



Unit 3: 明るさ, TI-Innovator™ Hub でのifとwhile

Skill Builder 3: 明るさと色

このレッスンでは、明るさセンサを使ってカラーLEDを制御します。

目標

- **brightness.range()**を使って明るさのスケール変更
- 明るさの値を使ってカラーLEDを点灯
- 数値変換の調査

TI-Innovator Hubライト(赤色LED)とは異なり、カラーLEDの明るさはさまざまです。つぎに、明るさセンサを使ってそのLEDを制御します。



1. 明るさ測定プログラムの別のコピーを作成します(または前レッスンの明るさと光のプログラム...それらはほとんど同じです)。

カラーLEDの3つのカラーチャンネルは0~255の値しか取れないため、明るさの範囲を0~255に設定します。

brightness.range(0, 255)

これで、明るさの値**b**を3つのカラーチャンネルの値として使えるようになりました...おそらく。

```

1.1 1.2 1.3 ▶ *Unit 3 Pyt...ile RAD 12/21
* u3sb3b.py
from ti_system import get_key
#-----
cls()
text_at(13, "Press [esc] to end", "center")

brightness.range(0,255)

while get_key() != "esc":
    b=brightness.measurement()
    ..
    ..

```

2. **brightness.measurement()**ステートメントの後にステートメントを追加して、3つのチャンネルはすべて変数**b**を使いカラーLEDを点灯させます。

color.rgb(b,b,b)

プログラムを実行します。

```

1.4 1.5 1.6 ▶ *Unit 3 Pyt...ile RAD 13/25
* u3sb3b.py
from ti_system import get_key
#-----
cls()
text_at(13, "Press [esc] to end", "center")

brightness.range(0,255)

while get_key() != "esc":
    b=brightness.measurement()
    color.rgb(b,b,b)
    ..
    ..

```

3. 光のレベルが上がるとLEDが明るくなることに注意してください。これは逆です! 部屋が暗いほど、光は明るくなります。変数**b**を変更して、LEDが低い値の場合は明るく、大きな値の場合は暗くします。(brightness.measurement()とcolor.rgb()の間に式を置きます。右画面の**b=???**を参照してください。)

次のステップに進む前に、やってみましょう。

```

1.2 1.3 1.4 ▶ *Unit 3 Pyt...ile RAD 17/21
* u3sb3b.py
text_at(13, "Press [esc] to end", "center")

brightness.range(0,255)

while get_key() != "esc":
    b=brightness.measurement()
    ..
    b= ???
    color.rgb(b,b,b)
    text_at(7, "brightness = "+str(b), "left")

```



4. 答えの1つは、次のとおりです。

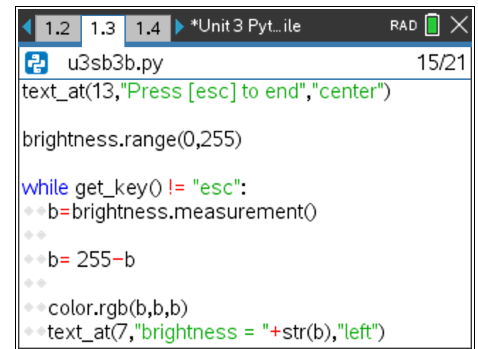
$$b=255 - b$$

bが0のとき、式**255-b**は**b**の値を255に変更します。**b**が255のとき、式**255-b**は**b**の値を0に変更します。

変換された値ではなく、**b**の元の値を表示するには、プログラム内で `text_at()`ステートメントを移動する必要があるときもあります。

プログラムを変更して、白以外の色を生成できますか。

プログラムが終了しても、カラーLEDが点灯したままになる場合があります。ループの最後(インデントなし)にステートメントを追加して、カラーLEDをオフにします。



```
1.2 1.3 1.4 *Unit 3 Pyt...ile RAD [X]
u3sb3b.py 15/21
text_at(13,"Press [esc] to end","center")
brightness.range(0,255)
while get_key() != "esc":
    b=brightness.measurement()
    b= 255-b
    color.rgb(b,b,b)
    text_at(7,"brightness = "+str(b),"left")
```