

Projeto 01

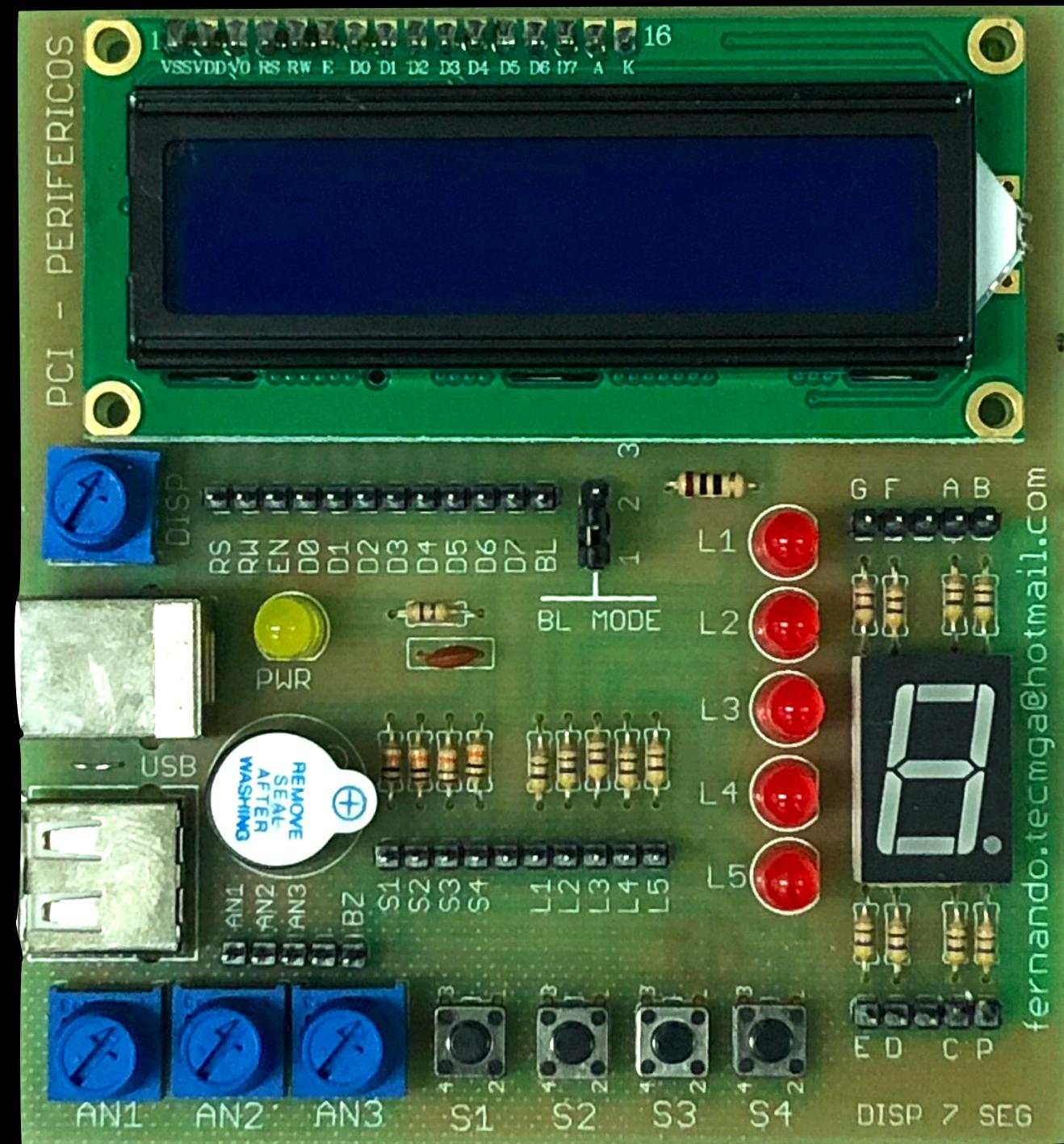
Controle de Mídia

Jan K. S. – janks@puc-rio.br

ENG 1419 – Programação de Microcontroladores

Testes Iniciais

GPIO 2, 3, 4, 5, 6, 7



GPIO 21
GPIO 22
GPIO 23
GPIO 24
GPIO 25

GPIO 11, 12, 13, 14

Conexões dos Botões, LEDs e LCD de Caracters com as Portas da GPIO



Testes Iniciais

Mude o estado do LED 1 (aceso/apagado) cada vez que alguém **apertar** o Botão 1.

↪ DICA: use as funções `when_pressed` e `toggle`.

Pisque 4 vezes o LED 2 ao apertar o Botão 2.

↪ DICA: use a função `blink`.

Pisque continuamente o LED 3, mantendo-o aceso por 1 segundo e apagado por 3 segundos.

↪ DICA: use os parâmetros opcionais da `blink`.

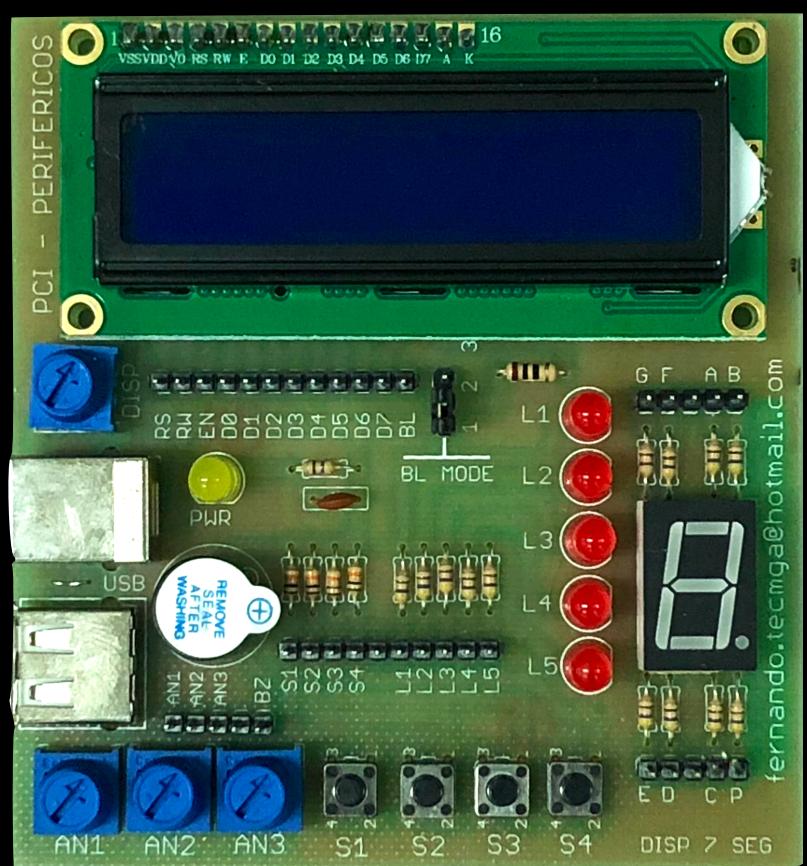
Mostre no LCD de caracteres **quantas vezes** o Botão 2 foi apertado.

↪ DICA: use uma variável global para contar.

Acenda o LED 5 se e somente se os botões 3 e 4 estiverem **simultaneamente pressionados**.

↪ DICA: use um `while True` com `sleep`.

Implementação



+



||

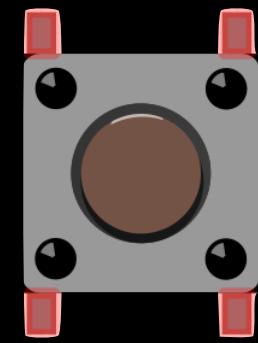


piPod

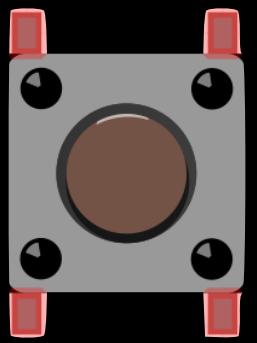
Controle de Mídia

acesso ao tocar
piscando na pausa

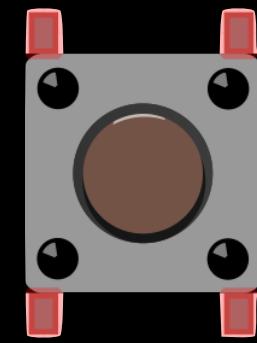
Nome da Música



Faixa Anterior

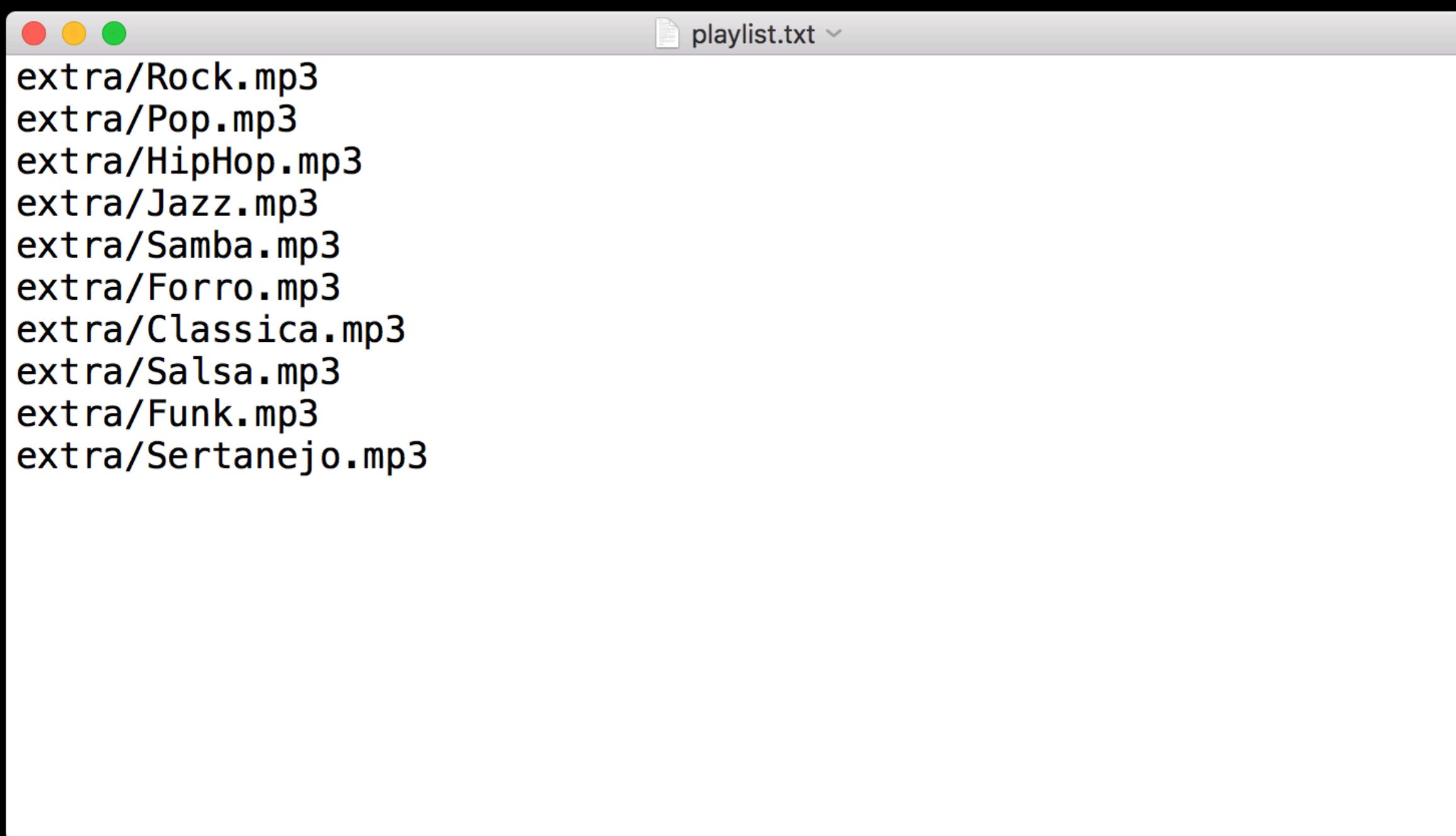


Tocar/Pausar



Próxima Faixa

Controles de Mídia com Botões, LED e LCD



Lista de Reprodução

Implementação



Implemente a função de **Tocar / Pausar**.

Use o LED 1 para indicar o status do player: se estiver pausado, pisque continuamente o LED; caso contrário, mantenha-o aceso.

Implemente a função de avançar para **Próxima Faixa**.

Implemente a função de **ir para a Faixa Anterior**. Contudo, se o tempo atual for maior que 2 segundos, o player deve **voltar ao começo da faixa atual**.

Exiba e mantenha atualizado o **nome da faixa atual** na linha superior do LCD de caracteres.

↪ DICA: use um while True com sleep.

Aperfeiçoamento



01b_implementacao.py

cópia
----->



01c_aperfeicoamento.py

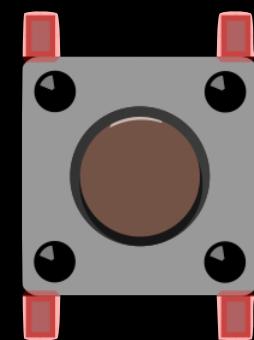
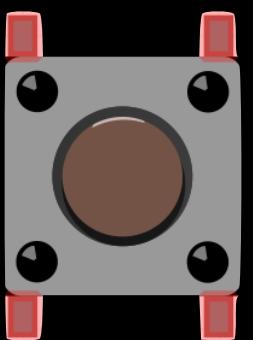
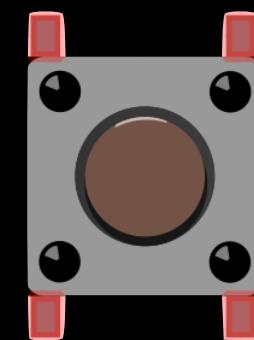
Cópia do Código da Implementação para o Aperfeiçoamento

Rolagem de Nomes Grandes

acesso ao tocar
piscando na pausa



Nome da Música
Tempo Atual e Total

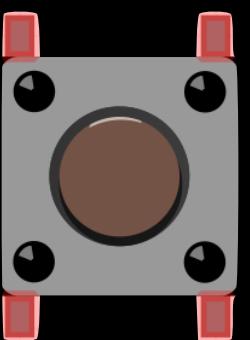


Faixa Anterior

Tocar/Pausar

Próxima Faixa
ou Acelerar

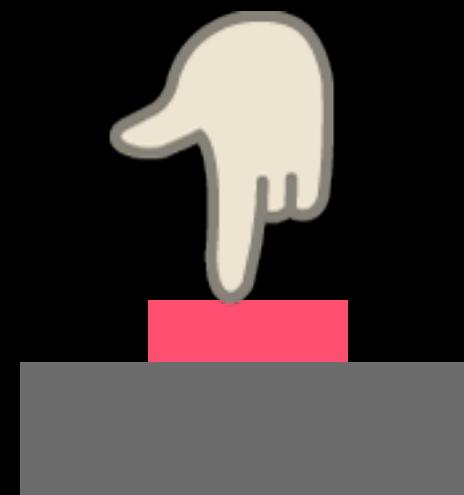
Novos Controles de Mídia com Botões, LED e LCD



Como fazer para
acelerar e pular faixa
com o mesmo botão?

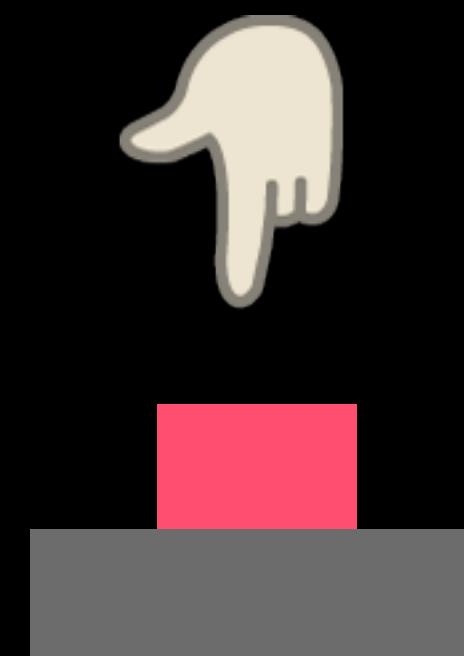


Próxima Faixa ou Acelerar



mantendo pressionado

Aumenta velocidade



ao soltar
se velocidade
estiver alta

Volte a velocidade ao normal

caso contrário

Vai para a próxima faixa

Lógica de Avançar ou Acelerar



Aperfeiçoamento

Acelere a faixa ao segurar o botão de Avançar. Volte ao normal ou pule de faixa ao soltar.

↪ DICA: substitua a função `when_pressed` pelas funções `when_held` e `when_released`.

Exiba o **tempo atual** e o **tempo total** da música na linha inferior do LCD (exatamente conforme ilustrado)

↪ DICA: pesquise no Google como formatar um número com zeros à esquerda em Python.

Role o **nome da faixa** no topo caso ela seja maior que o display.

↪ DICA: use a notação `texto[inicio:fim]` para capturar um trecho do texto.

Desafio Extra

```
playlist.txt
extra/Rock.mp3
extra/Pop.mp3
extra/HipHop.mp3
extra/Jazz.mp3
extra/Samba.mp3
extra/Forro.mp3
extra/Classica.mp3
extra/Salsa.mp3
extra/Funk.mp3
extra/Sertanejo.mp3
```

embaralha



```
nova_playlist.txt — Editado
extra/Samba.mp3
extra/Classica.mp3
extra/Sertanejo.mp3
extra/Forro.mp3
extra/HipHop.mp3
extra/Funk.mp3
extra/Jazz.mp3
extra/Rock.mp3
extra/Pop.mp3
extra/Salsa.mp3
```

Nova Lista de Reprodução



Toque as músicas em uma ordem aleatória. Para isso, leia o arquivo original da lista de músicas e gere um novo arquivo com as faixas embaralhadas.
↪ DICA: pesquise no Google como ler/escrever um arquivo e como embaralhar uma lista em Python.

Desafio Extra



janks.link/micro/projeto01.zip

Material do Projeto 01