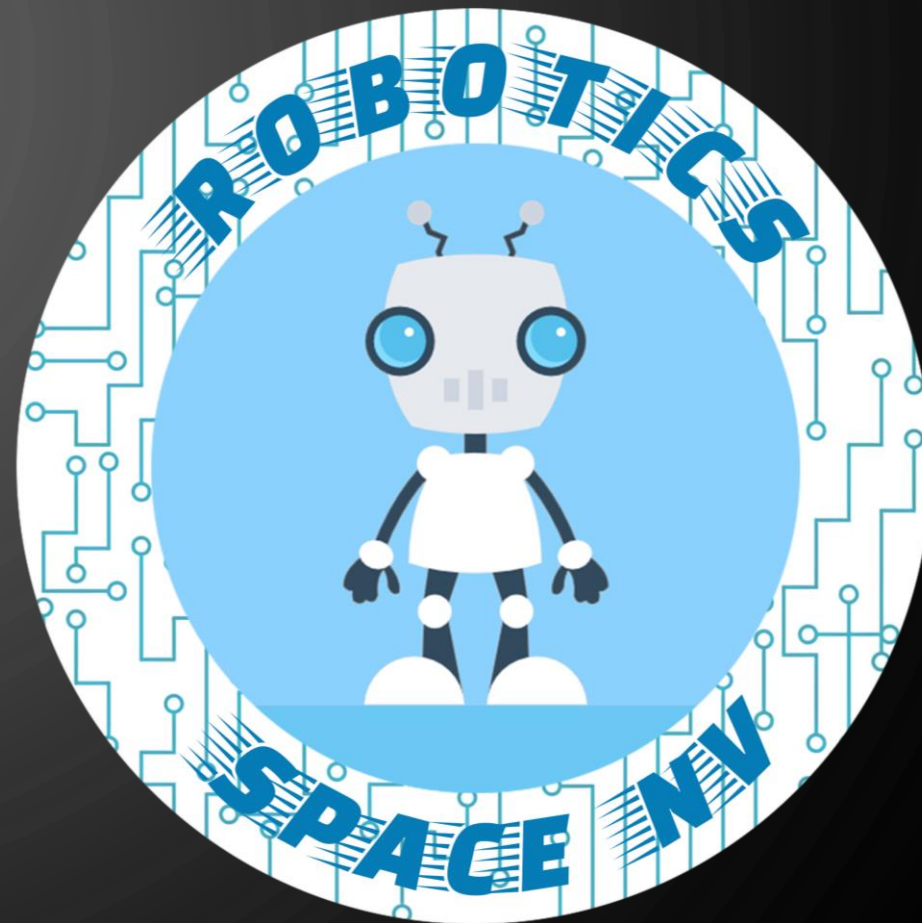
@NagibVallejos



Robotics Space NV



<https://github.com/nagibvalejos/Robotics-Space-NV>



TUTOR: NAGIB LUIS VALLEJOS M.