

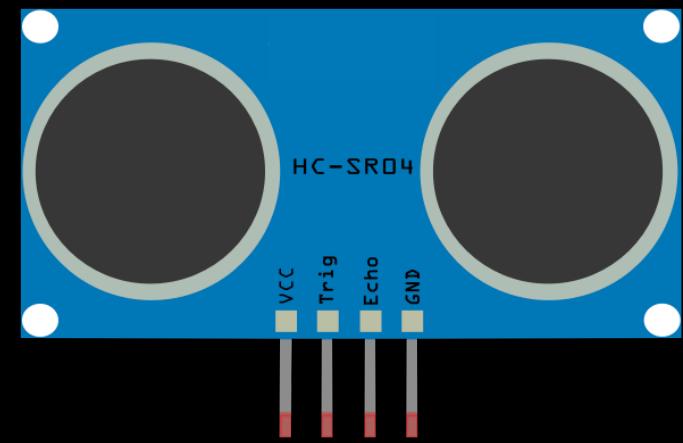
# Projeto 05

## Controle Automático – Prática

Jan K. S. – [janks@puc-rio.br](mailto:janks@puc-rio.br)

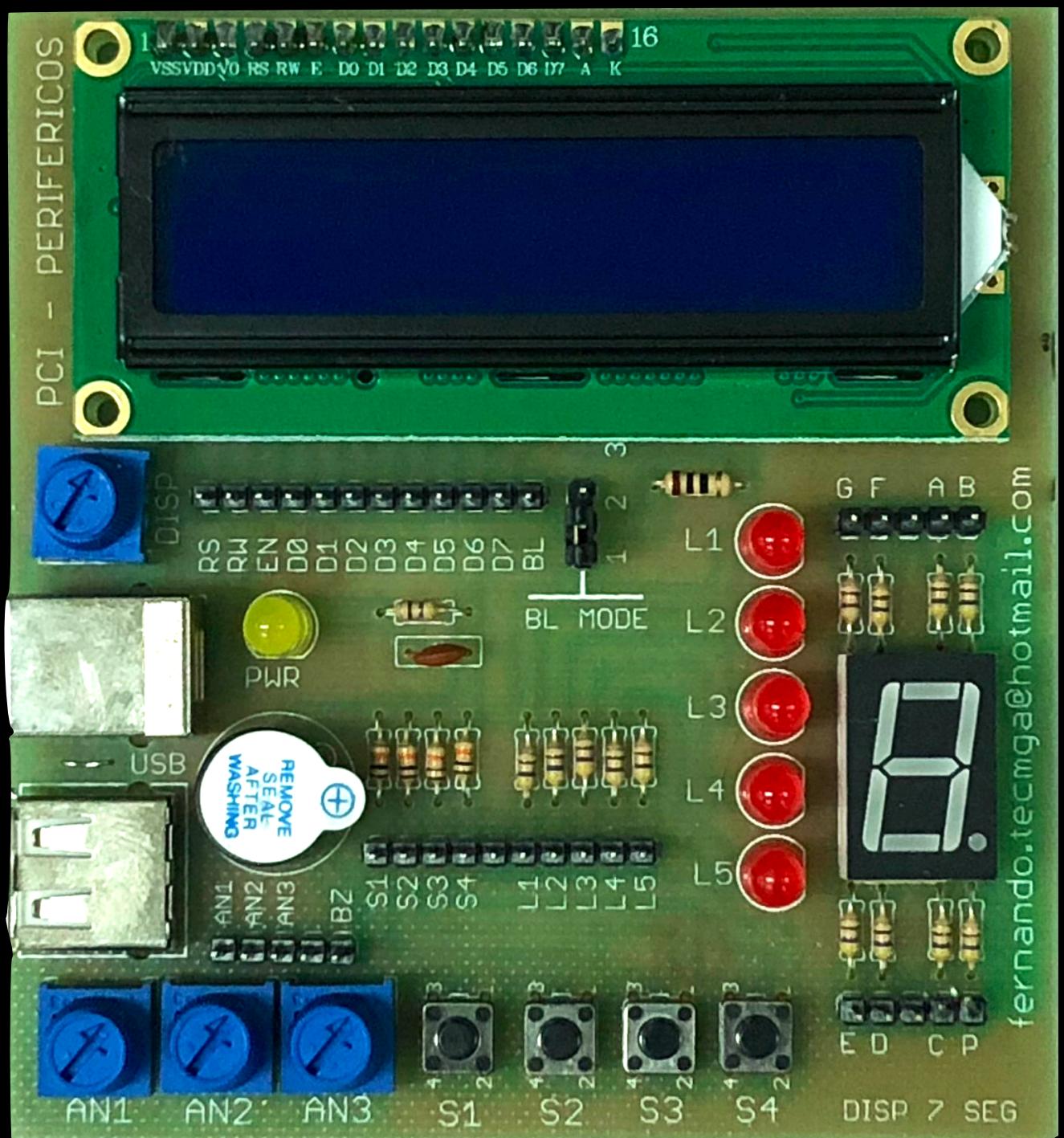
ENG1419 – Programação de Microcontroladores

# Testes Iniciais



GPIO 17, 18

GPIO 16



GPIO 11, 12, 13, 14

GPIO 21  
GPIO 22  
GPIO 23  
GPIO 24  
GPIO 25



GPIO 27

Conexões com as Portas da GPIO



## Testes Iniciais

Acenda o LED 1 ao **detectar um movimento**, e apague-o ao **detectar a inércia**.  
↪ DICA: use `when_motion` e `when_no_motion`.

Acenda o LED 2 ao detectar um movimento e apague-o **somente se não houver movimento por 8 segundos**.  
↪ DICA: crie um novo Timer único em caso de inércia e cancele-o se houver movimento.

**Crie um applet do IFTTT que, ao receber um evento tipo Webhook, coloque os 2 valores fornecidos em uma linha de um documento do Google Docs, no formato "Valor1 % de luz / Valor2 cm".**

Ao apertar o Botão 1, **chame o evento do applet acima**, enviando o valor do sensor de luz (em porcentagem) e o do sensor de distância (em cm).

...

```
print("Enviando linha para planilha...")
```

...

```
print("Linha enviada!")
```

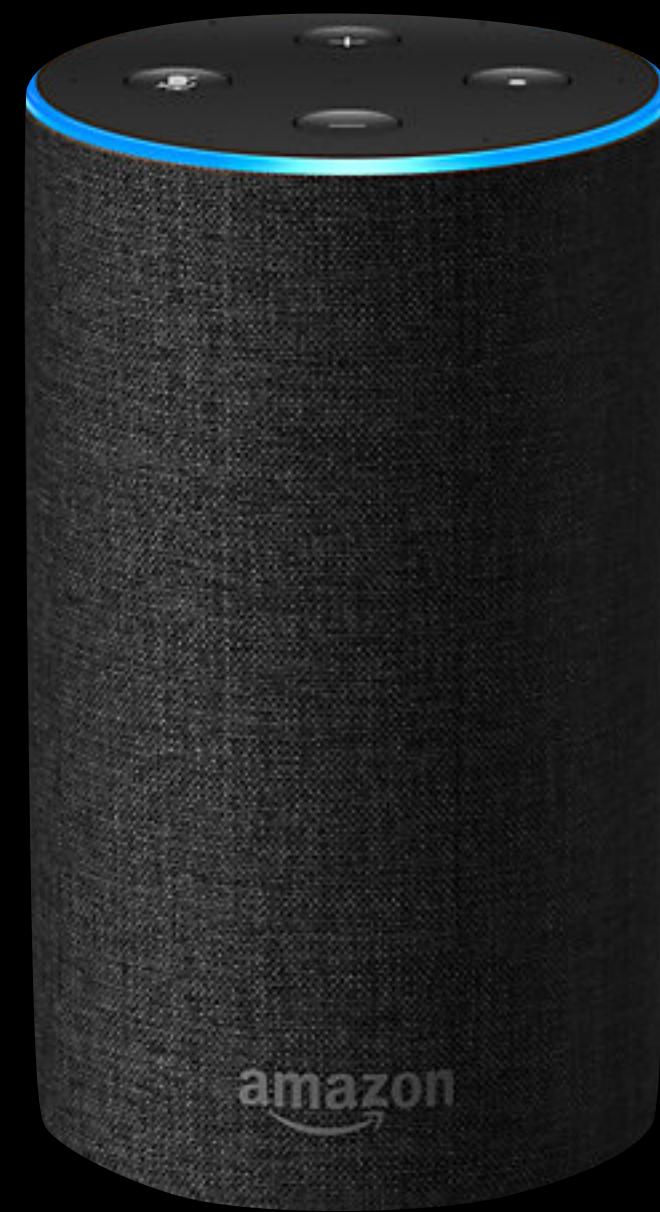
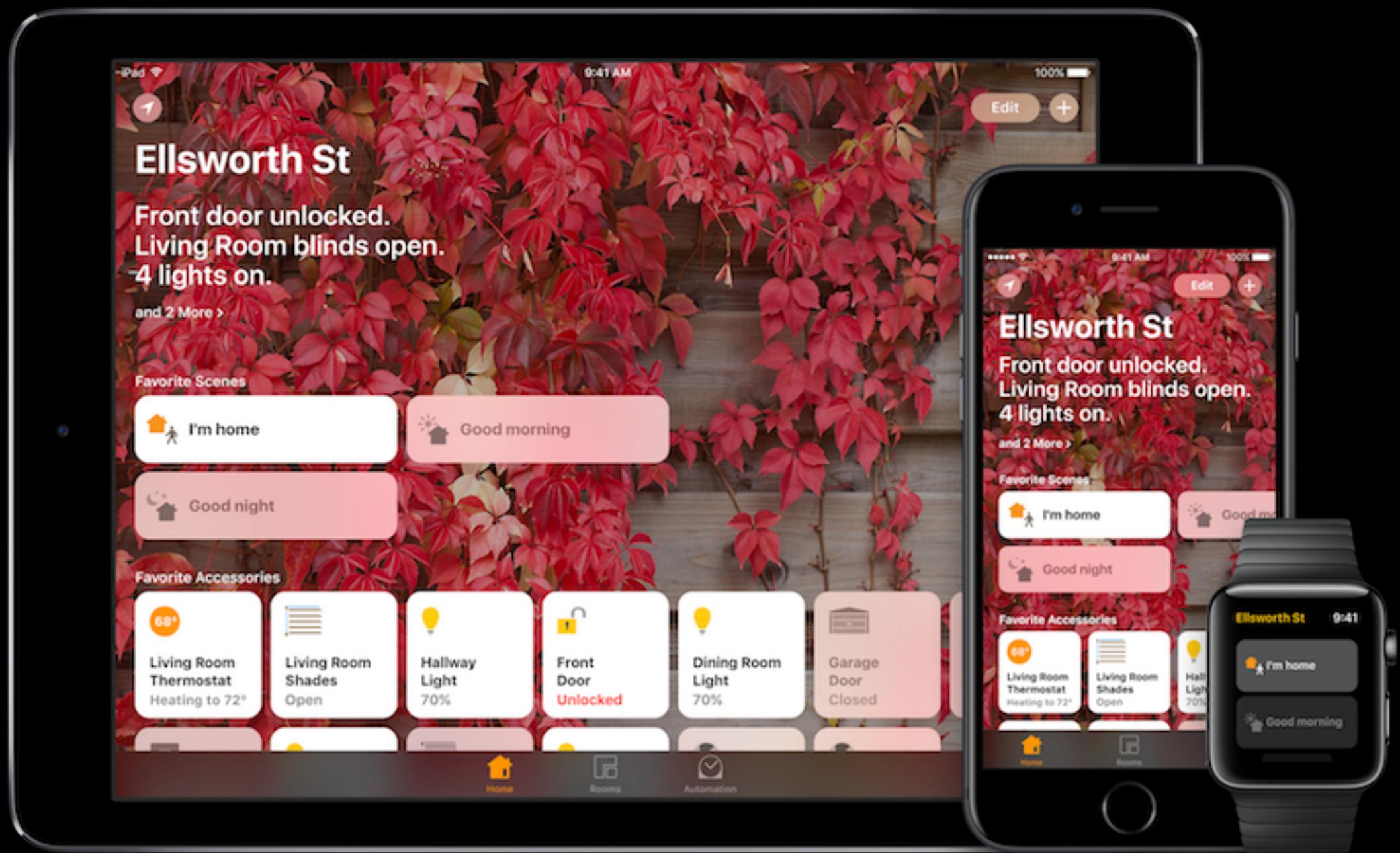
...

```
resposta = post(endereco, json=dados)
print(resposta.text)
```

...

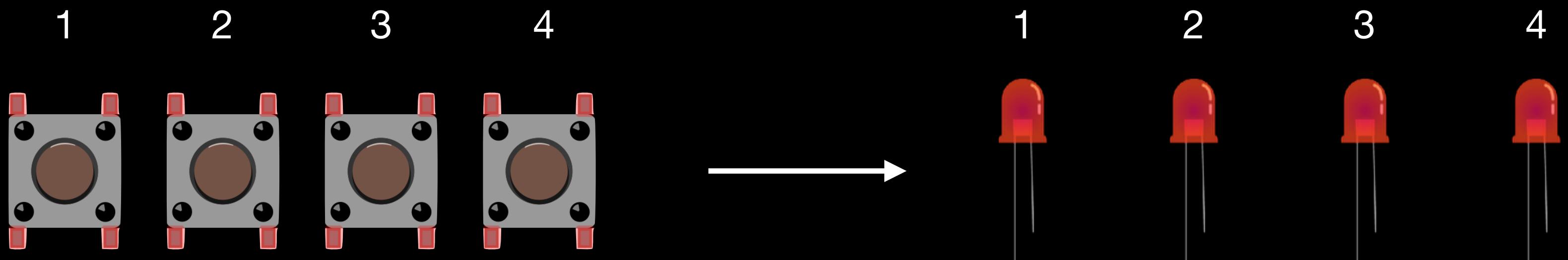
DICA: Acompanhamento das Etapas com Print

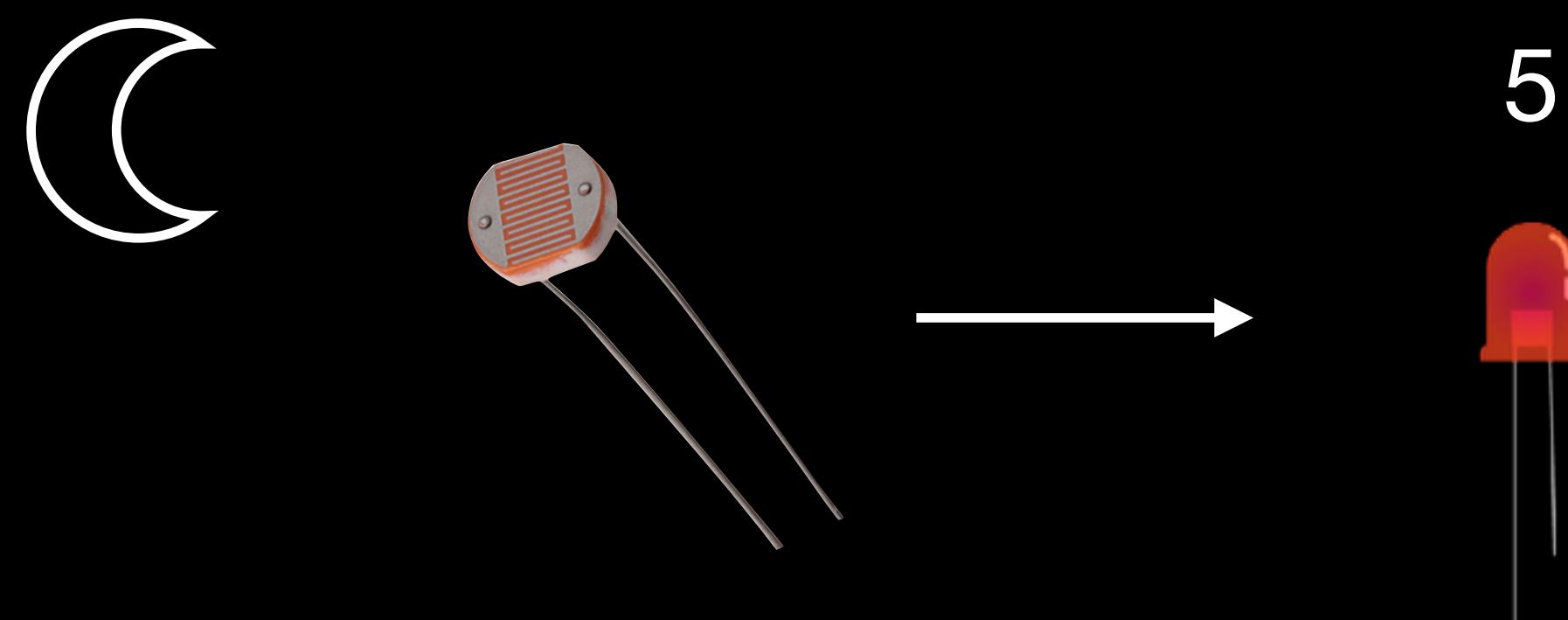
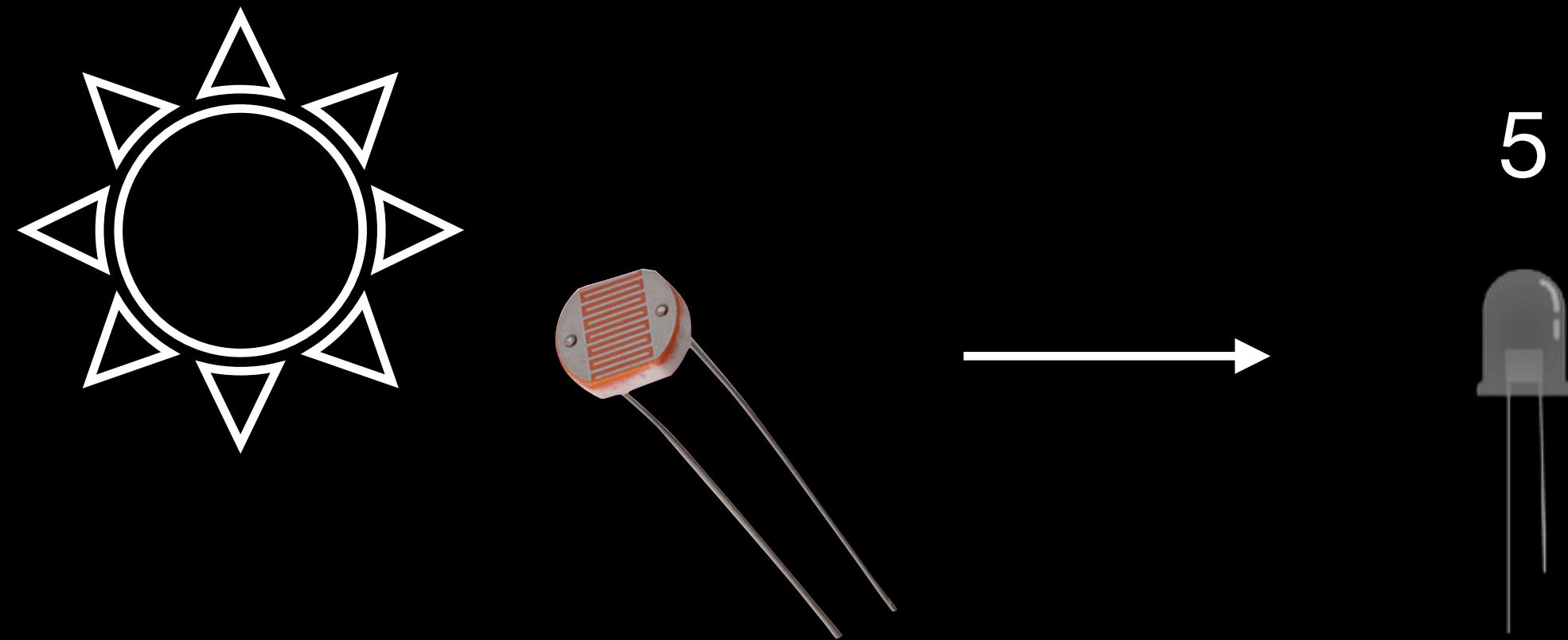
# Implementação



Smart Home

aperte para ligar, aperte de novo para desligar





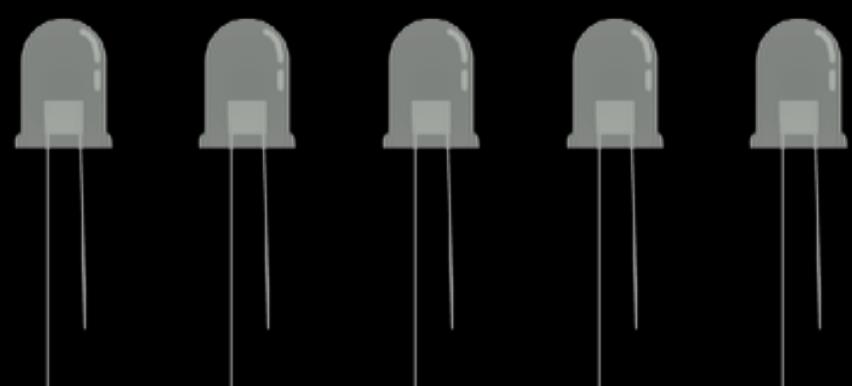
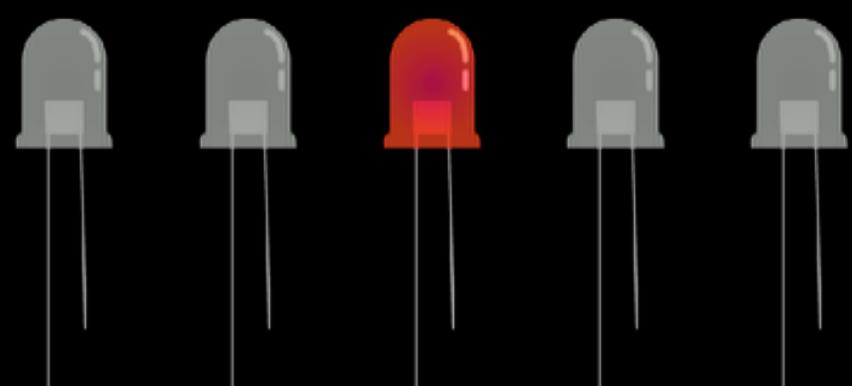
Controle do LED 5 de Acordo com a Iluminação

turn **on** light number 3

turn **off** light number 3

<http://.../luz/3/on>

<http://.../luz/3/off>





Ao apertar os Botões 1, 2, 3 e 4, alterne o estado (aceso/apagado) dos LEDs 1, 2, 3 e 4, respectivamente.

Controle o LED 5 de acordo com a luz captada: acenda-o caso esteja escuro, e apague-o caso esteja claro.

↪ DICA: use as propriedades `when_dark` / `when_light`.

Crie um servidor com uma página que receba o número do LED e o estado desejado (on/off), e altere o LED solicitado. Teste a página em `localhost:5000` e depois com o ngrok.

↪ DICA: use os parâmetros `int` e `string` na rota.

## Implementação

Integre o Google Assistant com o IFTTT para controlar as luzes por voz pelo app.

↪ DICA: verifique **atentamente** se o endereço do webhook está sem espaços em branco. E não esqueça de colocar o idioma do aplicativo em inglês.

# Aperfeiçoamento



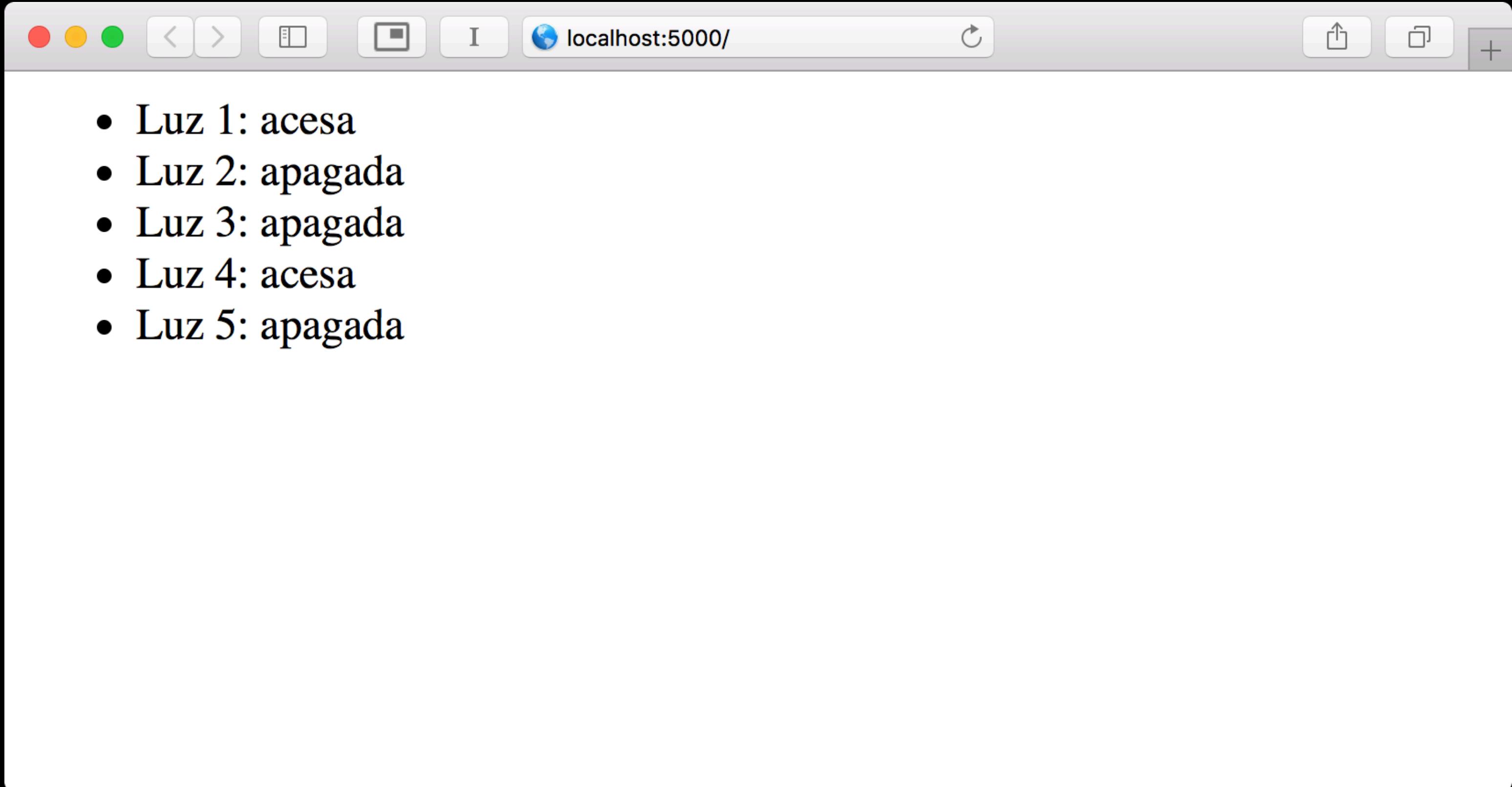
05b\_implementacao.py

cópia  
----->

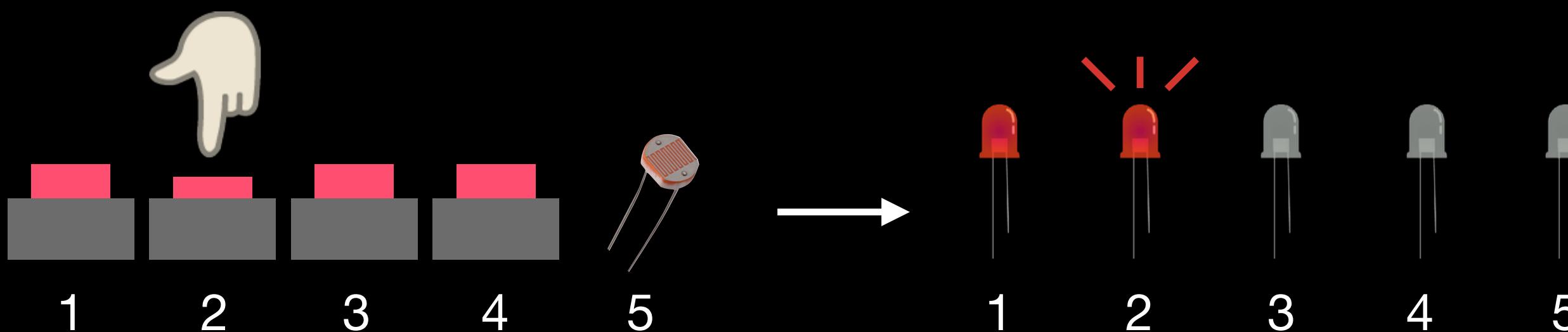


05c\_aperfeicoamento.py

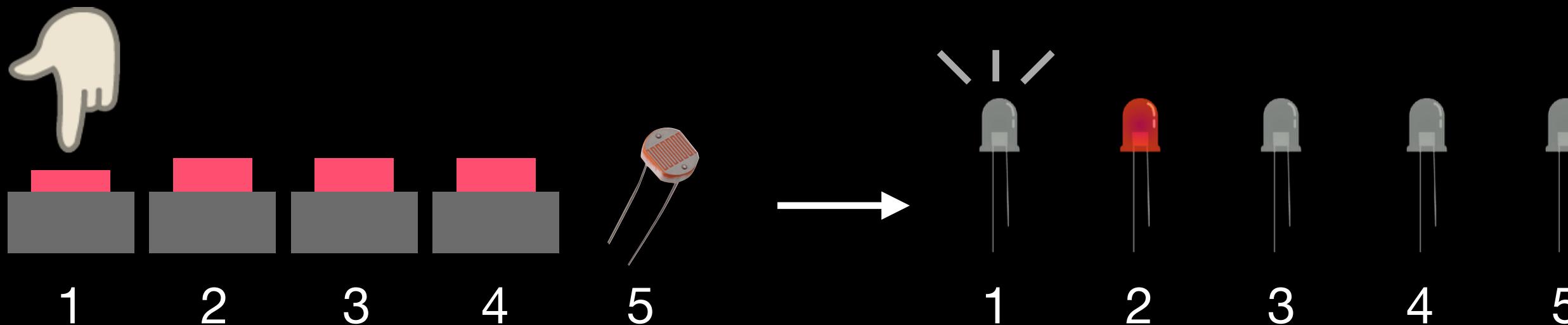
Cópia do Código da Implementação para o Aperfeiçoamento



Página com o Estado de Todos os LEDs



```
{  
    "data": datetime(2018, 8, 20, 15, 30, 45),  
    "estado_dos_leds": [True, True, False, False, False]  
}
```



```
{  
    "data": datetime(2018, 8, 20, 15, 52, 27),  
    "estado_dos_leds": [False, True, False, False, False]  
}
```

Estado das Luzes no Banco de Dados

Crie uma função que salve a data/horário atual + lista com o estado de cada LED em uma coleção do banco de dados. Teste a função no shell do Thonny e verifique o banco acessando localhost:1234 no navegador.



Modifique o código da Implementação para chamar a função acima cada vez que um LED for alterado. Verifique novamente os dados em localhost:1234 no navegador.  
↪ DICA: o código vai ficar meio repetido mesmo.

## Aperfeiçoamento

Crie uma página que busque os dados no banco e exiba o estado mais recente dos 5 LEDs.  
↪ DICA: faça uma busca com ordenação e gere o HTML dentro da função concatenando strings dentro de um for.

# Desafio Extra



05c\_aperfeicoamento.py

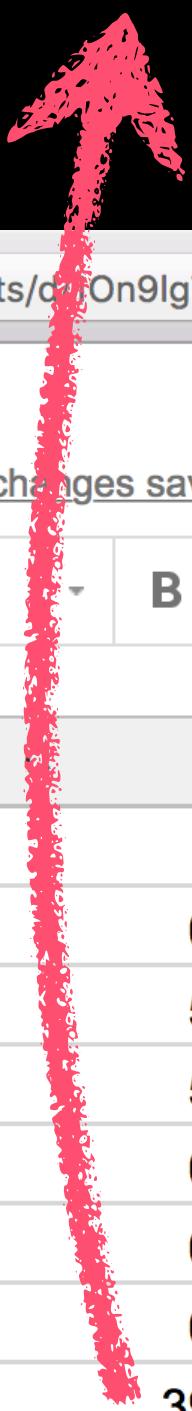
cópia  
----->



05d\_desafio.py

Cópia do Código do Aperfeiçoamento para o Desafio

entre 12:18:00 até 12:18:59,  
Luz 3 ficou acesa durante 39 segundos



A screenshot of a Google Sheets document titled "IFTTT\_Maker\_Webhooks\_Events". The document contains a table with data from September 17, 2018, at 12:12PM to 12:24PM. The columns are labeled A through F. Column A is "Data", B is "Luz 1", C is "Luz 2", D is "Luz 3", E is "Luz 4", and F is "Luz 5". The "Luz 3" column shows values: 0, 8, 8, 0, 0, 0, 0, 51, 1, 0, 29, 29, 0, 0, 0, 0, 0. The value 39 is highlighted with a red arrow pointing from the explanatory text above.

	A	B	C	D	E	F
5	Data	Luz 1	Luz 2	Luz 3	Luz 4	Luz 5
6	September 17, 2018 at 12:12PM		0	0	0	0
7	September 17, 2018 at 12:13PM		8	5	4	4
8	September 17, 2018 at 12:14PM		8	5	4	8
9	September 17, 2018 at 12:15PM		0	0	0	0
10	September 17, 2018 at 12:16PM		0	0	0	0
11	September 17, 2018 at 12:17PM		0	0	10	13
12	September 17, 2018 at 12:18PM	51	39	18	0	6
13	September 17, 2018 at 12:19PM	1	1	0	0	3
14	September 17, 2018 at 12:21PM	0	0	0	0	0
15	September 17, 2018 at 12:21PM	29	0	0	0	0
16	September 17, 2018 at 12:22PM	29	0	0	3	2
17	September 17, 2018 at 12:23PM	0	0	0	3	2
18	September 17, 2018 at 12:24PM	0	0	0	0	0

Planilha com Resumo do Tempo Total de Iluminação de Cada Luz

## LED N

18:31:00 (agora)	-----					
18:30:53	[ ...	True	... ]	+ 7 segundos		
18:30:48	[ ...	True	... ]	+ 5 segundos		
18:30:34	[ ...	False	... ]	+ 0 segundos		
18:30:13	[ ...	False	... ]	+ 0 segundos		
18:30:00	-----	True	-----	+ 13 segundos		
18:29:51	[ ...	True	... ]	TOTAL: 25 segundos		

Exemplo de Contagem do Tempo a Partir de Vários Dados



## Desafio Extra

Crie uma função que receba o número de um LED e uma data/horário, **retornando total de segundos em que o LED ficou aceso após aquela data**. Teste a função no shell do Thonny.

↪ DICA: use as funções `find` / `find_one`, os operadores `$lt` / `$gt` e a ordenação decrescente.

Crie uma função que obtenha o tempo aceso dos 5 LEDs desde 1 minuto atrás e **escreva esses valores + a data em uma linha na planilha**. Teste a função no shell do Thonny.

↪ DICA: lembre-se que existe um limite no número de parâmetros que podem ser passados para o IFTTT.

**Crie um Timer** que chame a função acima a cada 1 minuto.



[janks.link/micro/projeto05.zip](https://janks.link/micro/projeto05.zip)

Material do Projeto 05