

# Este é o CS50x

OpenCourseWare

Doar  (<https://cs50.harvard.edu/donate>)

David J. Malan (<https://cs.harvard.edu/malan/>)

malan@harvard.edu

 (<https://www.facebook.com/dmalan>)  (<https://github.com/dmalan>) 

(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>)

 (<https://orcid.org/0000-0001-5338-2522>) 

(<https://www.quora.com/profile/David-J-Malan>) 

(<https://www.reddit.com/user/davidjmalan>)  (<https://twitter.com/davidjmalan>)

## Lab 7: Songs

You are welcome to collaborate with one or two classmates on this lab, though it is expected that every student in any such group contribute equally to the lab.

GitHub now requires that you use SSH or a personal access token instead of a password to log in, but you can still use `check50` and `submit50`! See [cs50.ly/github](https://cs50.ly/github) (<https://cs50.ly/github>) for instructions if you haven't already!

Escreva consultas SQL para responder a perguntas sobre um banco de dados de músicas.

## Quando fazer

No sábado, 1º de janeiro de 2022, 1h59 GMT-3  (<https://time.cs50.io/2021-12-31T23:59:00-05:00>).

## Começando

Veja como baixar este laboratório em seu próprio CS50 IDE. Faça login no [CS50 IDE](https://ide.cs50.io/) (<https://ide.cs50.io/>) e, em uma janela de terminal, execute cada um dos [itens](https://ide.cs50.io/) (<https://ide.cs50.io/>) abaixo.

- Execute `cd` para garantir que você está em `~/` (ou seja, seu diretório inicial, também conhecido como `~`).
- Execute `wget https://cdn.cs50.net/2020/fall/labs/7/lab7.zip` para baixar um arquivo ZIP (compactado) com a distribuição desse problema.
- Execute `unzip lab7.zip` para descompactar esse arquivo.
- Execute `rm lab7.zip` seguido por `yes` ou `y` para excluir esse arquivo ZIP.
- Execute `ls`. Você deve ver um diretório chamado `lab7`, que estava dentro desse arquivo ZIP.
- Execute `cd lab7` para mudar para esse diretório.
- Execute `ls`. Você deve ver um `songs.db` arquivo e também alguns `.sql` arquivos vazios.

## Entendimento

---

Fornecido a você é um arquivo chamado `songs.db`, um banco de dados SQLite que armazena dados do [Spotify](https://developer.spotify.com/documentation/web-api/) (<https://developer.spotify.com/documentation/web-api/>) sobre músicas e seus artistas. Este conjunto de dados contém as 100 melhores músicas transmitidas no Spotify em 2018. Em uma janela de terminal, execute `sqlite3 songs.db` para que você possa começar a executar consultas no banco de dados.

Primeiro, quando `sqlite3` solicitar que você forneça uma consulta, digite `.schema` e pressione Enter. Isso produzirá as `CREATE TABLE` instruções que foram usadas para gerar cada uma das tabelas no banco de dados. Ao examinar essas declarações, você pode identificar as colunas presentes em cada tabela.

Observe que todo `artist` tem um `id` e um `name`. Observe, também, que cada música tem um `name`, um `artist_id` (correspondendo ao `id` do artista da música), bem como valores para dançabilidade, energia, tom, volume, fala (presença de palavras faladas em uma faixa), valência, andamento e duração da música (medido em milissegundos).

O desafio à sua frente é escrever consultas SQL para responder a uma variedade de perguntas diferentes, selecionando dados de uma ou mais dessas tabelas.

## Detalhes de Implementação

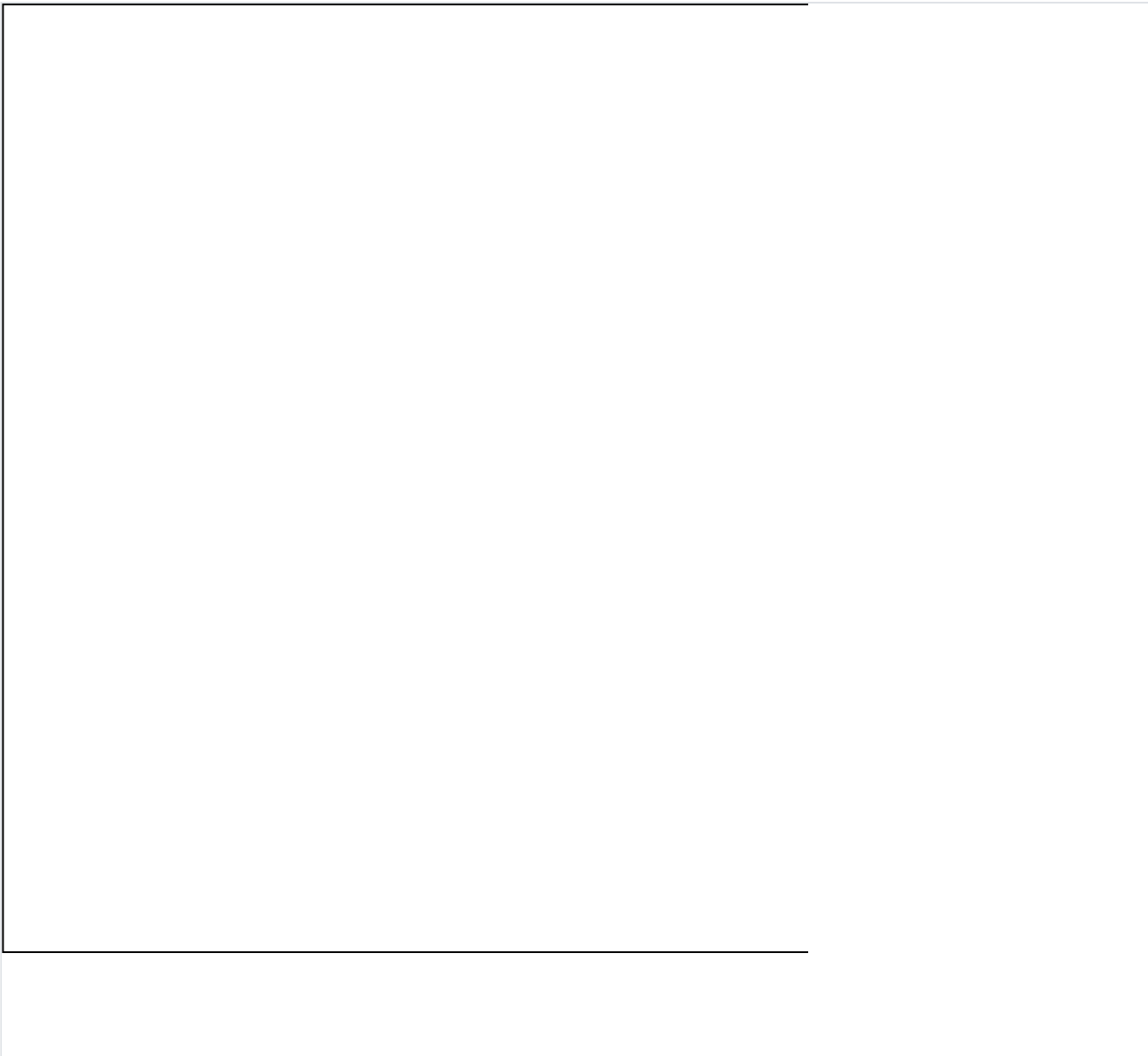
---

Para cada um dos problemas a seguir, você deve escrever uma única consulta SQL que produza os resultados especificados por cada problema. Sua resposta deve ter a forma de uma única consulta SQL, embora você possa aninhar outras consultas dentro de sua consulta. Você **não deve** presumir nada sobre os programas `id` de nenhuma música ou artista em particular: suas perguntas devem ser precisas, mesmo se o `id` de qualquer música ou pessoa em particular for diferente. Finalmente, cada consulta deve retornar apenas os dados necessários para responder

eficiente. Finalmente, cada consulta deve retornar apenas os dados necessários para responder à pergunta: se o problema pede apenas que você envie os nomes das músicas, por exemplo, então sua consulta não deve também produzir o tempo de cada música.

1. Em `1.sql`, escreva uma consulta SQL para listar os nomes de todas as músicas no banco de dados.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.
2. Em `2.sql`, escreva uma consulta SQL para listar os nomes de todas as músicas em ordem crescente de andamento.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.
3. Em `3.sql`, escreva uma consulta SQL para listar os nomes das 5 músicas mais longas, em ordem decrescente de duração.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.
4. Em `4.sql`, escreva uma consulta SQL que liste os nomes de todas as músicas que possuem dancilidade, energia e valência maior que 0,75.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.
5. Em `5.sql`, escreva uma consulta SQL que retorne a energia média de todas as músicas.
  - Sua consulta deve gerar uma tabela com uma única coluna e uma única linha contendo a energia média.
6. Em `6.sql`, escreva uma consulta SQL que liste os nomes das músicas de Post Malone.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.
  - Você não deve fazer suposições sobre o que é Post Malone `artist_id`.
7. Em `7.sql`, escreva uma consulta SQL que retorne a energia média das músicas de Drake.
  - Sua consulta deve gerar uma tabela com uma única coluna e uma única linha contendo a energia média.
  - Você não deve fazer suposições sobre o que é Drake `artist_id`.
8. Em `8.sql`, escreva uma consulta SQL que liste os nomes das músicas que apresentam outros artistas.
  - Músicas que apresentam outros artistas incluirão “feat.” em nome da música.
  - Sua consulta deve gerar uma tabela com uma única coluna para o nome de cada música.

## Passo a passo



## Dicas

- Consulte [esta referência de palavras-chave SQL](https://www.w3schools.com/sql/sql_ref_keywords.asp) ([https://www.w3schools.com/sql/sql\\_ref\\_keywords.asp](https://www.w3schools.com/sql/sql_ref_keywords.asp)) para alguma sintaxe SQL que pode ser útil!

### ► Não sabe como resolver?

## Testando

Execute o seguinte para avaliar a exatidão do seu código usando `check50`.

```
check50 cs50/labs/2021/x/songs
```

## Como enviar

Execute o seguinte para enviar seu trabalho.

```
submit50 cs50/labs/2021/x/songs
```

## Reconhecimentos

Conjunto de dados do [Kaggle](https://www.kaggle.com/nadintamer/top-spotify-tracks-of-2018) (<https://www.kaggle.com/nadintamer/top-spotify-tracks-of-2018>)

.

