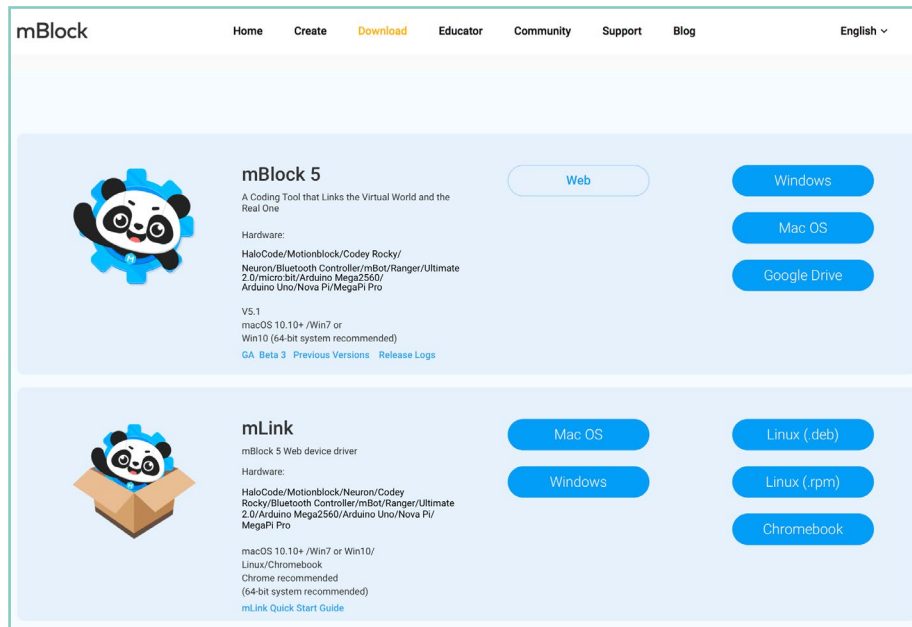
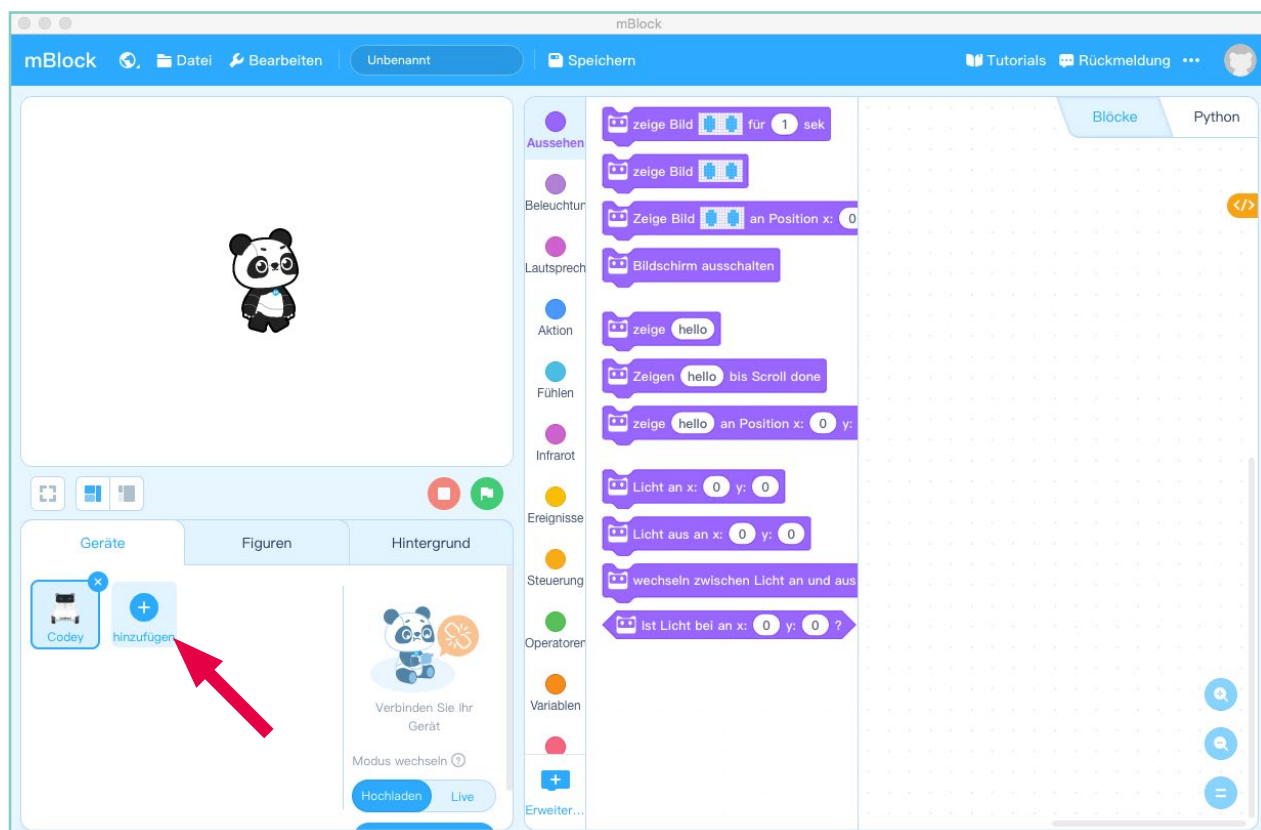


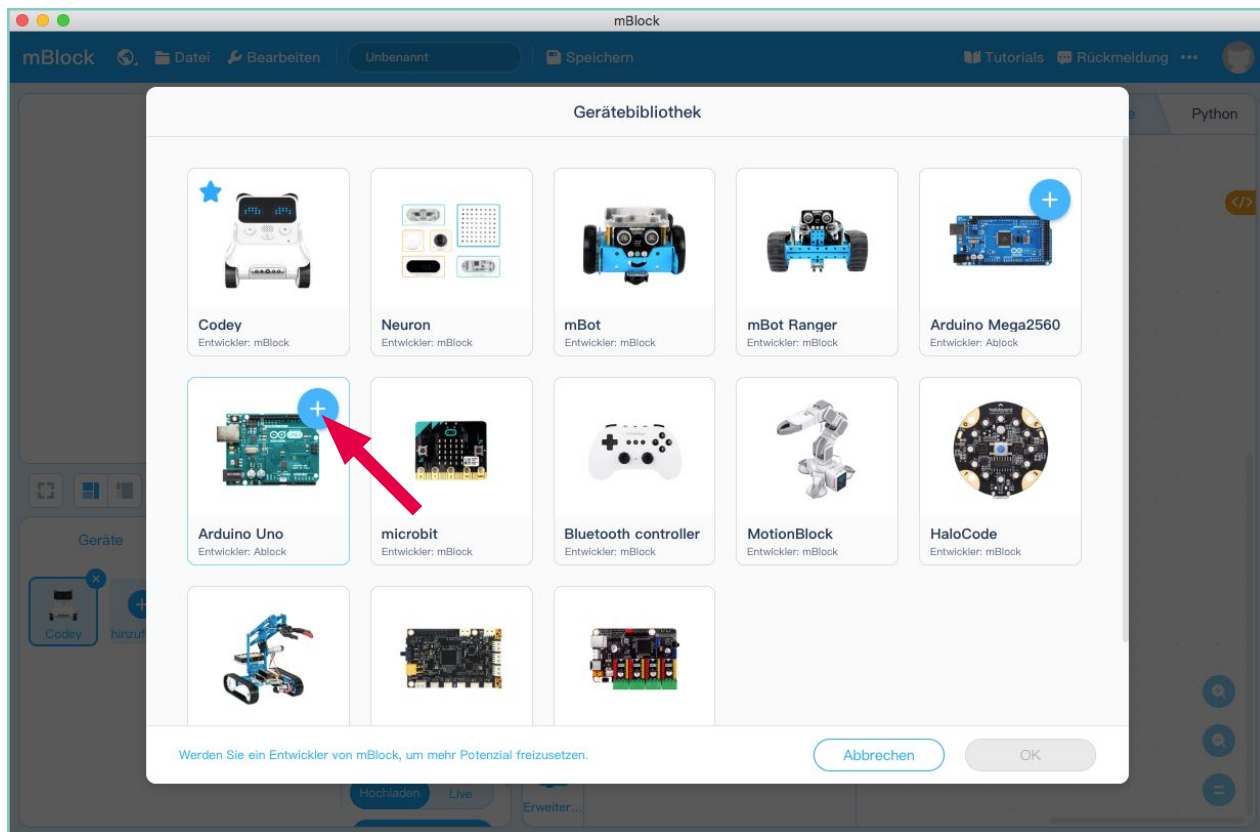
Auf [www.mblock.cc/mblock-software/](http://www.mblock.cc/mblock-software/) die Version für dein Betriebssystem herunterladen. Aktuell (Juli 2019) ist die Version **mBlock 5**. Die Software **mLink** brauchst du nur dann, wenn du **mBlock** nicht herunterladen und installieren willst, sondern die **Web**-Variante im Browser benutzen möchtest. **mLink** stellt in diesem Fall die Verbindung zwischen dem Browser und dem **Arduino** über USB her. Wir empfehlen, die Offline-Version, also **mBlock 5**, zu verwenden.



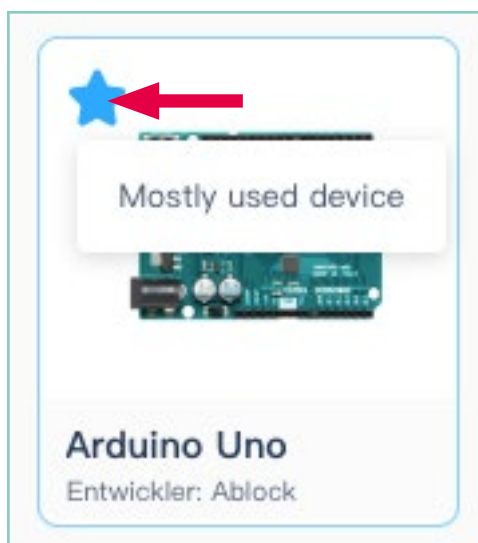
Nach der Installation startest du **mBlock** und klickst unten links bei **Geräte** auf **+ hinzufügen**



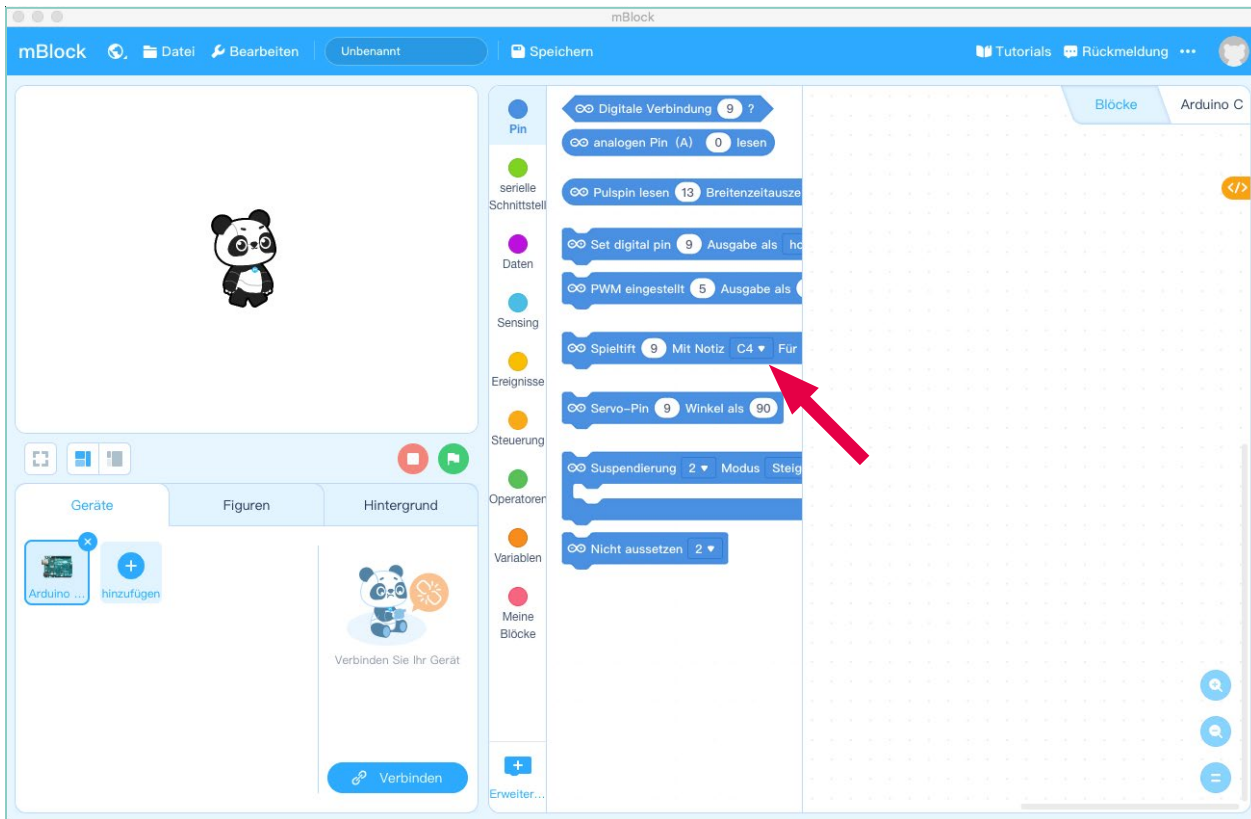
Klick bei **Arduino Uno** auf das runde blaue **Plus-Symbol**. Das Programm lädt jetzt noch kurz das **Arduino-Modul** herunter. Danach klickst du nochmals auf den **Arduino Uno** und dann auf **OK**. Das Modul muss nur einmalig heruntergeladen werden, danach kannst du mBlock auch offline verwenden.



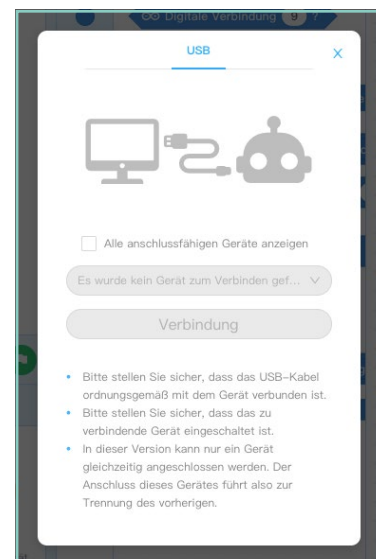
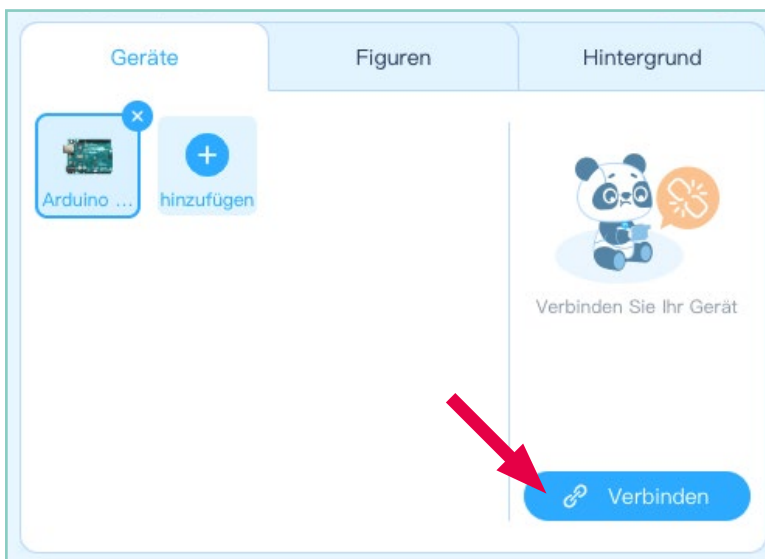
Tipp: Wenn ihr beim **Arduino Uno** auch noch zusätzlich auf das **Sternchen** klickt, ist er beim nächsten Programmstart automatisch als Standard-Gerät ausgewählt. Wenn ihr **mBlock** nur bzw. hauptsächlich für die Programmierung von **Arduinos** benutzen möchtet, spart das euren Teilnehmenden Zeit und mögliche Missverständnisse.



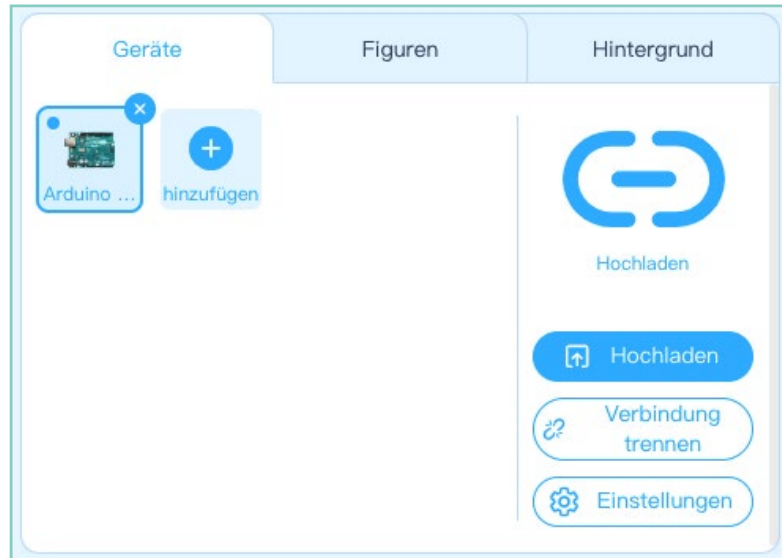
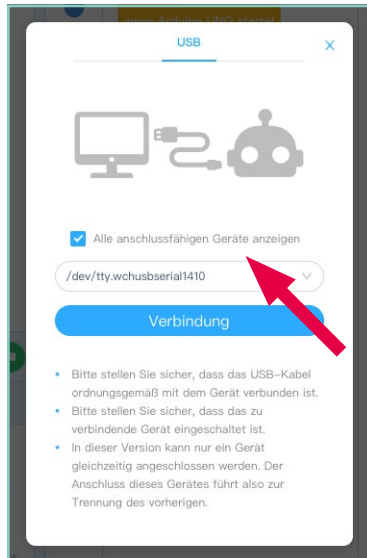
Mit der Auswahl des **Arduino Uno** haben sich auch die verfügbaren Programmier-Bausteine geändert. Vergleiche das folgende Bild mal mit dem Screenshot auf Seite 1. Deshalb ist es wichtig, das richtige Gerät auszuwählen, bevor man anfängt zu coden!



Schließe jetzt den **Arduino** an den USB-Port deines Computers an und klicke dann auf den blauen **Verbinden-Button** im **Geräte-Fenster**. Jetzt erscheint ein Popup mit weiteren Optionen. Sollte der Button **Verbindung** ausgegraut sein, wie hier im Screenshot, dann setze das Häkchen bei **Alle an-schlussfähigen Geräte anzeigen** und schaue, ob dann ein Eintrag im Feld direkt darunter auftaucht. Wenn es auch jetzt grau bleibt, dann schau auf der letzten Seite bei der Fehlerbehebung nach.

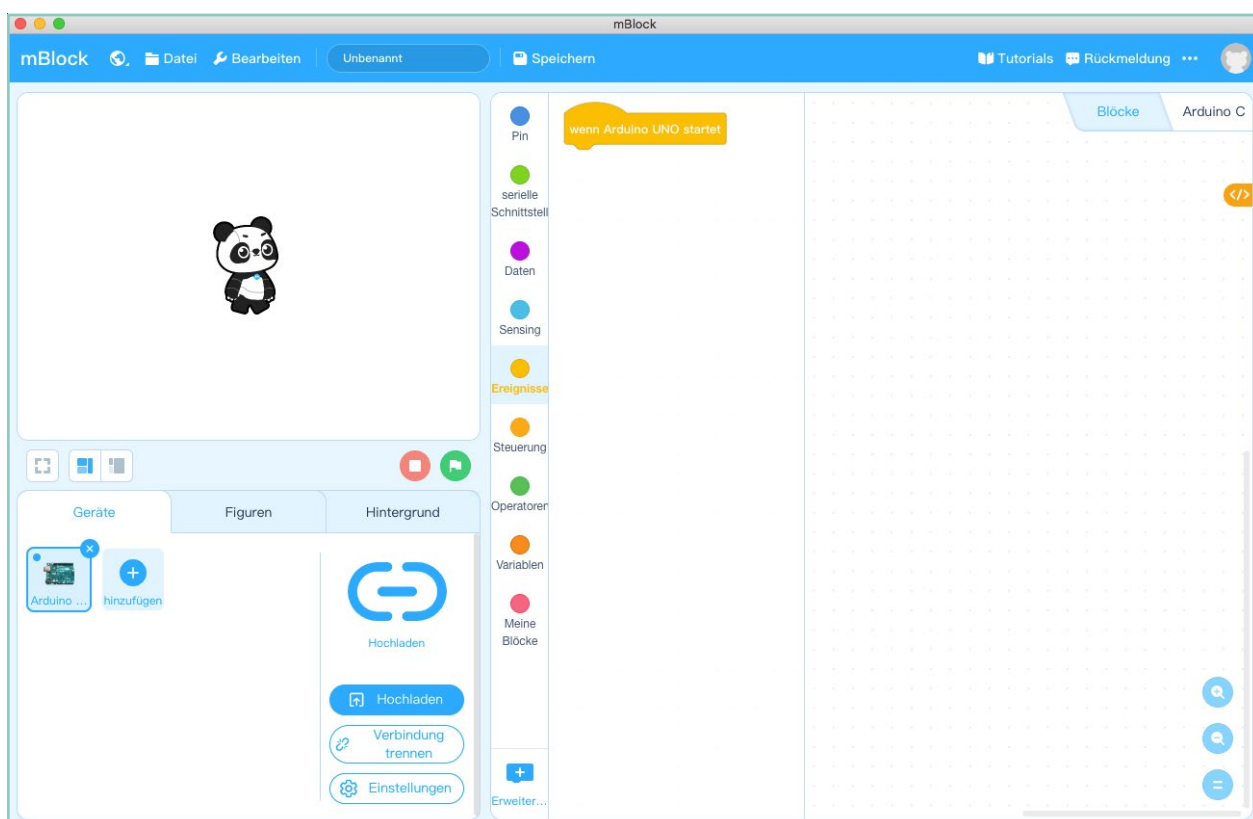


Wenn dein **Arduino** erkannt wurde, dann klicke jetzt auf **Verbindung**. Danach sollte das **Geräte-Fenster** so aussehen wie auf dem rechten Bild. Jetzt ist alles bereit, um deinen ersten Programmcode auf den **Arduino** zu laden!

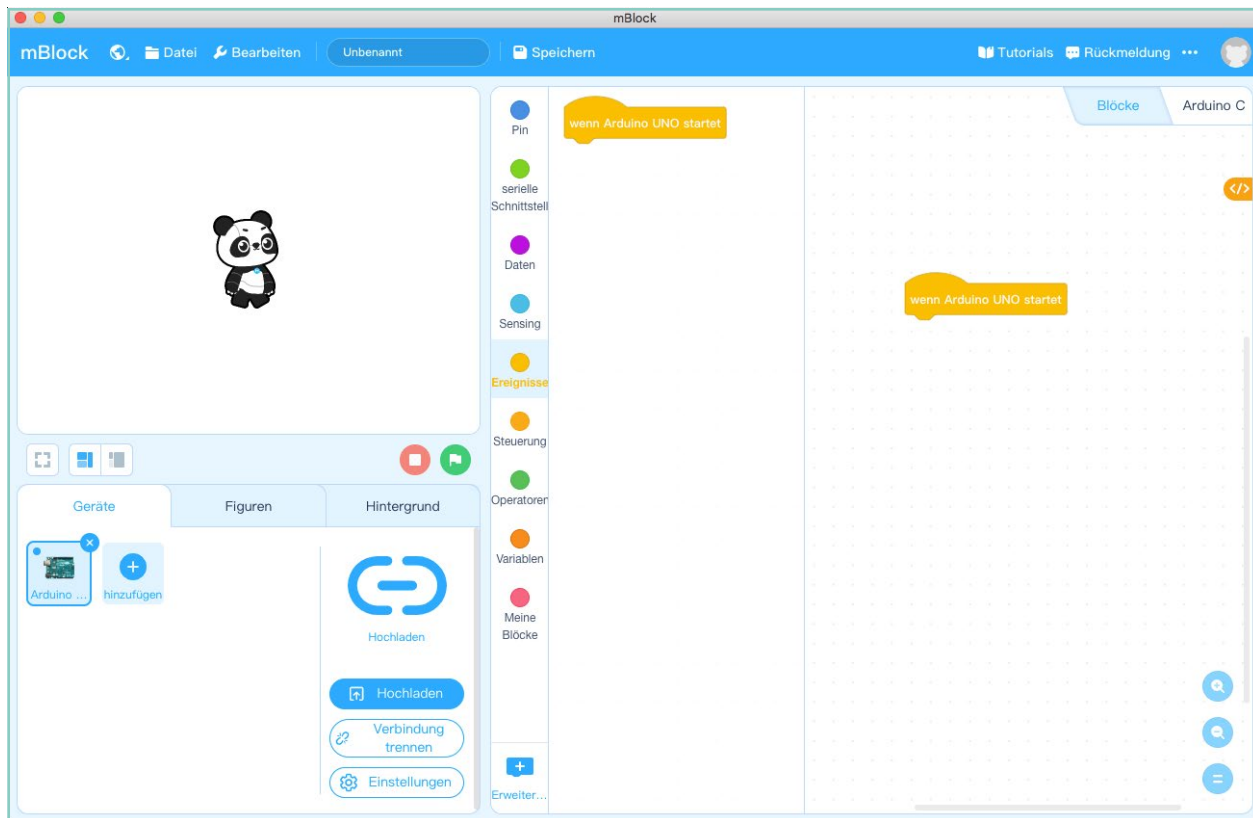


Rechts vom **Geräte-Fenster** und der **Bühne** – der Kasten mit dem Panda, den wir für das **Arduino**-Coding nicht brauchen, findest du einen Baukasten voller verschiedener Befehle. Die Befehle sind nach Themen und Funktionen gruppiert, jede Gruppe hat eine andere Farbe.

Jedes Programm für den **Arduino** beginnt mit dem Block **wenn Arduino UNO startet** aus der Gruppe **Ereignisse**. Klicke zunächst die Gruppe an – sie enthält keine anderen Blöcke.



Klicke den Block mit dem Mauszeiger an und bewege ihn mit gedrückter Maustaste auf die gepunktete Fläche rechts. Das ist die Arbeitsfläche, auf der du deinen Programmcode zusammenbaust.



Alle Blöcke, die miteinander verbunden werden können, lassen sich wie magnetische Puzzlesteine zu Ketten zusammensetzen oder verschachteln. Wenn du einen Block anklickst und bewegst, nimmt er alle Blöcke unter sich mit – so kannst du dein Programm auftrennen, wenn du noch irgendwo etwas einfügen oder verändern möchtest. Ziehst du Blöcke zurück in den Baukasten, werden sie von der Arbeitsfläche entfernt. In weiße Felder kannst du etwas eintippen – meist Zahlen.

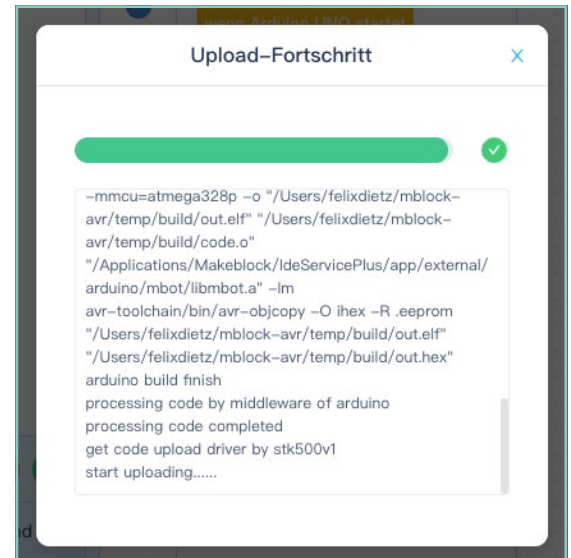


Mit diesem Beispiel-Code lässt du die eingebaute LED auf dem **Arduino** blinken. Sie ist intern verbunden mit **Pin 13**. Wenn das Programm ihn auf **hoch** setzt, wird er mit Strom versorgt und die LED leuchtet mit auf. Nach einer Wartezeit von 0.2 Sekunden setzt das Programm den **Pin 13** wieder auf **niedrig**, schaltet also den Strom ab und wartet dann wieder 0.2 Sekunden. Weil diese kleine Befehlskette in die Schleife **wiederhole fortlaufend** eingeklammert ist, geht die LED immer wieder an und aus – sie blinkt!

Aufgepasst: Komma-Zahlen müssen mit einem Punkt geschrieben werden, statt einem Komma ;)



Jetzt ist es endlich soweit! Klicke im **Geräte-Fenster** auf **Hochladen**, um deinen Programmcode auf den **Arduino** zu laden. Ein Popup öffnet sich und zeigt dir den Fortschritt des Uploads an.




Glückwunsch! Dein **Arduino** blinkt jetzt – oder tut, was du sonst so programmiert hast. Falls das nicht geklappt hat, liegt das vielleicht an Problemen, die häufiger auftauchen:

### Beim **Verbinden** wird mir kein Gerät angezeigt :(

Hast du das Häkchen bei **Alle anschlussfähigen Geräte anzeigen** gesetzt? Wenn immer noch nichts zu sehen ist, fehlen dir vielleicht die Arduino-Treiber. Auf dieser Seite findest du unter **Download the Arduino IDE** die richtige Version für dein Betriebssystem: [www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)  
Installiere das heruntergeladene Programm, starte deinen PC neu und probiere es nochmal.

### Es geht immer noch nicht!

Eventuell hast du einen günstigen **Arduino**-Klon der einen anderen Treiber benötigt. Hier findest du diesen Treiber für verschiedene Betriebssysteme. Ja, die Seite ist auf chinesisches aber du musst nur den Downloadbutton finden:  下载

Windows: [http://www.wch.cn/download/CH341SER\\_ZIP.html](http://www.wch.cn/download/CH341SER_ZIP.html)

Mac: [http://www.wch.cn/download/CH341SER\\_MAC\\_ZIP.html](http://www.wch.cn/download/CH341SER_MAC_ZIP.html)

Linux: <https://github.com/juliagoda/CH341SER>

In der Zip-Datei findest du eine setup.exe bzw. eine .pgk Datei, die den Treiber installiert.

### Ich habe noch ein anderes Problem!

Schicke mir – Felix – eine Mail an [dietz@fjmk.de](mailto:dietz@fjmk.de) und ich werde das nachtragen.