

Projeto de Bases de Dados (CC2005)

1. Elementos do grupo

Grupo nº G2D

Nº mecanográfico	Nome
202306055	Carolina Gomes Ferreira Baptista Proença
202306411	Daniel Rodrigues Gomes
202307178	Eduarda Meireles de Sousa Ricardo das Neves
202304201	Maria Miguel da Silva Morais

2. Universo considerado e modelo de classes UML

Tema: Discografia da Taylor Swift

O universo considerado para a base de dados em estudo é a discografia da artista norte-americana Taylor Swift, desde a primeira música lançada pela mesma em 2006 até ao ultimo feature, em junho de 2024.

Cada linha das 359 da base de dados representa uma música que a cantora faz parte e cada uma é descrita por 13 atributos:

- título (album_title) e url (album_url) do álbum em que a música está inserida: NA nos casos em que a música não faz parte de um álbum;
- categoria (category);
- número da faixa do álbum (album_track_number): 0 nos casos em que a música não faz parte de um álbum;
- título (song_title);
- url (song_url);
- data de lançamento (song_release_date);
- número de visualizações da página da música (song_page_views);
- letra (song_lyrics);
- artistas, produtores e escritores (song_artists, song_producers, song_writers);
- tags (song_tags);

Para modelar a base de dados que descreve a discografia da artista Taylor Swift, o diagrama de classes UML foi desenvolvido com base nas características principais de cada música e nas suas relações com os outros elementos. O objetivo foi representar de forma clara e precisa os atributos e associações entre as entidades envolvidas.

1. Classes Principais:

- **Song**: Representa cada música individual, que é a unidade central do modelo. Esta classe inclui atributos como título, URL, data de lançamento, letra e número de visualizações.
- **Album**: Representa os álbuns aos quais as músicas pertencem. Foi necessário incluir a possibilidade de músicas que não estão associadas a nenhum álbum.
- **Tag**: Permite categorizar músicas através de palavras-chave, representando informações adicionais como temas ou géneros.
- **Person**: Abrange artistas, produtores e escritores, utilizando uma relação com a classe **Role** para diferenciar os papéis desempenhados por cada pessoa.
- **Role**: Define os possíveis papéis associados às pessoas na produção de uma música, como cantor, produtor ou compositor.
- **Track**: Representa a relação entre músicas e suas posições específicas dentro de álbuns.

2. Decisões de Modelagem:

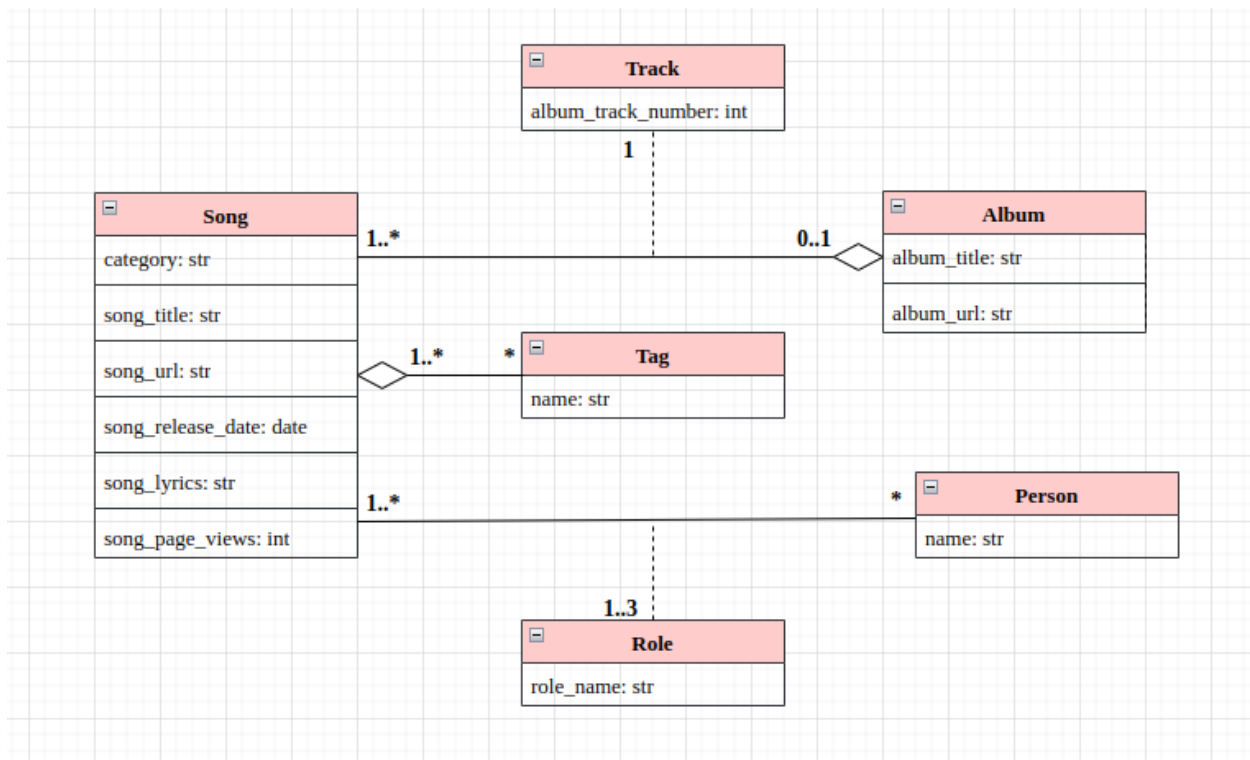
- Para capturar a relação entre **Song** e **Album**, foi definida uma associação opcional, já que nem todas as músicas pertencem a álbuns.
- As tags foram modeladas com um relacionamento de muitos-para-muitos para permitir que uma música tenha múltiplas tags e vice-versa.
- A relação entre **Person** e **Song** foi ajustada com o uso da classe-associação **Role**, permitindo uma maior flexibilidade na representação de múltiplos papéis para uma única pessoa em diferentes músicas.

3. Relações e Multiplicidades:

- As multiplicidades foram ajustadas para refletir a realidade do domínio:
 - Uma música pode ter no máximo uma posição num álbum, mas pode existir fora de qualquer álbum.
 - Um álbum pode conter várias músicas, mas também pode estar ausente para músicas individuais.
 - Uma música pode estar associada a várias pessoas que desempenham diferentes papéis.

Com base nessa modelagem, o diagrama UML a seguir foi criado, representando as classes, os seus atributos e as relações entre elas. Esta estrutura fornece uma visão clara da

organização da base de dados e permite uma implementação eficiente e flexível para o armazenamento e análise das informações.



3. Modelo relacional

Com base no diagrama UML apresentado, o modelo relacional foi construído considerando as dependências funcionais identificadas durante o mapeamento:

- **SongId** → {category, song_title, song_url, song_release_date, song_lyrics, song_page_views, tag_name}
- **AlbumId** → {album_title, album_url}
- {**SongId**, **AlbumId**} → album_track_number
- **PersonId** → person_name
- {**PersonId**, **SongId**} → role_name

A chave primária da relação foi identificada como {**SongId**, **AlbumId**, **PersonId**}, já que, a partir dessa combinação, é possível derivar todos os atributos restantes do modelo.

1. Tabela "Discography":

- Criada como elo de ligação para os atributos que compõem a chave primária.
- É uma **join table**, derivada da classe-associação "Role" no diagrama UML.
- Todos os atributos dessa tabela, incluindo **role_name**, são definidos como **Primary Keys**, considerando que uma pessoa pode desempenhar múltiplos papéis numa mesma música.

2. Tabelas "Songs", "Albums" e "People":

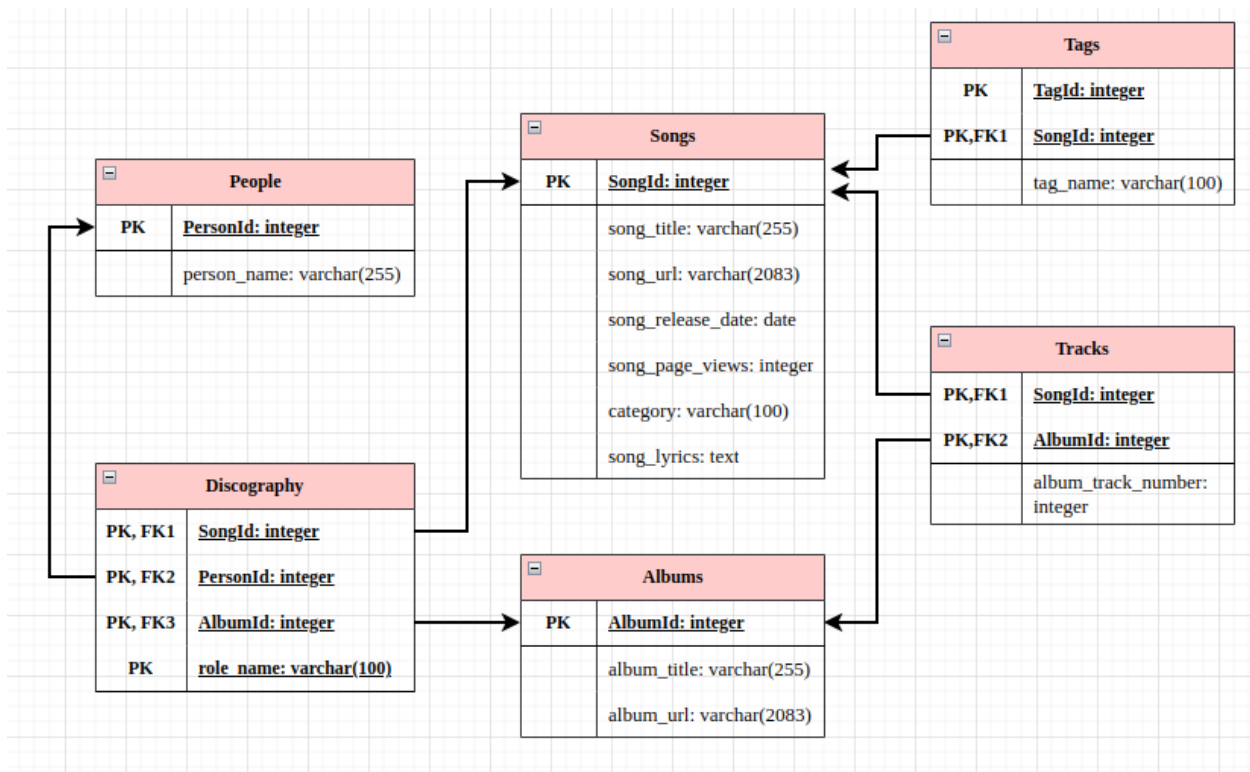
- Foram mapeadas diretamente das classes correspondentes no diagrama UML.
- Contêm os mesmos atributos descritos no diagrama e incluem o identificador único (Id) de cada objeto.

3. Tabela "Tracks":

- Derivada da classe-associação "Track" no diagrama UML.
- É representada no modelo relacional com o atributo **album_track_number**, tendo como **Primary Key** a combinação dos Ids das tabelas associadas (**Songs** e **Albums**).

4. Tabela "Tags":

- Representa a relação de agregação entre músicas e tags.
- Inclui o Id da tag como **Primary Key**, combinado com o Id da música, garantindo que cada tag esteja vinculada a uma ou mais músicas, conforme o diagrama UML.



4. Povoamento de tabelas

A forma como povoamos a base de dados está descrita no ficheiro "README.md".

- **Que fontes de dados foram utilizadas para preencher a base de dados?**

O ficheiro Excel "ts_discography.xlsx", que contém informação detalhada sobre a discografia da Taylor Swift foi a única fonte de dados utilizada para este projeto e todos os dados foram extraídos e transformados diretamente a partir dele.

- **Como é que a base de dados foi preenchida?**

A base de dados foi preenchida através de um processo sistemático:

1. Carregamento do ficheiro Excel:

- O ficheiro Excel foi lido com a ferramenta "Pandas".
- Os dados foram pré-processados para remover caracteres desnecessários, como parênteses retos ([]) e aspas (') das colunas "song_artists", "song_lyrics", "song_writers", "song_producers" e "song_tags", utilizando uma função de limpeza.

2. Criação e preenchimento de tabelas:

As tabelas da base de dados foram criadas e preenchidas numa ordem lógica para manter a integridade referencial:

- **Tabelas base:**
 - ◆ **Albums:** Criada a partir das colunas album_title e album_url. Os IDs exclusivos AlbumId) foram gerados automaticamente.
 - ◆ **Songs:** Criada a partir das colunas relacionadas com as músicas: song_title, song_url, song_release_date, song_page_views, category e song_lyrics. Os IDs únicos (SongId) foram gerados automaticamente.
 - ◆ **Tags:** Associadas às canções, foram processadas a partir da coluna song_tags. Cada tag foi ligada à canção correspondente utilizando o seu SongId.
 - ◆ **People:** Os indivíduos envolvidos (escritores, produtores e artistas) foram extraídos das respetivas colunas (song_writers, song_producers, song_artists) e normalizados numa tabela separada. Foram gerados IDs únicos (PersonId).
- **Tabelas de relacionamento:**
 - ◆ **Discography:** Relacionamentos mapeados entre músicas, álbuns e pessoas com funções específicas (escritor, produtor ou artista).
 - ◆ **Tracks:** Associa-se cada música ao seu número de faixa no álbum, utilizando a coluna album_track_number diretamente do ficheiro Excel.

3. Validação e transformação:

- Garantimos que todas as colunas necessárias estavam presentes no DataFrame antes do processamento
- Utilizamos 'drop_duplicates' para evitar entradas duplicadas em todas as tabelas;
- Mantivemos a integridade referencial:
 - Mapeamos as músicas para os respectivos IDs (SongId) na tabela Songs;
 - Álbuns mapeados para os seus IDs (AlbumId) na tabela Albums;
 - Ligamos pessoas (PersonId) com base em nomes normalizados na tabela People.

4. Inserção de dados na base de dados:

- Conectámos à base de dados SQLite (taylor_swift.db) usando a biblioteca sqlite3.

- Inserimos dados em cada tabela usando o método `to_sql` do “Pandas” com a opção `if_exists='replace'`. Isso garantiu que cada execução do script substituísse os dados existentes, criando um processo de população da base de dados limpo e reproduzível.
- Depois de inserir dados em todas as tabelas, a conexão com a base de dados foi fechada.

5. Testes

- Após a povoação estar completa foram realizados diversos testes pelos membros do grupo para assegurar que não havia perda de informação nem informação redundante relativamente ao ficheiro Excel inicial.

- **Número de linhas por cada tabela**

Nome da tabela	Nº de entradas
People	2
Discography	4
Songs	7
Albums	3
Tags	3
Tracks	3

5. Interrogações SQL

Interrogação 1: Quais são os nomes dos albuns e os seus respetivos url's?

O que se pretende obter: Uma coluna com os nomes de todos os albuns da discografia e uma coluna com os urls da respetiva página da internet.

Código:

```
SELECT a.album_title, a.album_url
FROM Albums a
```

Extrato de Resultado:

album_title	album_url
Taylor Swift	https://genius.com/albums/Taylor-Swift/Taylor-Swift
Fearless (Platinum Edition)	https://genius.com/albums/Taylor-Swift/Fearless-Platinum-Edition
Fearless (Taylor's Version)	https://genius.com/albums/Taylor-Swift/Fearless-Taylors-Version
Speak Now (Deluxe)	https://genius.com/albums/Taylor-Swift/Speak-Now-Deluxe
Speak Now (Taylor's Version)	https://genius.com/albums/Taylor-Swift/Speak-Now-Taylors-Version
Red (Deluxe Version)	https://genius.com/albums/Taylor-Swift/Red-Deluxe-Version
Red (Taylor's Version)	https://genius.com/albums/Taylor-Swift/Red-Taylors-Version

Interrogação 2: Quais músicas têm ‘Taylor's Version’ no nome e a tag “Ballad” atribuída? Ordene pelos nomes das músicas.

O que se pretende obter: Uma coluna com o nome das músicas ordenadas alfabeticamente, em que o título tem “Taylor's Version” e tem associada a tag “Ballad”

Código:

```
SELECT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Tags t
WHERE s.song_title LIKE "%Taylor's Version%" and t.tag_name='Ballad'
GROUP BY s.song_title
ORDER BY s.song_title;
```

Extrato de resultado:

song_title
"Slut!" (Taylor's Version) [From The Vault]
All Too Well (10 Minute Version) (Taylor's Version) [From The Vault]
Back To December (Taylor's Version)
Begin Again (Taylor's Version)
Better Man (Taylor's Version) [From The Vault]
Breathe (Taylor's Version) (Ft. Colbie Caillat)
Castles Crumbling (Taylor's Version) [From The Vault] (Ft. Hayley Williams)

Interrogação 3: Quais são as músicas da categoria 'Non-Album Songs'?
Apresente o nome, o url e a data de lançamento e ordene pela data de lançamento.

O que se pretende obter: Uma coluna com o nome da música, uma coluna com o url respetivo da música e uma coluna com a data de lançamento da mesma, ordenadas pela última coluna

Código:

```
SELECT DISTINCT s.song_title, s.song_url, s.song_release_date
FROM Discography d
NATURAL JOIN Songs s
WHERE s.category='Non-Album Songs'
ORDER BY s.song_release_date;
```

Extrato de resultado:

song_title	song_url	song_release_date
Baby Don't You Break My Heart Slow	https://genius.com/Taylor-swift-baby-lyrics	2006-10-26 00:00:00
Umbrella (Live from SoHo)	https://genius.com/Taylor-swift-umbrella-live-from-soho-lyrics	2007-09-01 00:00:00
Last Christmas	https://genius.com/Taylor-swift-last-christmas-lyrics	2007-10-14 00:00:00
Christmases When You Were Mine	https://genius.com/Taylor-swift-christmas-es-when-you-were-mine-lyrics	2007-10-14 00:00:00
Santa Baby	https://genius.com/Taylor-swift-santa-baby-lyrics	2007-10-14 00:00:00
Silent Night	https://genius.com/Taylor-swift-silent-night-lyrics	2007-10-14 00:00:00
Christmas Must Be Something More	https://genius.com/Taylor-swift-christmas-must-be-something-more-lyrics	2007-10-14 00:00:00

Interrogação 4: Quais são as faixas número 13 de cada album? Apresente o nome das músicas e dos respetivos albuns.

O que se pretende obter: Uma coluna com o nome da música e uma coluna com o nome do álbum em que essa música é a faixa número 13.

Código:

```
SELECT DISTINCT s.song_title, a.album_title
FROM Discography d
NATURAL JOIN Tracks t
NATURAL JOIN Songs s
NATURAL JOIN Albums a
WHERE t.album_track_number=13;
```

Extrato de resultado:

song_title	album_title
Invisible	Taylor Swift
Breathe (Ft. Colbie Caillat)	Fearless (Platinum Edition)
Change (Taylor's Version)	Fearless (Taylor's Version)
Last Kiss	Speak Now (Deluxe)
Last Kiss (Taylor's Version)	Speak Now (Taylor's Version)
The Lucky One	Red (Deluxe Version)
The Lucky One (Taylor's Version)	Red (Taylor's Version)

Interrogação 5: Quais músicas do álbum 'reputation' são produzidas por Jack Antonof? Ordene pelos nomes das músicas.

O que se pretende obter: Uma coluna com o nome das músicas do album "reputation" produzidas por Jack Antonof, ordenadas alfabeticamente.

Código:

```
SELECT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN People p
NATURAL JOIN Albums a
WHERE a.album_title='reputation' and p.person_name='Jack Antonoff'and
d.role_name='Producer'
ORDER by s.song_title;
```

Resultado:

song_title
Call It What You Want
Dress

song_title

Getaway Car

Look What You Made Me Do

New Year's Day

This Is Why We Can't Have Nice Things

Interrogação 6: Quais as músicas que são parcerias com outros artistas? Apresente o nome das músicas e os dos artistas convidados, ordenando pelo nome dos artistas convidados.

O que se pretende obter: Uma coluna com o nome das músicas e uma coluna com o nome dos artistas que fazem parte da parceria na respetiva música, ordenados pelo nome do artista.

Código:

```
SELECT DISTINCT s.song_title, p.person_name
FROM Discography d
NATURAL JOIN People p
NATURAL JOIN Songs s
WHERE d.role_name='Artist' AND p.person_name!='Taylor Swift'
ORDER BY p.person_name;
```

Extrato de resultado:

song_title	person_name
Both of Us	B.o.B
Two Is Better Than One	BOYS LIKE GIRLS
Birch	Big Red Machine
Renegade	Big Red Machine
Anti-Hero (Remix)	Bleachers
exile (Ft. Bon Iver)	Bon Iver
evermore (Ft. Bon Iver)	Bon Iver

Interrogação 7: Apresente o número total de views das páginas das músicas do album 'folklore (deluxe version)'.

O que se pretende obter: Uma coluna (num) com apenas o valor correspondente ao total de visualizações das páginas das músicas do album 'folklore (deluxe version)'.

Código:

```
WITH tab AS(SELECT DISTINCT s.song_title, s.song_page_views
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Albums a
WHERE a.album_title='folklore (deluxe version)')
SELECT SUM(tab.song_page_views) num
FROM tab;
```

Resultado:

num
28111700

Interrogação 8: Apresente a média das views das páginas das músicas do album 'Speak Now (Deluxe)'.

O que se pretende obter: Uma coluna (num) com o número médio de visualizações das páginas das músicas do álbum "Speak Now (Deluxe)"

Código:

```
WITH tab AS(SELECT DISTINCT s.song_title, s.song_page_views
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Albums a
WHERE a.album_title='Speak Now (Deluxe)')
SELECT AVG(tab.song_page_views) num
FROM tab;
```

Resultado:

num
372441.17647058825

Interrogação 9: Quantas músicas foram lançadas a partir de 2010?

O que se pretende obter: Uma coluna (num) com o número de músicas lançadas a partir de 2010.

Código:

```
WITH tab AS(SELECT s.song_title
FROM Discography d
NATURAL JOIN Songs s
WHERE s.song_release_date NOT LIKE '200%'
GROUP BY s.song_title)
SELECT COUNT(*) num FROM tab;
```

Resultado:

num
311

Interrogação 10: Quantas vezes é que a palavra night é referida em toda a discografia?

O que se pretende obter: Uma coluna (num) com o número de vezes que a palavra "night" aparece na letra de uma música.

Código:

```
WITH tab AS(SELECT s.song_title
FROM Discography d
NATURAL JOIN Songs s
WHERE s.song_lyrics LIKE '%night%'
GROUP BY s.song_title)
SELECT COUNT(*) num FROM tab;
```

Resultado:

num
163

Interrogação 11: Quais são as músicas lançadas antes 2010 em que a Taylor não é produtora? Apresente o nome das músicas e o número de views da página, ordenando decrescentemente pelo número de views da pagina.

O que se pretende obter: Uma coluna com os nomes das músicas lançadas antes de 2010 nas quais a Taylor não é produtora e uma coluna com o total de visualizações da página da respetiva música, ordenadas pelo número de visualizações descendente.

Código:

```
SELECT DISTINCT s.song_title, s.song_page_views
FROM Discography d
```

```

NATURAL JOIN People p
NATURAL JOIN Songs s
WHERE s.song_title NOT IN(SELECT DISTINCT s.song_title
FROM Discography d
NATURAL JOIN People p
NATURAL JOIN Songs s
WHERE s.song_release_date LIKE '200%' AND d.role_name='Producer' AND
p.person_name='Taylor Swift')
ORDER BY s.song_page_views DESC

```

Extrato de resultado:

song_title	song_page_views
All Too Well (10 Minute Version) (Taylor's Version) [From The Vault]	4900000
Fortnight (Ft. Post Malone)	3400000
cardigan	3300000
exile (Ft. Bon Iver)	2700000
So Long, London	2700000
Lover	2600000
The Tortured Poets Department	2600000

Interrogação 12: Quais os 3 albuns com o maior número de músicas? Apresente o nome dos albuns e o respetivo número de músicas, ordenando decrescentemente pelo número de músicas.

O que se pretende obter: Uma coluna com o nome dos 3 albuns e uma coluna (num) com o número de faixas do respetivo album, ordenadas pelo número de faixas descendente.

Código:

```

WITH tab AS(SELECT a.album_title, s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Albums a
GROUP BY a.album_title, s.song_title
ORDER BY a.album_title)
SELECT tab.album_title, COUNT(tab.song_title) num
FROM tab
GROUP BY tab.album_title
ORDER BY num DESC
LIMIT 3;

```

Resultado:

album_title	num
THE TORTURED POETS DEPARTMENT: THE ANTHOLOGY	31
Red (Taylor's Version)	30
Fearless (Taylor's Version)	25

Interrogação 13: Apresente o nome das músicas escritas unicamente pela Taylor Swift

O que se pretende obter: Uma coluna com o nome das músicas que foram escritas somente pela Taylor Swift.

Código:

```
SELECT DISTINCT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Albums a
NATURAL JOIN People p
WHERE d.role_name='Writer' AND p.person_name='Taylor Swift' AND s.song_title NOT IN
(SELECT DISTINCT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Albums a
NATURAL JOIN People p
WHERE d.role_name='Writer' and p.person_name!='Taylor Swift')
```

Extrato de resultado:

song_title
The Outside
Should've Said No
Our Song
Jump Then Fall
The Other Side Of The Door
Fifteen
Love Story

Interrogação 14: Quais as músicas que têm associadas simultaneamente a tag 'Rock' e a tag 'Orchestral' Apresente além do nome da música, também a sua respetiva categoria.

O que se pretende obter: Uma coluna com os nomes das músicas com as tags "Ballad" e "Orchestral" e uma coluna com as respetivas categorias.

Código:

```
SELECT DISTINCT s.song_title, s.category
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Tags t
WHERE s.song_title IN (SELECT DISTINCT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Tags t
WHERE t.tag_name='Rock') AND s.song_title IN (SELECT DISTINCT s.song_title
FROM Discography d
NATURAL JOIN Songs s
NATURAL JOIN Tags t
WHERE t.tag_name='Orchestral')
```

Extrato de resultado:

song_title	category
Dear John	Speak Now
Enchanted	Speak Now
Innocent	Speak Now
Haunted	Speak Now
Back To December (Taylor's Version)	Speak Now (TV)
Enchanted (Taylor's Version)	Speak Now (TV)
Innocent (Taylor's Version)	Speak Now (TV)

6. Aplicação Python

A forma de executar e poder interagir com a interface está descrita no ficheiro "README.md".

“Endpoint”	Funcionalidade
http://localhost:5001/	Página de entrada
http://localhost:5001/songs	Página que lista todas as músicas e os respetivos atributos
http://localhost:5001/songs/<int:id>	Página que apresenta os detalhes de uma música através do seu ID
http://localhost:5001/songs/<int:id>/lyrics	Letra de uma música através do seu ID
http://localhost:5001/songs/search/<expr>	Página que apresenta a(s) música(s) cujo título contém a expressão pesquisada
http://localhost:5001/albums	Página que lista todos os álbuns e os respetivos atributos
http://localhost:5001/albums/<int:id>	Página que apresenta as faixas do respetivo álbum
http://localhost:5001/albums/search/<expr>	Página que apresenta o(s) álbum(s) cujo nome contém a expressão pesquisada
http://localhost:5001/people	Página que lista todas as pessoas e detalhes de cada uma das suas participações
http://localhost:5001/people/<int:id>	Página que apresenta os detalhes das participações de uma pessoa através do seu ID
http://localhost:5001/questions	Página que lista as perguntas
http://localhost:5001/questions/1 até /questions/14	Página que apresenta a pergunta e o respetivo resultado

7. Conclusão

Com este trabalho, alcançámos os resultados esperados, confirmando a ausência de falhas ou redundâncias de informação em relação ao ficheiro de Excel inicialmente proposto.

Os desafios encontrados durante a povoação da base de dados foram resolvidos com relativa facilidade, graças ao facto de o tema do universo em estudo, a música, ser bastante familiar para os membros do grupo.

Ao longo do processo, adquirimos conhecimentos valiosos, tanto na povoação de bases de dados como no desenvolvimento de interfaces HTML. Estamos confiantes de que estas aprendizagens serão de grande utilidade para o nosso percurso académico e profissional, contribuindo significativamente para o desenvolvimento das nossas competências técnicas.

8. Referências

- <https://chatgpt.com/>
- <https://github.com/jresende/MoviesApp>
- aula prática 9 MoodleUP
- aula teórica 12 MoodleUP