

# Exit Coding Game

## How to code

Basics zu Arduino  
und Programmierung



**How to code**

**Arduino is a microcontroller that can be used to create physical interactions and IoT applications**



ARDUINO UNO REV3



ARDUINO MKR WIFI 1010  
(CONCEPTUALLY  
SIMILAR TO UNO BUT WITH WIFI)



ARDUINO UNO WIFI REV2  
(CONCEPTUALLY  
SIMILAR TO UNO BUT WITH WIFI)

## How to code

**Arduino can read data from sensors, process the data and output data to actuators**

### USB

- power supply
- upload of programs



Digital In- and Output Pins

- 0 or 1

How to code

# Some things Arduino can sense



Temperature

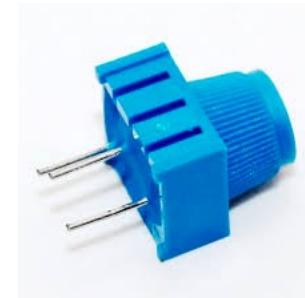
Light



Interaction  
(buttons)



Interactions  
(joystick)



Interactions  
(potentiometer)



Proximity  
(Range detector)

How to code

# Some things Arduino can do



Turn on  
light



Make  
noise



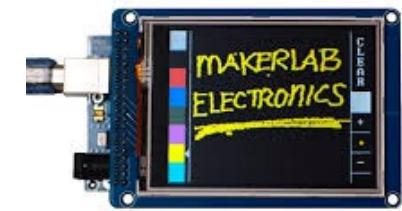
Display  
text



Control  
motor



Control  
servo

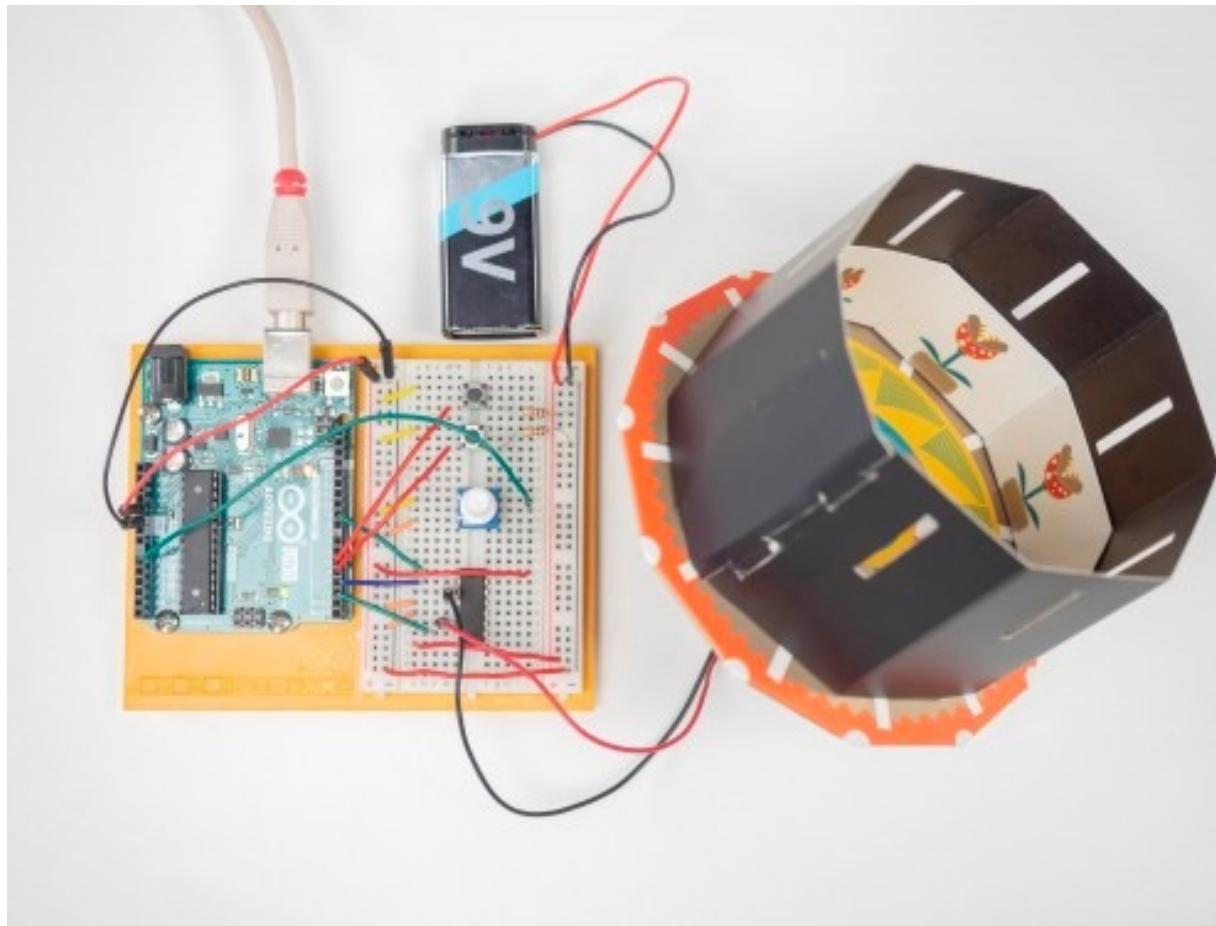


Display graphics  
(Raspberry PI  
is better at this...)

Wie könnten diese Aktoren helfen aus dem Raum zu entkommen?

## How to code

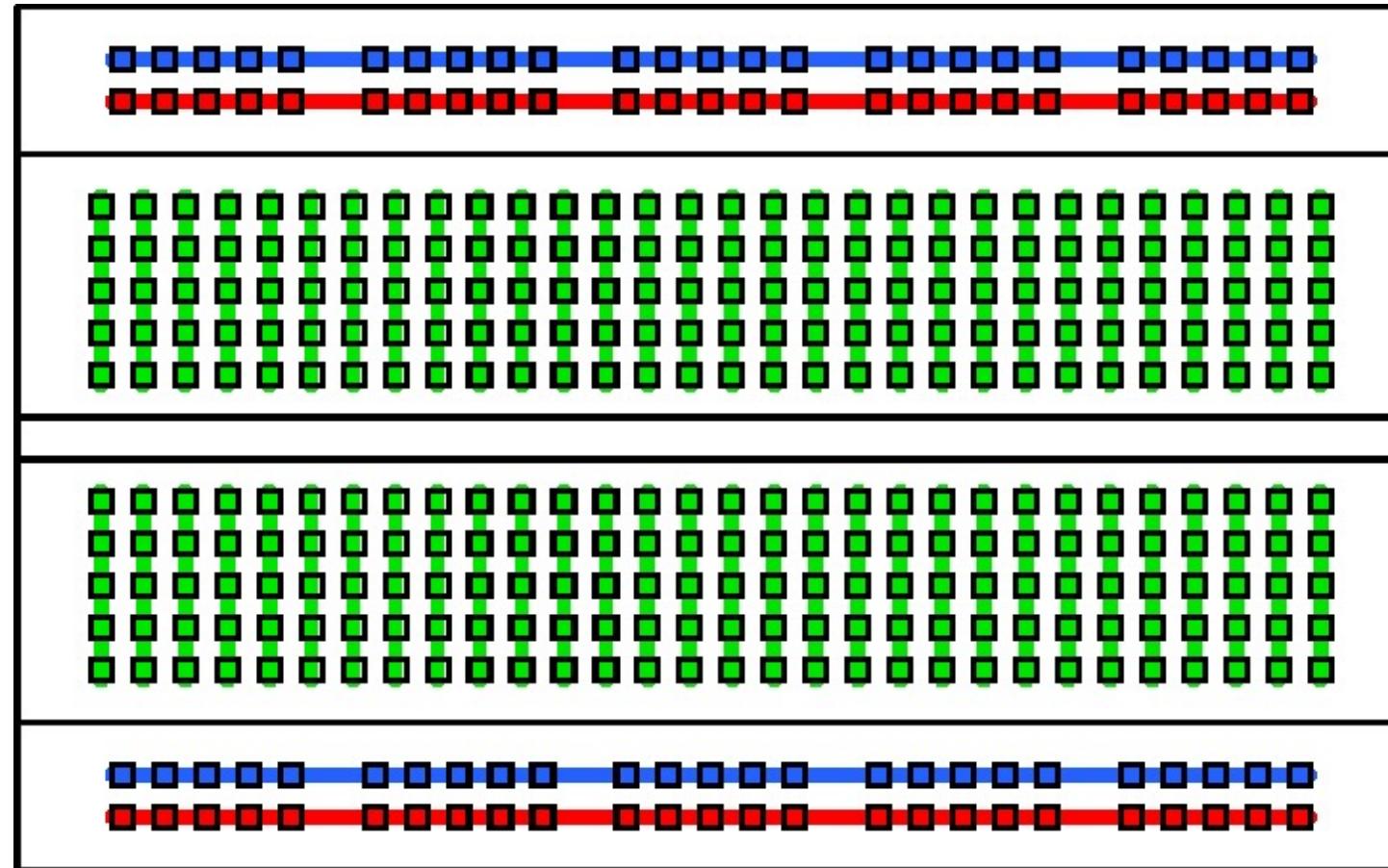
**Connecting sensors and actuators (i.e. creating circuits) is possible through a breadboard and jumper wires**



Arduino and breadboard

How to code

# Breadboard – How everything is connected

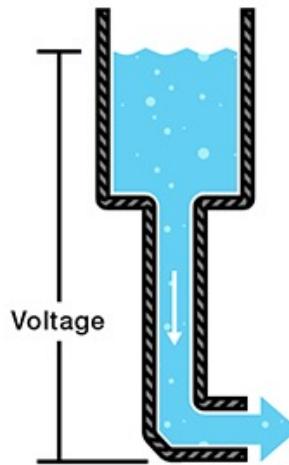


Source: <http://designbuildcode.weebly.com/breadboard-circuits.html>

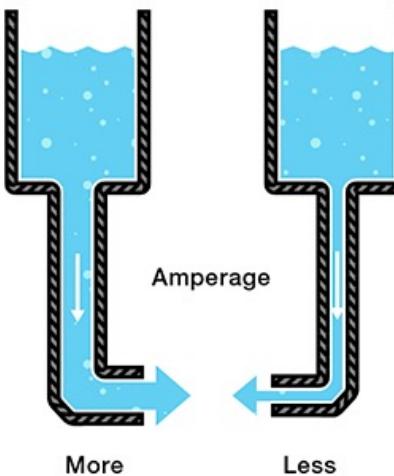
## How to code

# Basics of electronic circuits

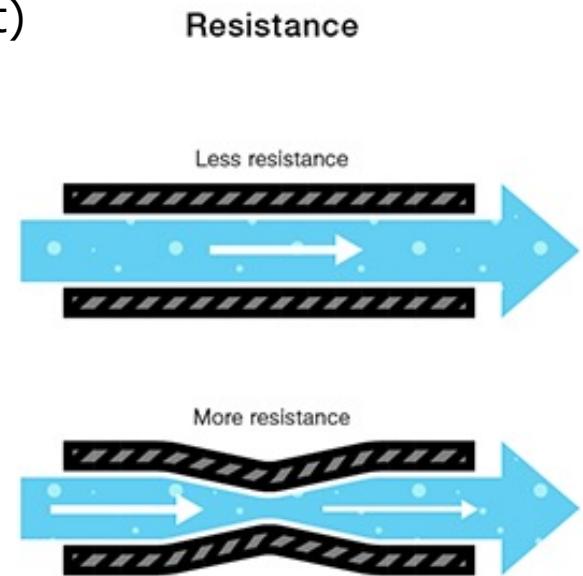
- **Voltage** is the difference in charge between two points.
- **Current** is the rate at which charge is flowing.
- **Resistance** is a material's tendency to resist the flow of charge (current)



Voltage (V) is the pressure  
at the end of the hose



More current (I) is flowing  
in wider hose

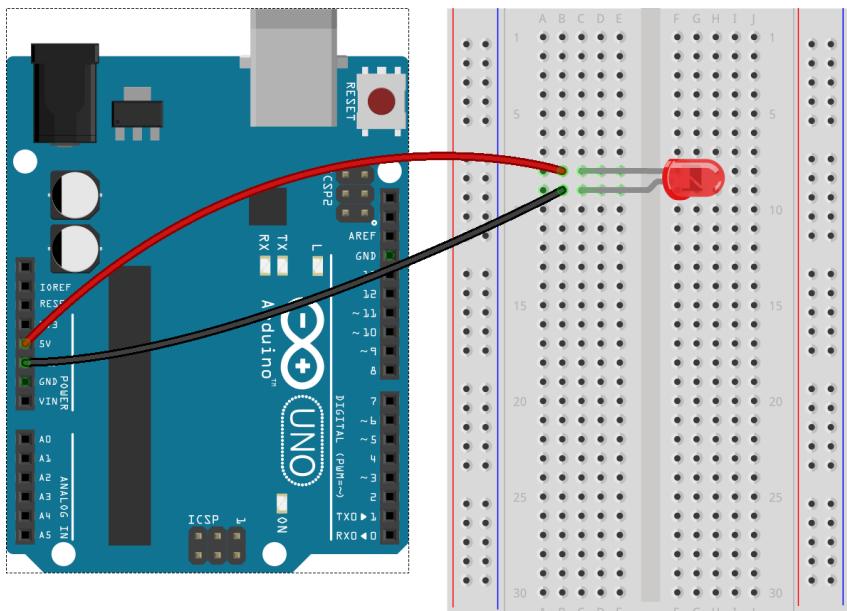


A resistor (R) limits the amount  
of charge that can flow

# How to code Some components require a limitation of current through a resistor

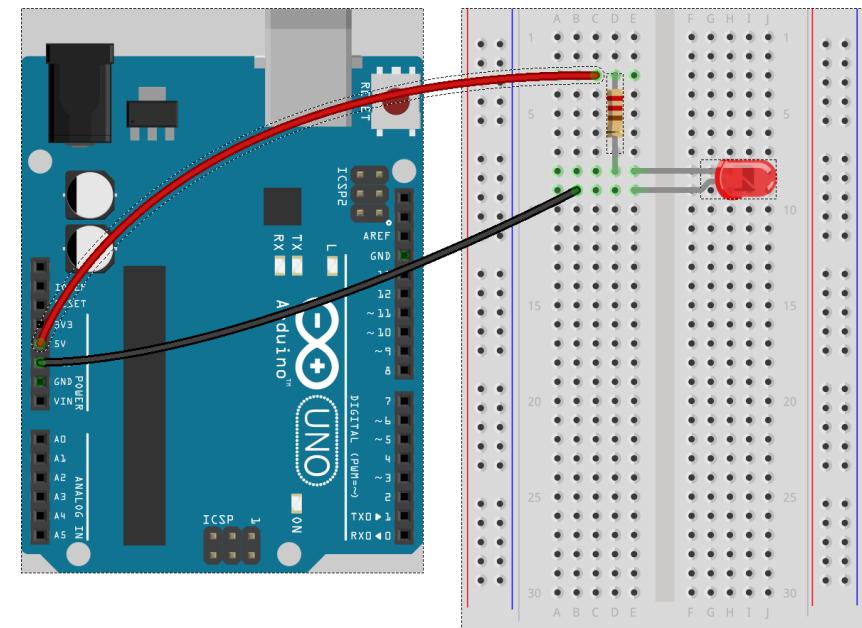


The current flows from the 5V pin to the GND pin



fritzing

**DO NOT TRY THIS**  
- It will fry your LED



fritzing

A resistor is needed to limit the amount of current that is flowing through the circuit

## How to code

**The Arduino IDE is a deliberately simple tool to create programs (sketches) for your microcontroller**

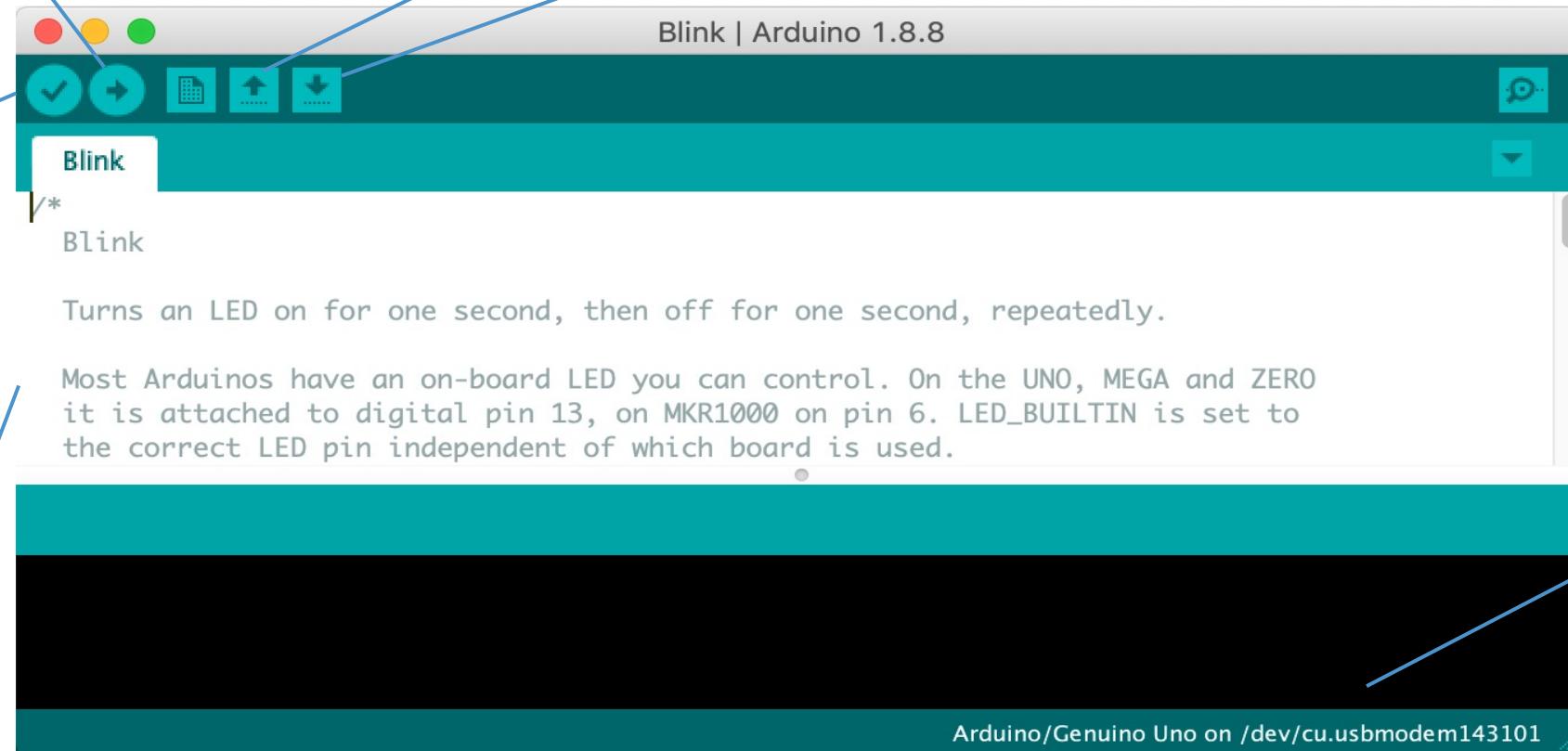
Upload code to  
Arduino

Open

Save

Compile your  
code (i.e.  
translate text  
into machine  
code that Arduino  
can understand)

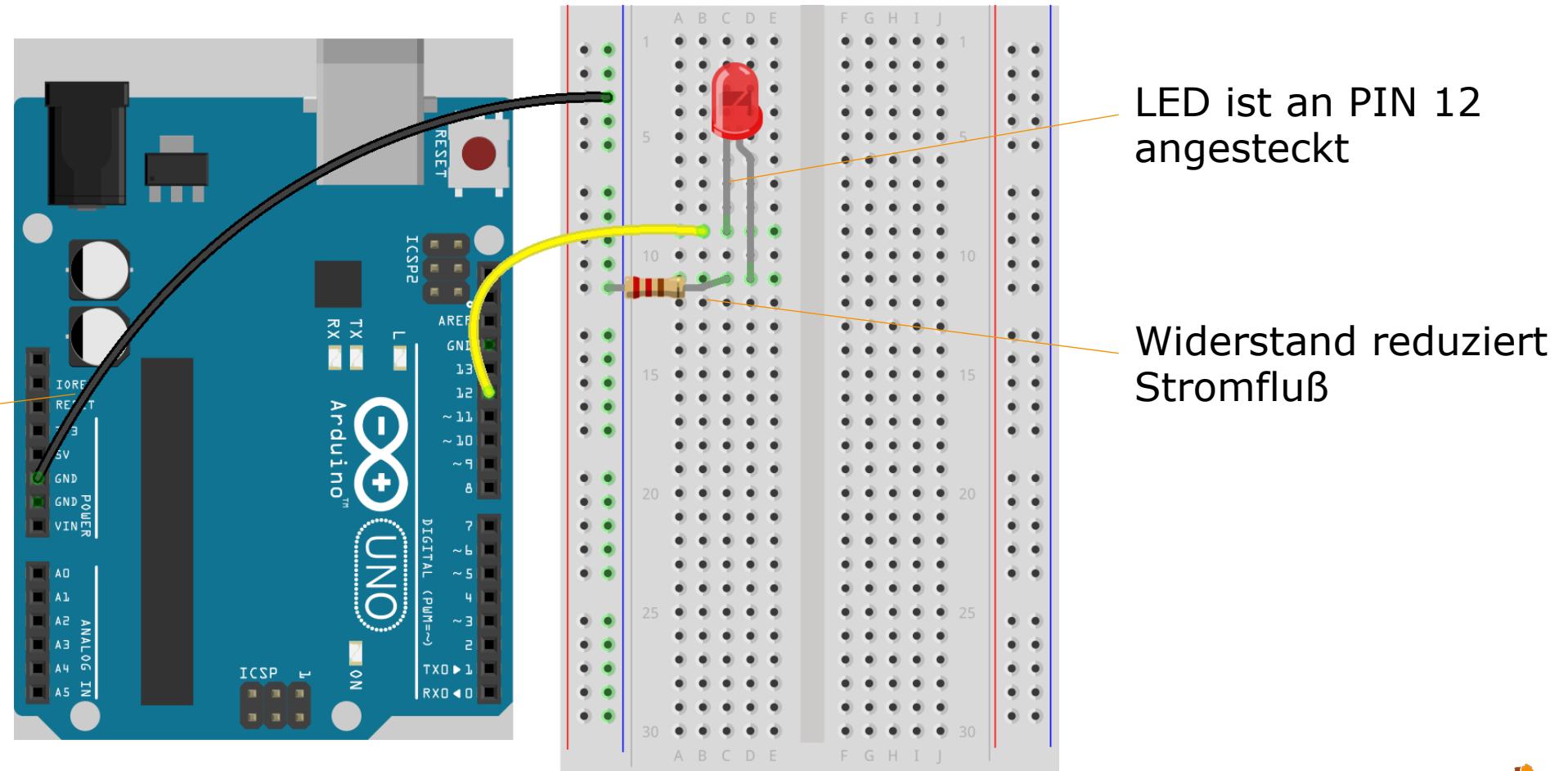
Code editor



Compiler and  
upload messages

How to code  
**Ein erstes, etwas spannenderes Beispiel – Eine LED soll blinken**

Stromkreis wird geschlossen



fritzing

## How to code

# Arduino - Variablen

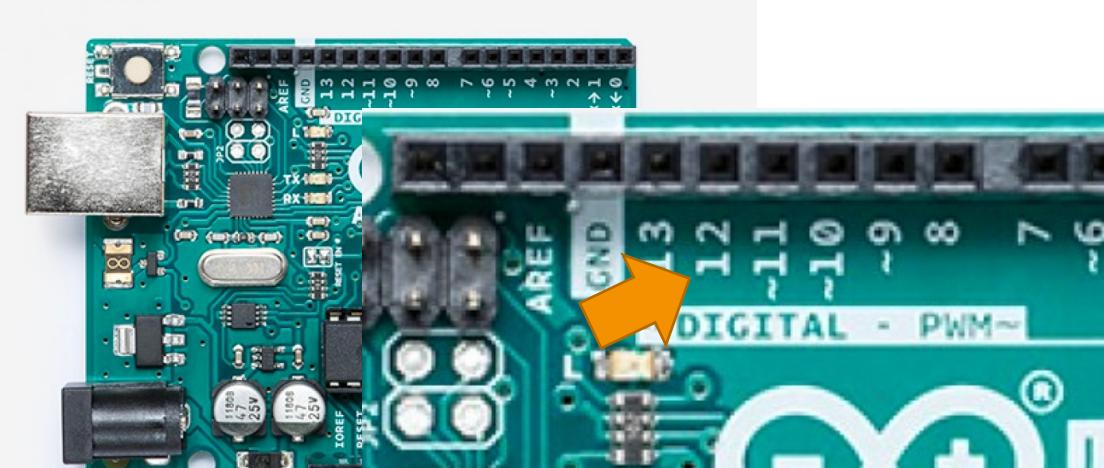
Legt eine Variable LED\_PIN an – Variablen sind “Schachteln im Computer”, in denen Werte abgespeichert werden, die man später wieder auslesen kann.

```
int LED_PIN = 12;
```

Speichert den Wert 12 in der Variable LED\_PIN  
(12 ist der PIN, an dem die LED angeschlossen ist)

```
void setup() {  
    pinMode(LED_PIN, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(LED_PIN, HIGH);  
    delay(1000);  
    digitalWrite(LED_PIN, LOW);  
    delay(1000);  
}
```



How to code

# Arduino - Funktionen

```
int LED_PIN = 12;  
  
void setup() {  
    pinMode(LED_PIN, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(LED_PIN, HIGH);  
    delay(1000);  
    digitalWrite(LED_PIN, LOW);  
    delay(1000);  
}
```

**setup** ist eine Funktion, d.h. eine Anweisung, die der Computer ausführen kann  
– Funktionen können weitere Funktionen enthalten.  
Setup wird immer ausgeführt wenn der Arduino startet  
(d.h. angesteckt wird)

**pinMode** ist eine weitere Funktion  
– Hier wird für den PIN 12, der Wert OUTPUT gesetzt, d.h. PIN 12 wird am Arduino in diesem Programm für die Ausgabe (Aktor) verwendet...

How to code

# Arduino - Funktionen

```
int LED_PIN = 12;
```

```
void setup() {  
    pinMode(LED_PIN, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(LED_PIN, HIGH);  
    delay(1000);  
    digitalWrite(LED_PIN, LOW);  
    delay(1000);  
}
```

Loop wird von Arduino laufend aufgerufen, d.h. alle Funktionen innerhalb von loop werden aufgerufen, dann geht es wieder von vorne los...

„Schaltet PIN 12 ein“ – LED leuchtet

Wartet eine Sekunde (PIN, d.h. LED bleibt „an“)

Schaltet LED aus

Wartet eine Sekunde (PIN, d.h. LED bleibt „aus“)

How to code

# Arduino - Schleifen

```
int LED_PIN = 12;

void setup() {
    pinMode(LED_PIN, OUTPUT);

    digitalWrite(LED_PIN, HIGH);
    delay(500);
    digitalWrite(LED_PIN, LOW);
    delay(500);

    digitalWrite(LED_PIN, HIGH);
    delay(500);
    digitalWrite(LED_PIN, LOW);
    delay(500);

    digitalWrite(LED_PIN, HIGH);
    delay(500);
    digitalWrite(LED_PIN, LOW);
    delay(500);
}

void loop() {
    // does nothing
}
```

Der Code lässt die LED 3-mal blinken.  
Das geht ist aber umständlich...

How to code

# Arduino - Schleifen

```
int LED_PIN = 12;

void setup() {
    pinMode(LED_PIN, OUTPUT);
    for (int i = 0; i < 3; i++) {
        digitalWrite(LED_PIN, HIGH);
        delay(500);
        digitalWrite(LED_PIN, LOW);
        delay(500);
    }
}

void loop() {
    // does nothing
}
```

Schleife wiederholt den Code innerhalb der geschweiften Klammern so oft wie in dem Ausdruck  $i < N$  – In diesem Beispiel also 3 mal...