

PRUEBAS CON MILLIS

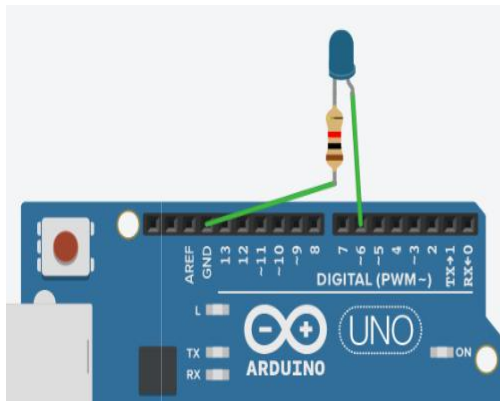
Ej 1) Haz que un led se encienda al segundo de conectar Arduino (y se mantenga encendido)

Código:

```
int pinled = 6; //Llamamos al pin 6 "pinled" para usarlo mas adelante
int par = 1000; //Tiempo en ms en el cual parpadeará el led
unsigned long comienzo = 0; //Tiempo en el que cambiará nuestro led
unsigned long tiempo = 0; //Momento actual en el programa
bool estadoled = LOW; //Posición o estado del led, si apagado o encendido

void setup() {
  pinMode(pinled,OUTPUT); //Configuramos pinled como salida
}

void loop() {
  tiempo = millis(); //Asociamos millis al tiempo actual
  if (tiempo >= comienzo + par) { //Si ha pasado más del tiempo o igual al
establecido...
    estadoled = !estadoled; //...cambiamos el estado del led
  }
  digitalWrite ( pinled, estadoled); //Marcamos en que estado se encuentra
ahora el led
}
```



```
1 int pinled = 6; //Llamamos al pin 6 "pinled" para usarlo mas adelante
2 int par = 1000; //Tiempo en ms en el cual parpadeará el led
3 unsigned long comienzo = 0; //Tiempo en el que cambiará nuestro led
4 unsigned long tiempo = 0; //Momento actual en el programa
5 bool estadoled = LOW; //Posición o estado del led, si apagado o encendido
6
7 void setup() {
8   pinMode(pinled,OUTPUT); //Configuramos pinled como salida
9 }
10
11 void loop() {
12   tiempo = millis(); //Asociamos millis al tiempo actual
13   if (tiempo >= comienzo + par) { //Si ha pasado más del tiempo o igual al establecido...
14     estadoled = !estadoled; //...cambiamos el estado del led
15   }
16   digitalWrite ( pinled, estadoled); //Marcamos en que estado se encuentra ahora el led
17 }
18
```

Nota:

Para este ejercicio, basándome en el ejemplo de millis de la presentación en las diapositivas solo he tenido que quitar del código (usando el del ejercicio resuelto a modo de guía y comprendiéndolo) la línea en la que relaciona el momento de cambio del estado de led con millis, de esta manera, por mucho que aumente el tiempo y que millis avance, no le decimos al programa que sobrescriba el estado del led de HIGH a LOW.

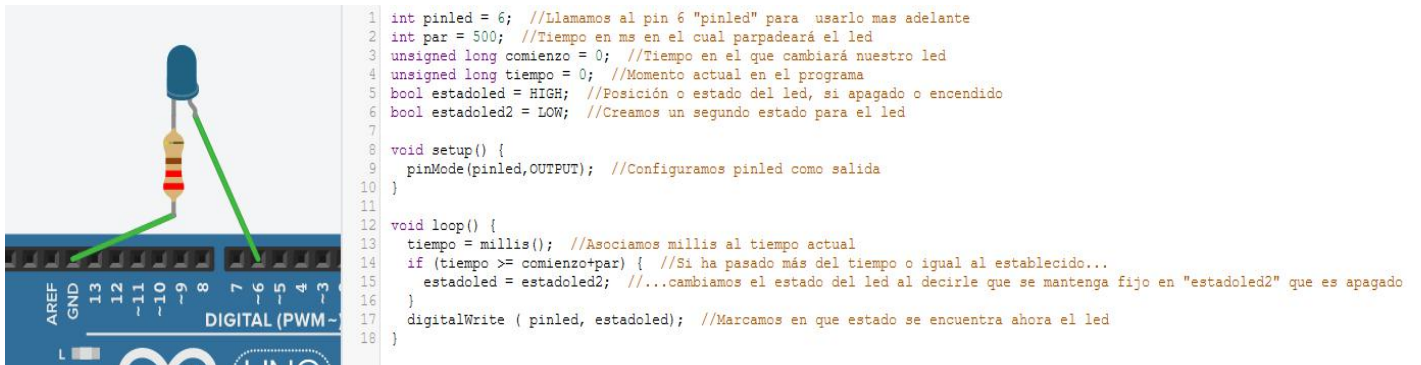
Ej 2) Haz que el led anterior se apague al medio segundo de encenderse

Código:

```
int pinled = 6; //Llamamos al pin 6 "pinled" para usarlo mas adelante
int par = 500; //Tiempo en ms en el cual parpadeará el led
unsigned long comienzo = 0; //Tiempo en el que cambiará nuestro led
unsigned long tiempo = 0; //Momento actual en el programa
bool estadoled = HIGH; //Posición o estado del led, si apagado o encendido
bool estadoled2 = LOW; //Creamos un segundo estado para el led
```

```
void setup() {
  pinMode(pinled,OUTPUT); //Configuramos pinled como salida
}

void loop() {
  tiempo = millis(); //Asociamos millis al tiempo actual
  if (tiempo >= comienzo+par) { //Si ha pasado más del tiempo o igual al
    establecido...
    estadoled = estadoled2; //...cambiamos el estado del led al decirle que
    se mantenga fijo en "estadoled2" que es apagado
  }
  digitalWrite ( pinled, estadoled); //Marcamos en que estado se encuentra
  ahora el led
}
```



Nota:

En este ejercicio, ya que pedía que el led quedase apagado a los 500ms de encenderse y permanecer en esta posición, lo que he hecho ha sido primero crear, mediante un nuevo "bool" un estado del led en apagado y más adelante, en el loop cuando se relacionaba el comienzo con millis, he sustituido ese millis por el estado del del que hemos llamado "estadoled2" y que lo ponía en estado apagado. Por otro lado he invertido el orden del led entre encendido y apagado para que, al conectar el arduino, nuestro led se encuentre ya en estado encendido y de esta forma, mediante el loop, poder pasarlo a los 500ms a estado apagado.

Ejercicio final

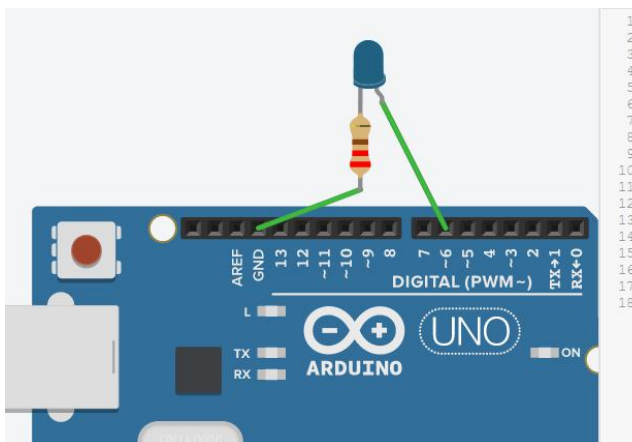
Haz que un led parpadee cada segundo con millis().

Código:

```
int pinled = 6; //Llamamos al pin 6 "pinled" para usarlo mas adelante
int par = 500; //Tiempo en ms en el cual parpadeará el led
unsigned long comienzo = 0; //Tiempo en el que cambiará nuestro led
unsigned long tiempo = 0; //Momento actual en el programa
bool estadoled = LOW; //Posición o estado del led, si apagado o encendido

void setup() {
  pinMode(pinled,OUTPUT); //Configuramos pinled como salida
}

void loop() {
  tiempo = millis(); //Asociamos millis al tiempo actual
  if (tiempo >= comienzo + par) { //Si ha pasado más del tiempo o igual al
establecido...
    estadoled = !estadoled; //...cambiamos el estado del led
    comienzo=millis(); //Reanudamos el contador de tiempo
  }
  digitalWrite ( pinled, estadoled); //Marcamos en que estado se encuentra
ahora el led
}
```



```
1 int pinled = 6; //Llamamos al pin 6 "pinled" para usarlo mas adelante
2 int par = 500; //Tiempo en ms en el cual parpadeará el led
3 unsigned long comienzo = 0; //Tiempo en el que cambiará nuestro led
4 unsigned long tiempo = 0; //Momento actual en el programa
5 bool estadoled = LOW; //Posición o estado del led, si apagado o encendido
6
7 void setup() {
8   pinMode(pinled,OUTPUT); //Configuramos pinled como salida
9 }
10
11 void loop() {
12   tiempo = millis(); //Asociamos millis al tiempo actual
13   if (tiempo >= comienzo + par) { //Si ha pasado más del tiempo o igual al establecido...
14     estadoled = !estadoled; //...cambiamos el estado del led
15     comienzo=millis(); //Reanudamos el contador de tiempo
16   }
17   digitalWrite ( pinled, estadoled); //Marcamos en que estado se encuentra ahora el led
18 }
```

Nota:

En este ejercicio, ya que nos pedía que a cada segundo parpadeara, la única modificación ha sido la de reducir el tiempo en el cual parpadeará el led de 1000 a 500 ms. De esta manera, y ya que cada vez que pasen dos veces por el valor 500ms realizará un ciclo de encendido y apagado, no hará falta modificar nada más del código.