

## Station 4 – Farbthermometer – Bonus 1

### Mehr als nur an und aus

Oben wurde ja schon angedeutet, dass man mit diesen drei Grundfarben alle sichtbaren Farben erzeugen kann. Allerdings reicht es dafür nicht aus, die LEDs einfach nur an- und auszuschalten. Zum Glück kann der Arduino mehr! Vielleicht ist euch schon aufgefallen, dass an einigen Pins des Arduino kleine Schlangen (~) abgebildet sind. Diese Pins werden auch PWM-Pins genannt. Auch die Pins 9-11, die ihr eben mit der RGB-LED verbunden habt, sind mit solchen ~ markiert. Diese können 256 verschiedene Werte annehmen!

Die passende Anweisung lautet:

```
analogWrite(<Pin-Name>, <Wert>);
```





Der <Pin-Name> ist wie im Einstiegsprojekt der Name der Variablen, die die Nummer eures entsprechenden Pins enthält. <Wert> ist eine Zahl von 0 bis 255 oder der Name einer Variablen, die eine solche Zahl enthält.

Wie lautet die Anweisung um die blaue LED komplett einzuschalten?

```
analogWrite(_____, _____);
```

#### Eure ToDo's:

- Nehmt jetzt euren bisherigen Sketch und speichert ihn nochmal unter einem neuen Namen.
- Ersetzt den Befehl `digitalWrite()` durch `analogWrite()` und mischt weitere Farben! Notiert unten die Werte, für die eure LED den dargestellten Farben am nächsten kommt!

	Rot	Grün	Blau
	255		0
	146		80
		50	160
	255	255	

Jetzt ihr wisst ihr wie die Farbmischung mit einer RGB-LED funktioniert und könnt wirklich alle Farben erzeugen. Nun könnt ihr euer Farbthermometer beliebig anpassen. Viel Spaß dabei..

Temperatur	Farbe	R(ot)	G(rün)	B(lau)

*Hinweis: Hier reicht es jetzt nicht mehr nur einzutragen, ob die entsprechende LED an- oder ausgeschaltet ist, sondern auch der entsprechende Wert ist wichtig.*

Wer immer noch nicht genug hat und sich noch größeren Herausforderungen stellen möchte, kann die Betreuer auch noch nach dem **zweiten Bonus-Blatt** fragen.