

These slides serve as a visual aid for the lecture, not as a comprehensive document or script.

Please refrain from printing these slides to help protect the environment.

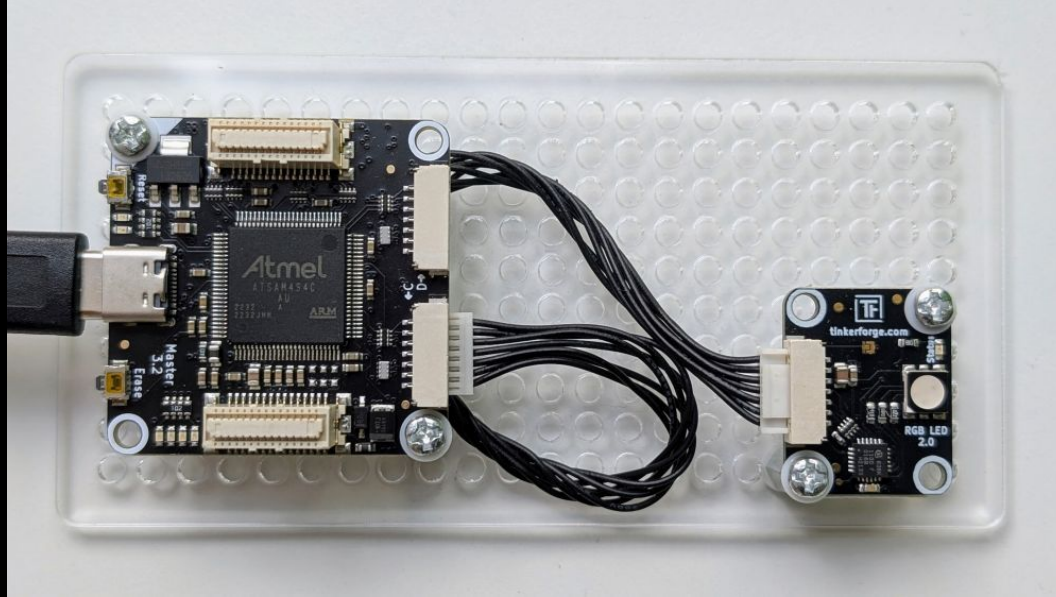
For any comments or feedback, please contact n.meseth@hs-osnabrueck.de.

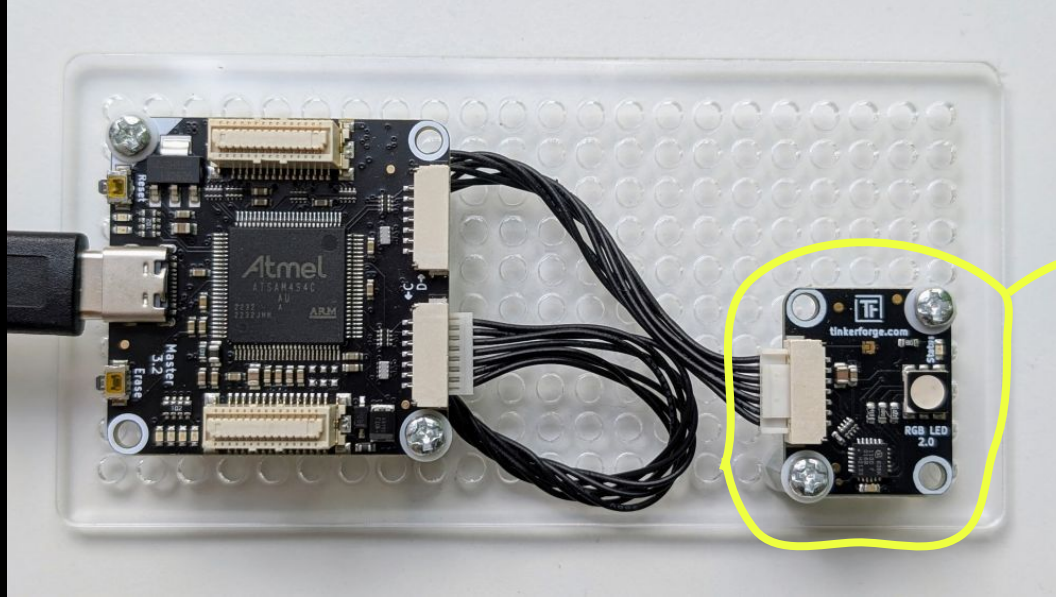


COLORS

Supporting slides for chapter 1 of the book
Hands-On Computer Science

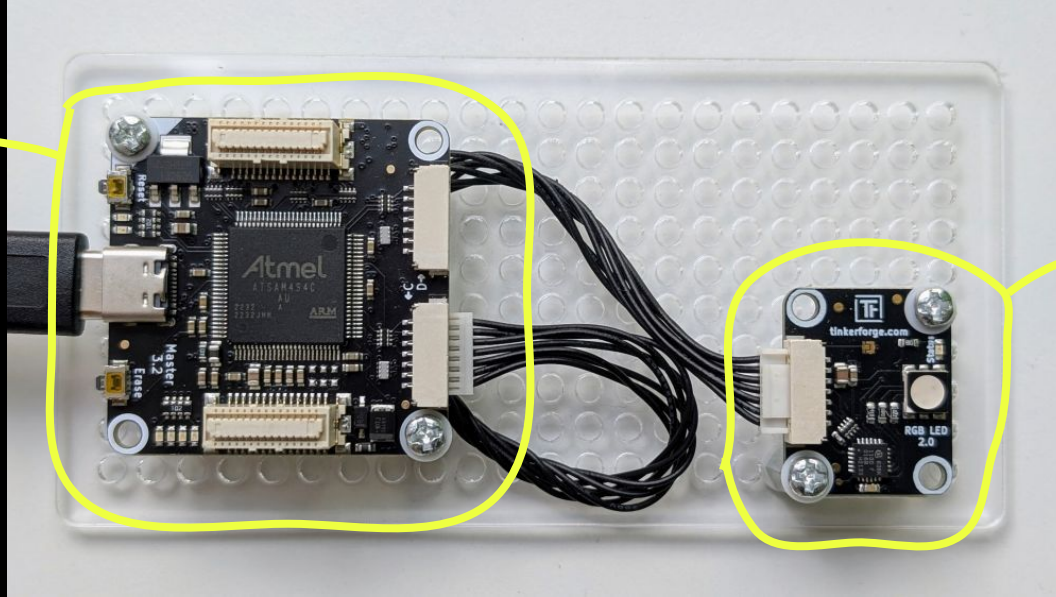
EXPERIMENT SETUP



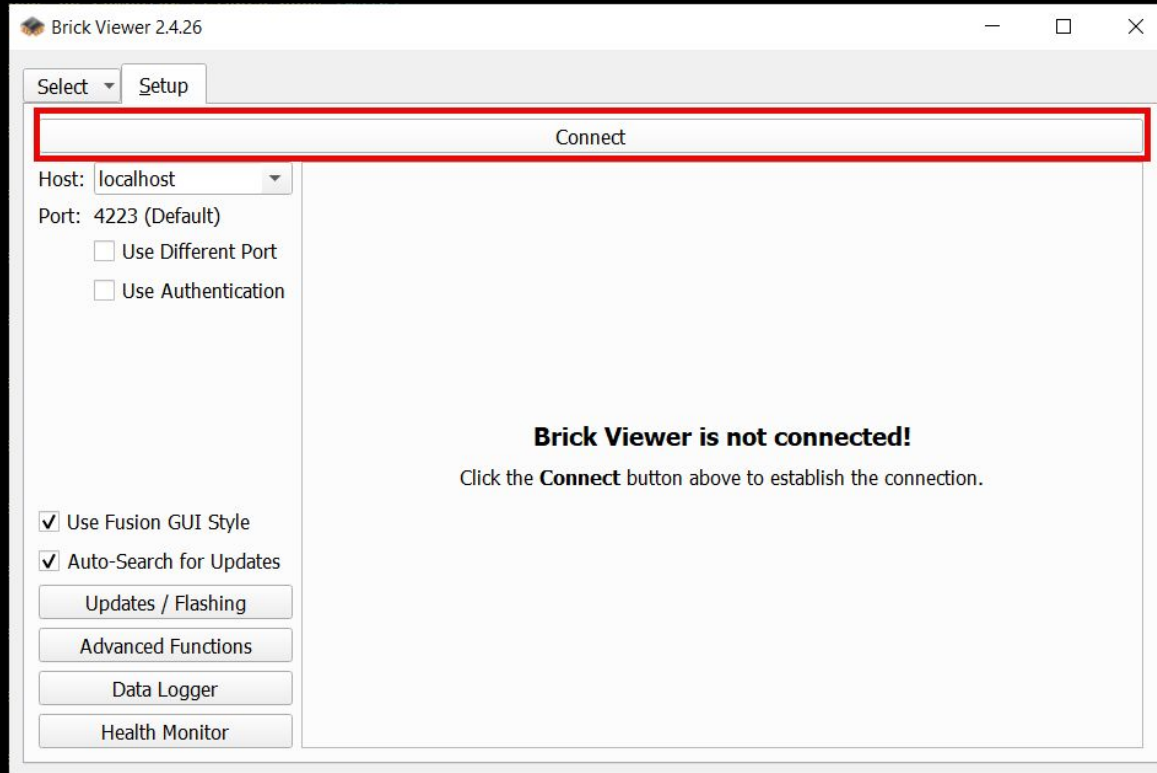


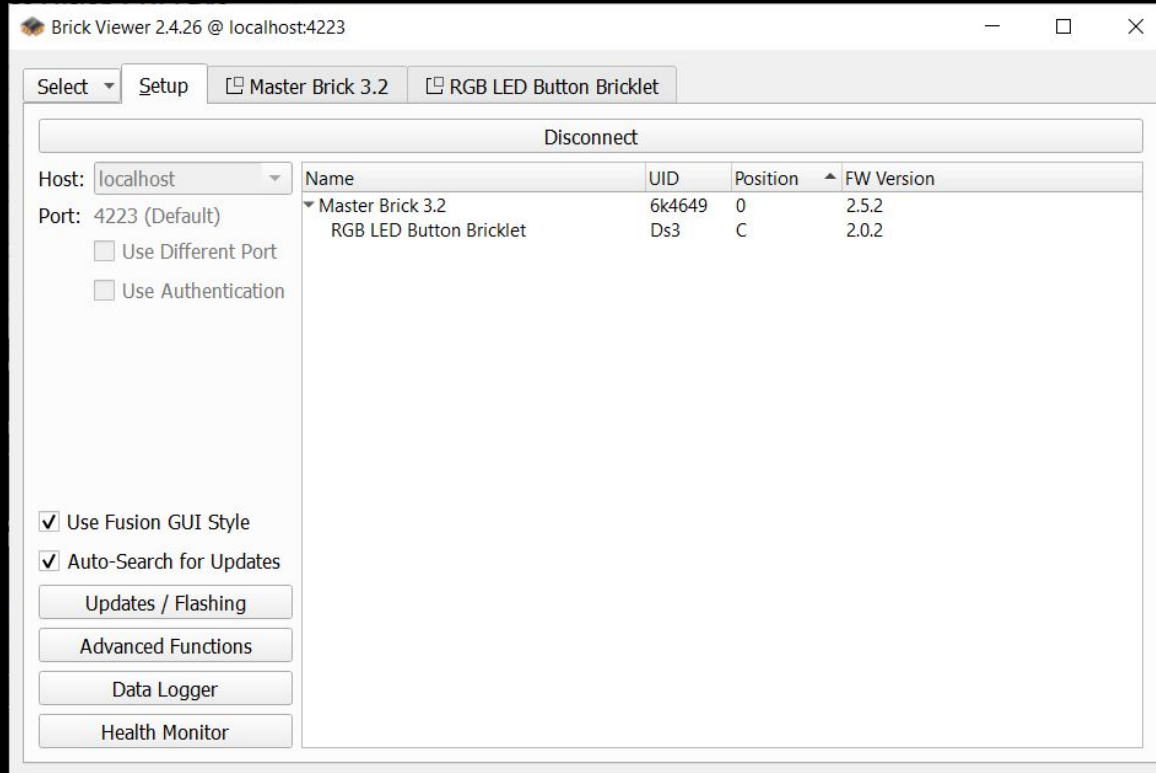
RGB LED

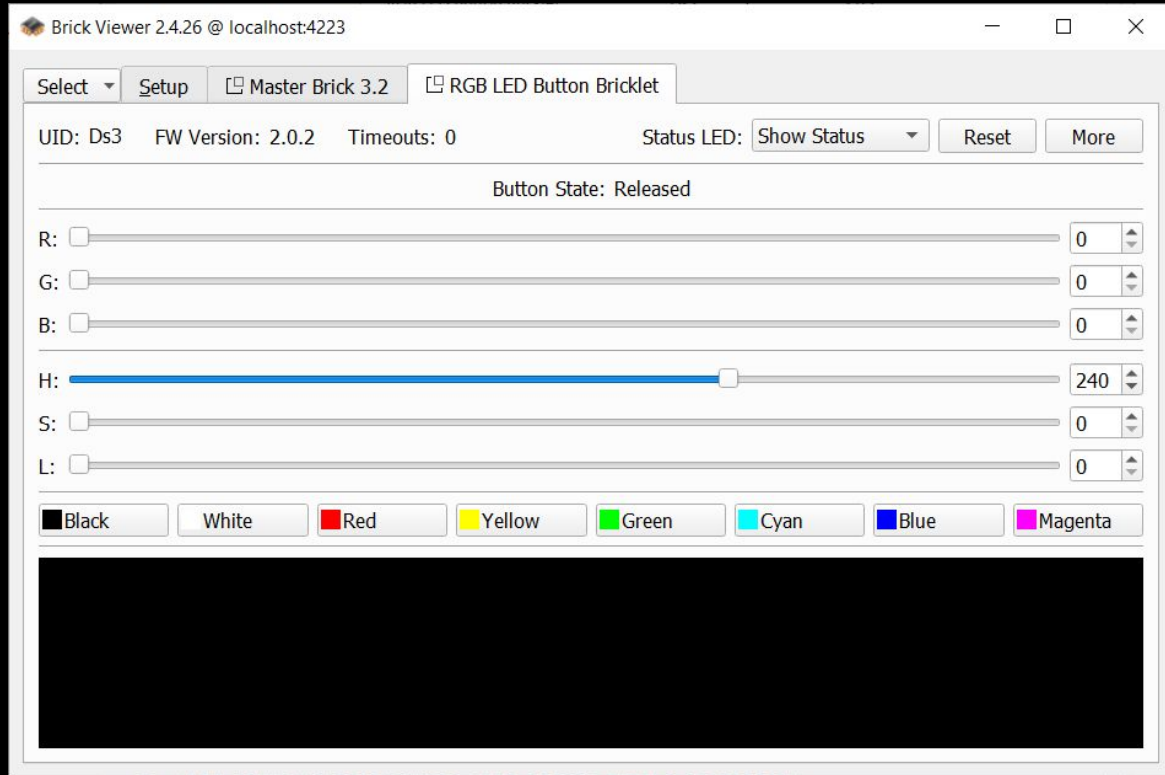
*Master Brick
(Microcontroller)*



RGB LED







FIRST PROGRAM

boilerplate code

```
from tinkerforge.ip_connection import IPConnection
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2

ipcon = IPConnection()
ipcon.connect("localhost", 4223)
led = BrickletRGBLEDV2("ZEP", ipcon)
led.set_rgb_value(0, 255, 0)
```

libraries or modules

```
from tinkerforge.ip_connection import IPConnection  
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2
```

```
ipcon = IPConnection()  
ipcon.connect("localhost", 4223)  
led = BrickletRGBLEDV2("ZEP", ipcon)  
led.set_rgb_value(0, 255, 0)
```


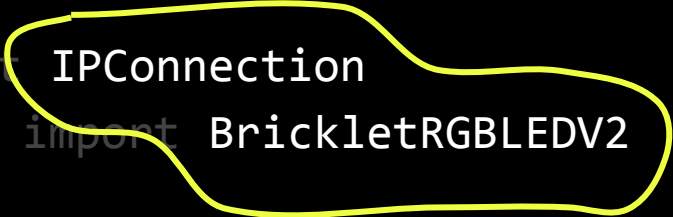
classes and objects

```
from tinkerforge.ip_connection import IPConnection
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2

ipcon = IPConnection()
ipcon.connect("localhost", 4223)
led = BrickletRGBLEDV2("ZEP", ipcon)
led.set_rgb_value(0, 255, 0)
```

classes and objects

```
from tinkerforge.ip_connection import IPConnection  
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2
```

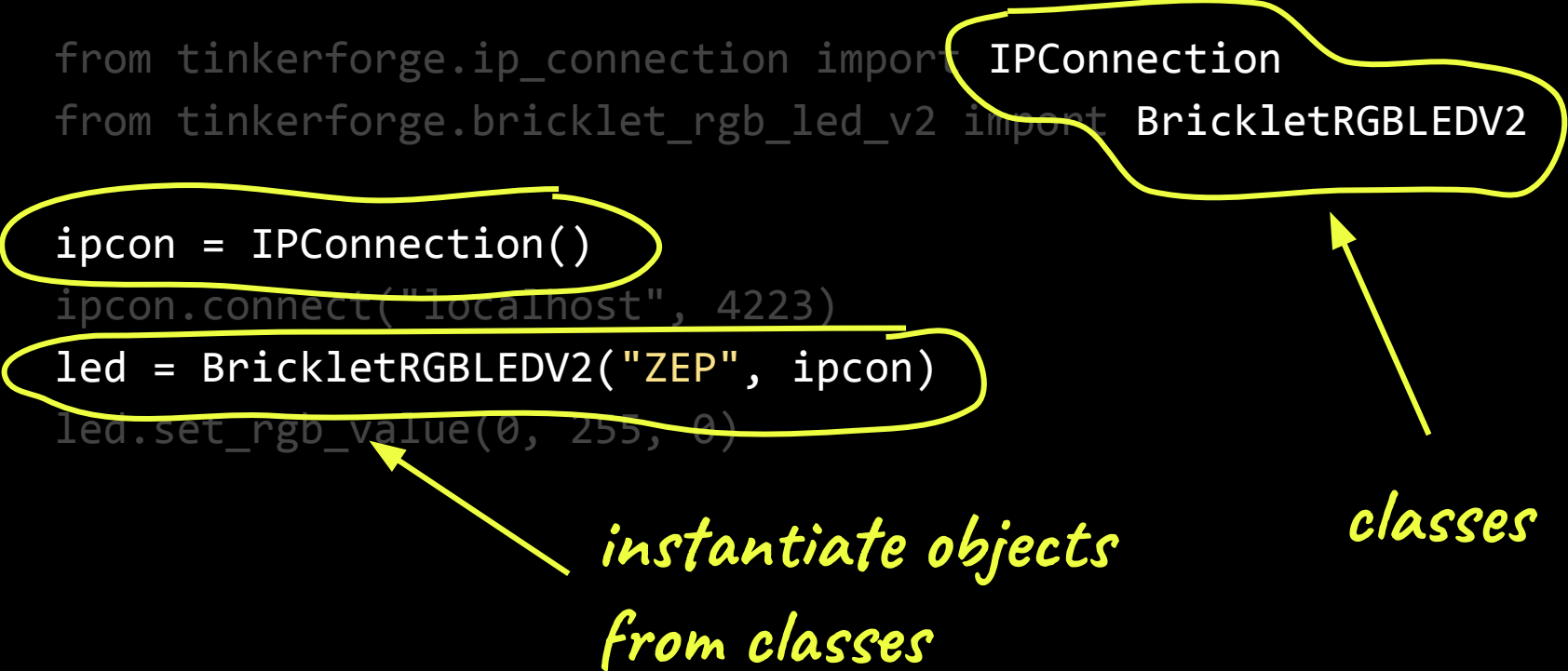


```
ipcon = IPConnection()  
ipcon.connect("localhost", 4223)  
led = BrickletRGBLEDV2("ZEP", ipcon)  
led.set_rgb_value(0, 255, 0)
```

classes

classes and objects

```
from tinkerforge.ip_connection import IPConnection  
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2
```



```
ipcon = IPConnection()
```

```
ipcon.connect("localhost", 4223)
```

```
led = BrickletRGBLEDV2("ZEP", ipcon)
```

```
led.set_rgb_value(0, 255, 0)
```

*instantiate objects
from classes*

classes

reserved keywords

```
from tinkerforge.ip_connection import IPConnection
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2

ipcon = IPConnection()
ipcon.connect("localhost", 4223)
led = BrickletRGBLEDV2("ZEP", ipcon)
led.set_rgb_value(0, 255, 0)
```


methods

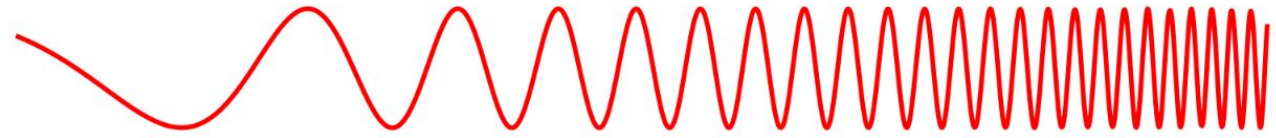
```
from tinkerforge.ip_connection import IPConnection
from tinkerforge.bricklet_rgb_led_v2 import BrickletRGBLEDV2

ipcon = IPConnection()
ipcon.connect("localhost", 4223)
led = BrickletRGBLEDV2("ZEP", ipcon)
led.set_rgb_value(0, 255, 0)
```

LIGHT AND COLORS

Durchdringt die
Erdatmosphäre?

Ja Nein Ja Nein



Strahlungstyp
Wellenlänge (m)

Radio
 10^3

Mikrowellen
 10^{-2}

Infrarot
 10^{-5}

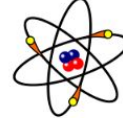
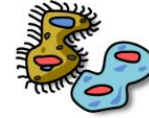
Sichtbar
 0.5×10^{-6}

Ultraviolett
 10^{-8}

**Röntgen-
strahlung**
 10^{-10}

**Gamma-
strahlung**
 10^{-12}

Größenordnung
der Wellenlänge



Gebäude

Mensch

Schmetterling

Nadelspitze

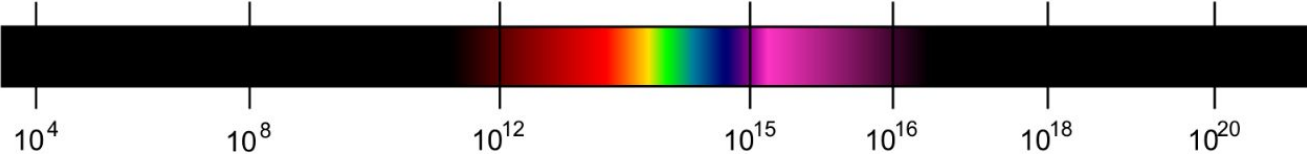
Einzeller

Molküle

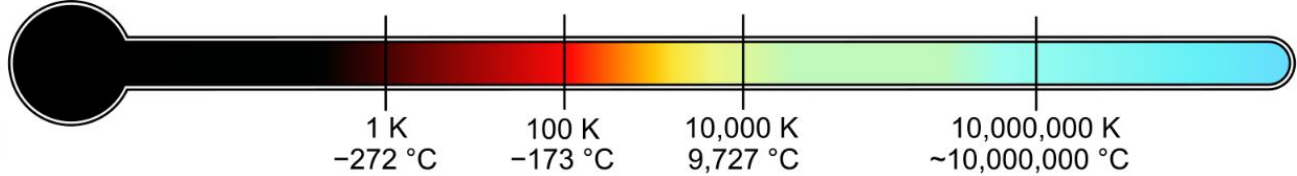
Atome

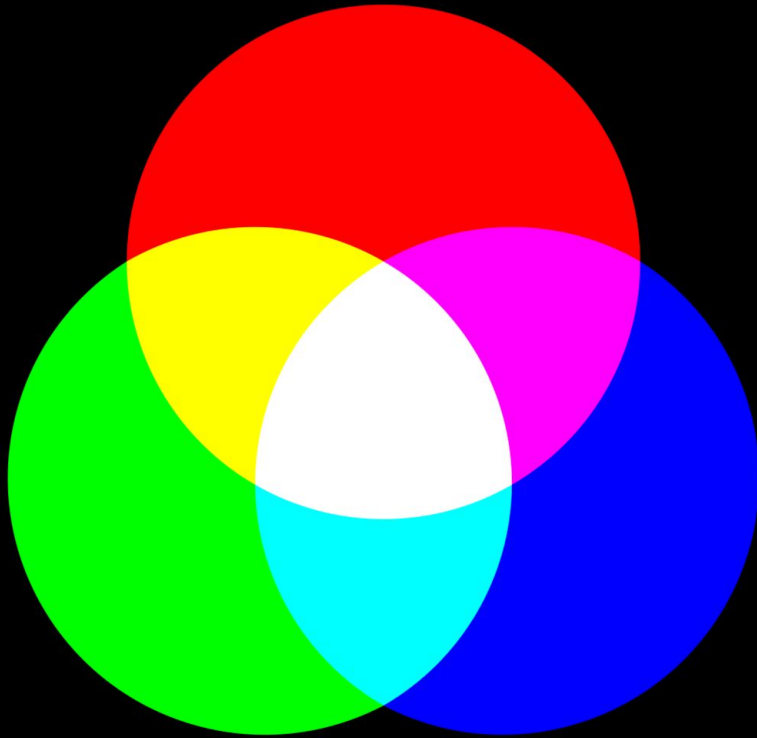
Atomkerne

Frequenz (Hz)

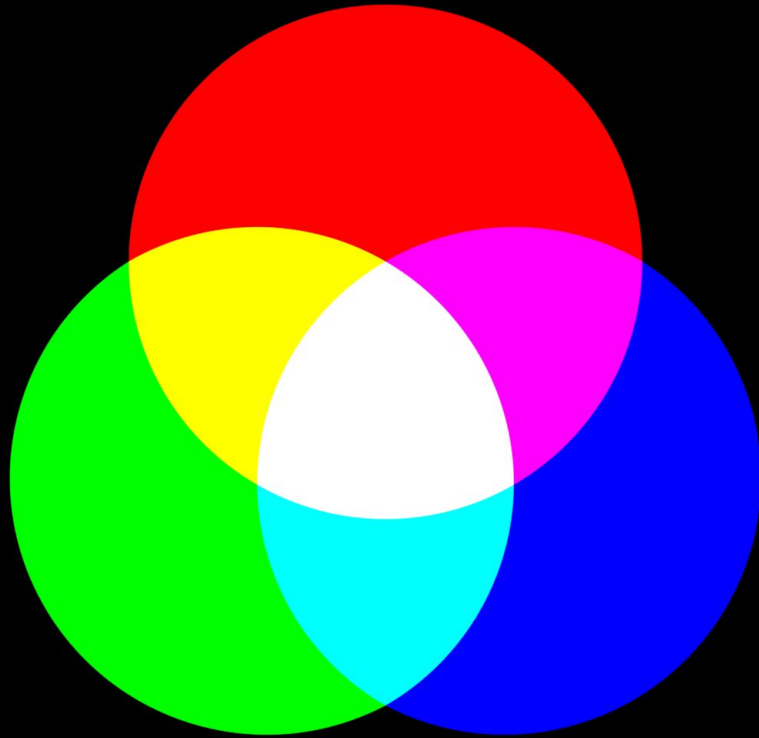


Temperatur von
Körpern, bei denen
diese Wellenlänge
am stärksten
abgestrahlt wird

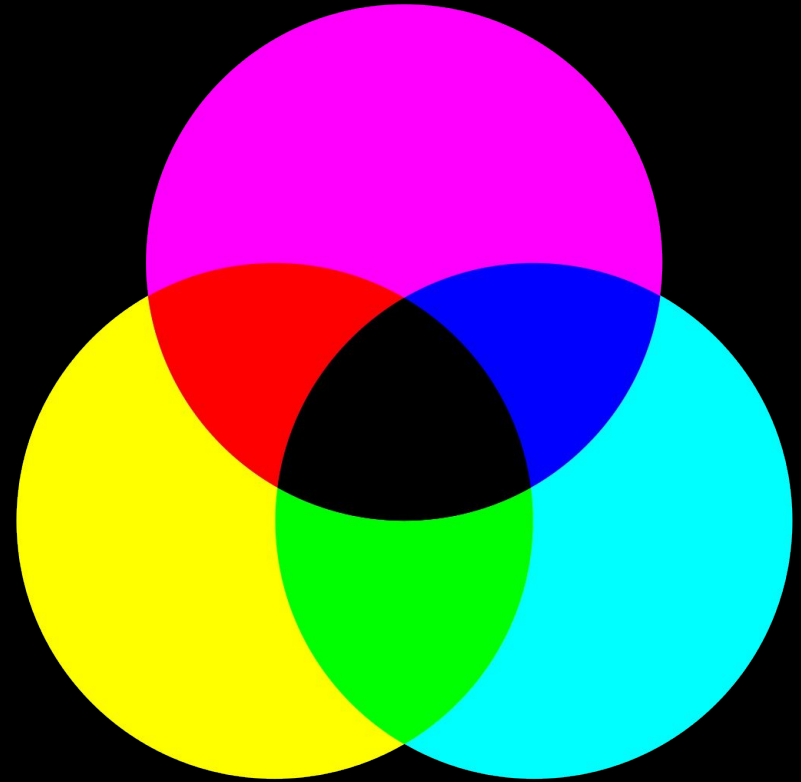




additive color mixing



additive color mixing



subtractive color mixing



PULSATING LED

FOR-LOOP

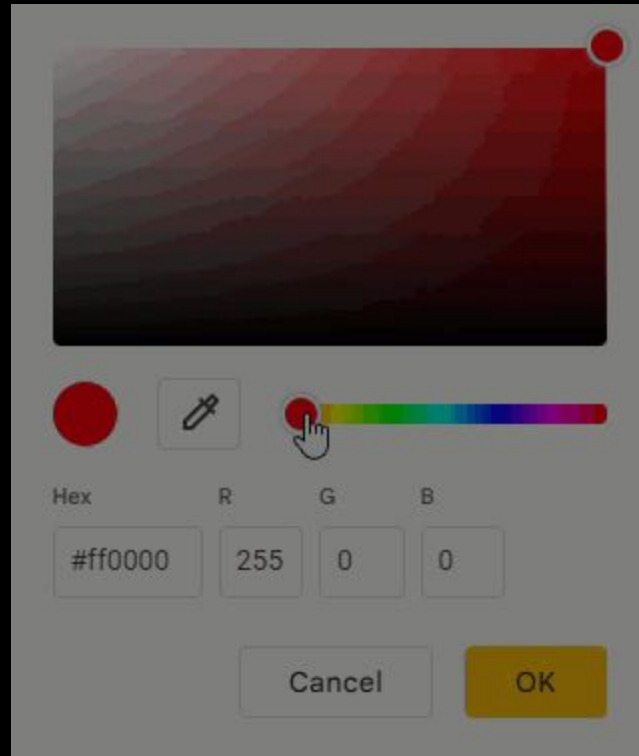

```
for r in range(256):  
    led.set_rgb_value(r, 0, 0)
```

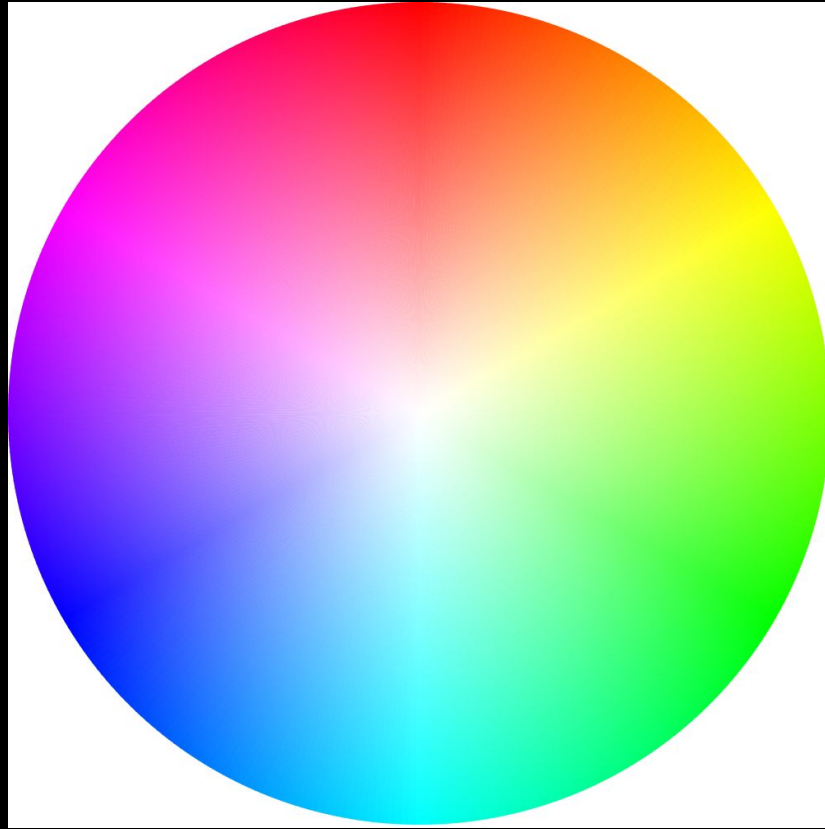
```
time.sleep(0.001)
```

WHILE-LOOP

```
while True:  
    print("I will loop forever")  
    time.sleep(1)
```

COLOR CIRCLES





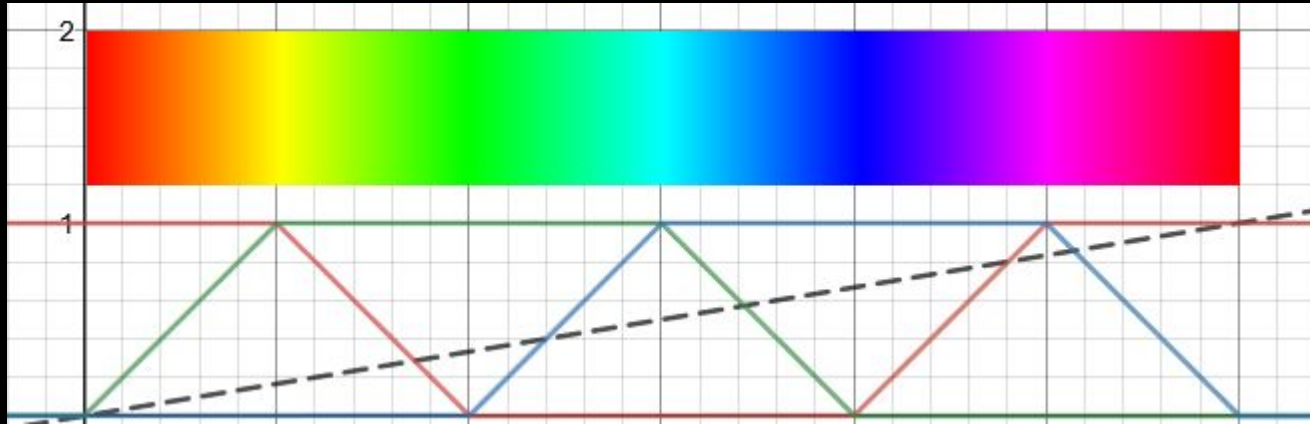
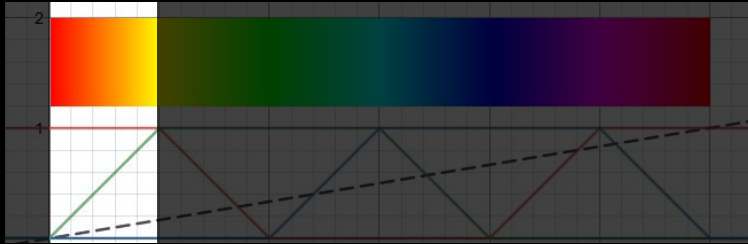
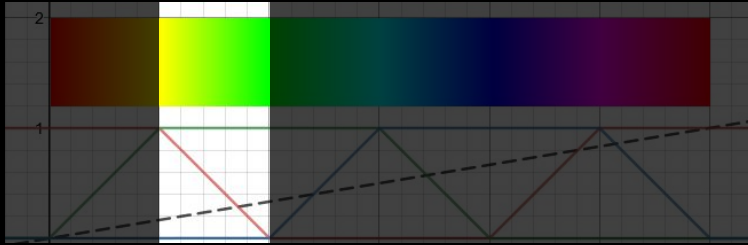


Image Source: <https://www.ronja-tutorials.com/post/041-hsv-colorspace/>

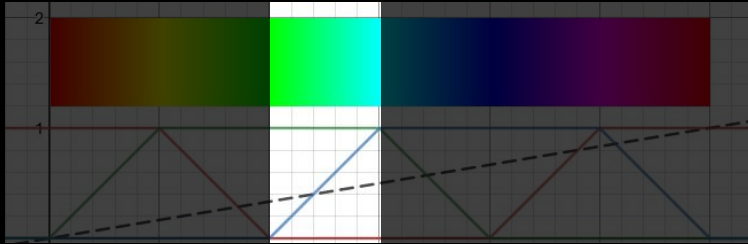
RAINBOW LED



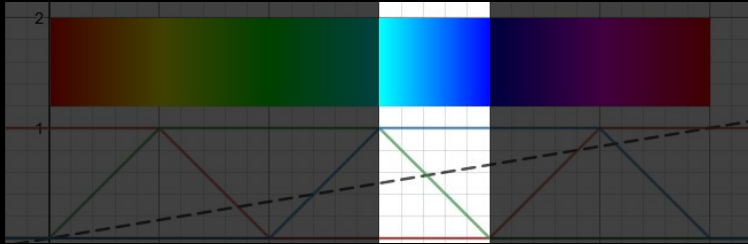
```
# phase 1: red = 255, blue = 0
for green in range(256):
    led.set_rgb_value(255, green, 0)
```



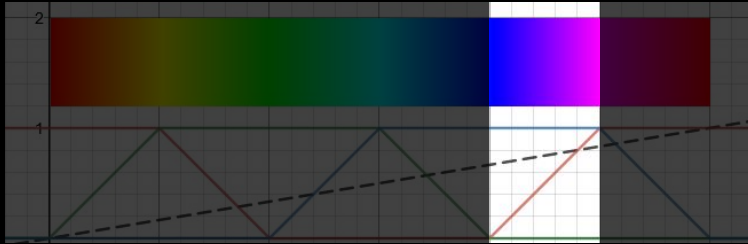
```
# phase 2: green = 255, blue = 0
for red in range(255, -1, -1):
    led.set_rgb_value(red, 255, 0)
```



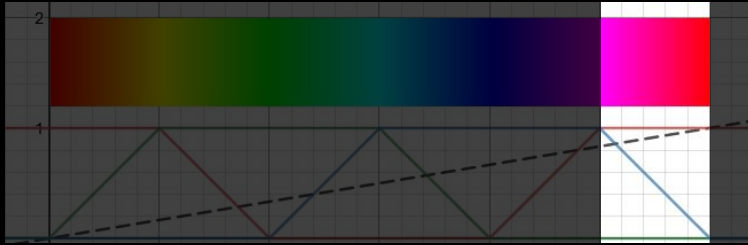
```
# phase 3: green = 255, red = 0
for blue in range(256):
    led.set_rgb_value(0, 255, blue)
```



```
# phase 4: blue = 255, red = 0
for green in range(255, -1, -1):
    led.set_rgb_value(0, green, 255)
```



```
# phase 5: blue = 255, green = 0
for red in range(256):
    led.set_rgb_value(red, 0, 255)
```



```
# phase 6: red = 255, green = 0
for blue in range(255, -1, -1):
    led.set_rgb_value(255, 0, blue)
```

CONTROLLING SPEED