****

Tinkercad Circuits Level 2:

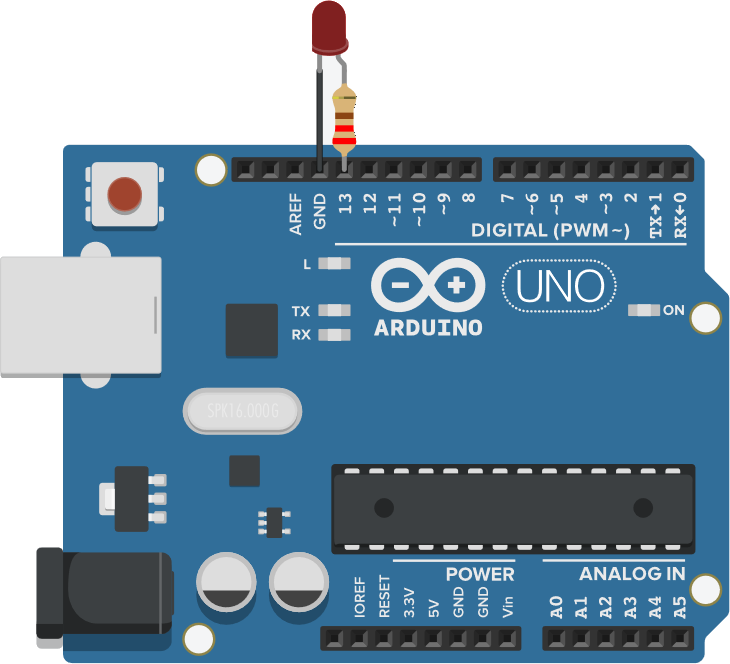
Programmiere einen Arduino und lass eine LED Blinken

**Mit „Tinkercad Circuits“ kannst du ganz einfach virtuelle elektrische Schaltungen bauen und dabei spielerisch die Welt der Elektronik entdecken! Du brauchst dafür nichts weiter als diese Anleitung und einen Computer oder ein Tablet mit Internetzugang.**

**Für diese Anleitung solltest du bereits das Level 1 gemacht haben, indem du lernst wie du in Tinkercad Circuits Bauteile miteinander verbindest und eine Simulation startest.**

**So geht’s:**

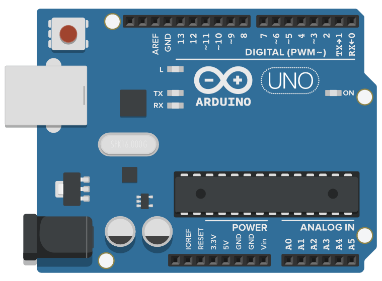
1. Öffne deinen Internet-Browser und logge dich auf [www.tinkercad.com](http://www.tinkercad.com) ein.
2. Klicke links auf  und dann auf 
3. Los geht‘s! Ziehe eine **LED**, einen **Widerstand** und einen **Arduino**   
     
   auf die Arbeitsfläche (siehe rechts). Mit der H:\Museumspädagogik\techLAB\easy things to do\_Grafikvorlagen\Keyboard_icons\300dpi\big\r_big.png -Taste kannst du ein angeklicktes Bauteil drehen



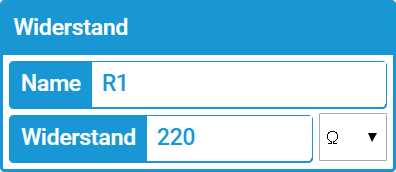
1. Verbinde die „Anode“ der LED (das lange Beinchen) mit dem Widerstand und die andere Seite des Widerstands mit „**Pin 13**“ am Arduino (siehe rechts). Die Kathode der LED (das kurze Beinchen) wird mit einem Draht mit dem „GND“ Pin verbunden.

Tipp: *Schau dir nochmal* ***Level 1*** *an, falls du Hilfe beim Verbinden der Bauteile brauchst.*

Was ist ein Arduino?

**Ein Arduino ist ein kleiner Computer, den du selbst programmieren kannst. Mit ihm kannst du LEDs ein- und ausschalten, Druck- und Drehknöpfe lesen, Sensormessungen an deinen Computer schicken und vieles mehr. Die Bauteile schließt du an seinen Pin-Anschlüssen an. Der Arduino kann alles was du brauchst um elektronische Musikinstrumente, nützliche Maschinen oder deine eigene Roboterarmee zu bauen!**

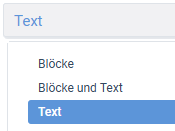
1. Klicke auf den Widerstand H:\Museumspädagogik\techLAB\easy things to do\P_Circuits_Level1_LED\gfx\Widerstand_freigestellt_2.png und ändere im kleinen Fenster rechts oben seinen Wert auf 220 Ω. Achte auf die richtige Einheit Ω !



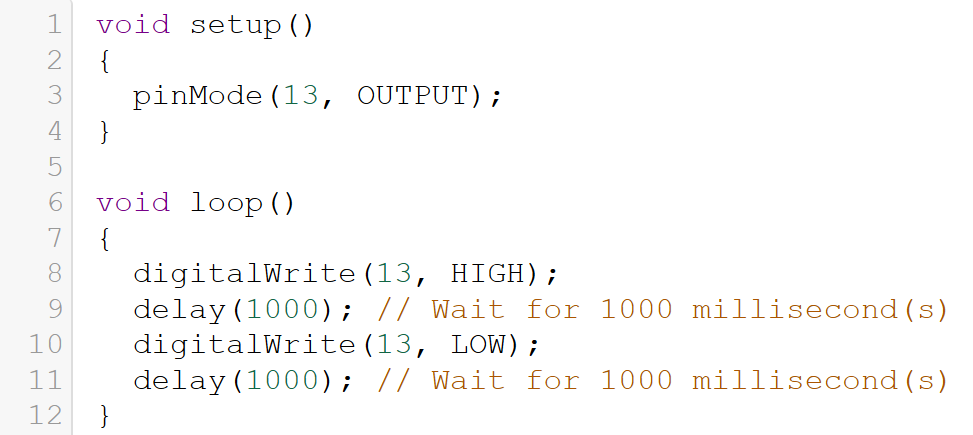
1. Klicke auf . Die Led sollte nun Blinken!  
   Klicke auf  bevor du weitermachst. Blinkt deine LED **nicht**?   
   🡪 Überprüfe, ob alle Teile richtig miteinander verbunden sind!

Warum blinkt die LED eigentlich?

Der Programm-Code macht‘s!

1. Klicke rechts oben auf  und wähle im Menü darunter „Text“ aus (bestätige die Warnung). Jetzt siehst du rechts den Code, der die LED blinken lässt. Der Code ist für den Arduino eine Art „Schritt-für-Schritt Anleitung“ – er macht genau das, was du ihm sagst!

Arduino Code-Struktur

Ein Arduino-Programm besteht aus den zwei Hauptfunktionen **setup()** und **loop()**.Alle Befehle die zwischen den geschwungenen Klammern **{ }** stehen, werden vom Arduino nacheinander ausgeführt. Jeder Befehl **muss** mit einem Strichpunkt **;** enden.

Alles was in **loop()** steht, wird danach endlos wiederholt.

Alles was in **setup()** steht, wird nach dem Einschalten nur einmal ausgeführt.

Doppelte Schrägstriche **//** Kennzeichnen ein Kommentar. Alles was **danach** in derselben Zeile steht (in Orange), wird vom Arduino ignoriert. Du kannst // verwenden, um dir im Code Notizen zu machen!

1. Sehen wir uns den Code etwas genauer an! Der Befehl in Zeile 8:



bewirkt, dass der Pin 13 (an den du die LED angeschlossen hast) „eingeschaltet“ und mit Strom versorgt wird. **Die LED leuchtet!** Der Befehl in der nächsten Zeile:



lässt den Arduino für 1000 Millisekunden (= 1 Sekunde) warten. Der Pin 13 wird weiter mit Strom versorgt und die LED leuchtet, sonst passiert nichts.



Schaltet den Strom auf Pin 13 wieder aus. Die LED leuchtet nicht mehr.

1. Jetzt ist es Zeit selbst zu programmieren! Ändere die Dauer der beiden Wartezeiten von delay(1000)auf delay(200). Was Passiert? Probiere verschiedene Werte aus!

[](https://emojipedia.org/google/android-10.0-march-2020-feature-drop/thinking-face/)Bonus-Level!

Mache folgenden Änderungen an deinem Schaltkreis und im   
Programm-Code (vergiss nicht die Simulation zu stoppen/starten):

* Verbinde den Widerstand mit Pin 11 anstelle von Pin 13. Blinkt die LED, wenn du die Simulation startest?
* Suche die Stellen im Programmcode, die du ändern musst, um die LED wieder zum Blinken zu bekommen. Tipp: *Du musst den Code an 2 Stellen ändern!*
* Versuche 2 LEDS an den Arduino anzuschließen und sie abwechselnd blinken zu lassen. Du kannst ihnen auch unterschiedliche Farben geben.
* Du könntest mit deinem Wissen sogar eine Ampelschaltung nachbauen!

Im **techLAB** haben wir „echte“ Arduinos und viele elektronische Bauteile mit denen du experimentieren kannst! Im techLAB gibt es aber noch viel mehr spannende Technologien zu entdecken. Arbeite mit 3D-Drucker, Laser Cutter, Stickmaschinen und mehr im **Technischen Museum Wien**. Die Öffnungszeiten des techLAB findest du [online](https://www.technischesmuseum.at/techlab). Der Eintritt ins Museum ist unter 19 Jahren gratis!   
  
**Wir freuen uns auf deinen Besuch!**