

# Base de Dados I

Trabalho de avaliação

César Damásio Nº 17177

Vasco Oliveira Nº17888

DOCENTE: ELSA RODRIGUES



## Índice

Introdução.....	4
Modelo Conceptual.....	5
Normalização de dados .....	6
Modelo Relacional.....	7
Criação da base de dados.....	8
Modelo Físico .....	11
Inserção de dados em SQL .....	12
Consultas e subconsultas em SQL.....	12
Access.....	27
Formulários.....	31
Relatórios.....	34
Conclusão .....	37
Bibliografia .....	38

## **Introdução**

Este trabalho prático consiste na construção de uma base de dados, e numa perspectiva geral, permite a aplicação dos conhecimentos e informações aprendidas.

Pretende-se desenvolver uma base de dados de músicas, que possa armazenar informações relevantes e necessárias para possíveis consultas.

Primeiramente será desenvolvido um modelo conceptual, que esquematiza a base de dados a ser implementada, assim como um modelo relacional que irá servir de base para o código a ser gerado em SQL no software SQL Server Management Studio 2014.

Como objetivos deste projeto tem-se a organização do código e o bom funcionamento do mesmo, contribuindo para a criação de uma base de dados acessível, tal como a eficácia das consultas e subconsultas efetuadas.

## Modelo Conceptual

A criação do modelo conceptual é fundamental no desenvolvimento de qualquer sistema de Bases de Dados, sendo a base de todo o processo pois fornece auxílio na sua estruturação.

Consiste essencialmente na definição das entidades e dos seus atributos, assim como os seus respectivos relacionamentos entre elas e o tipo de cardinalidade existente.

Pretende-se com este modelo evitar redundâncias, limitar a utilização de entidades fracas e não utilizar entidades se for suficiente um atributo. Importa mencionar que o modelo conceptual aqui apresentado não constitui a nossa primeira versão. Procedeu-se a diversas alterações, no sentido de diminuir a probabilidade de ocorrência de redundâncias e inconsistências.

Na Figura X apresenta-se o nosso modelo conceptual para a base de dados em questão.

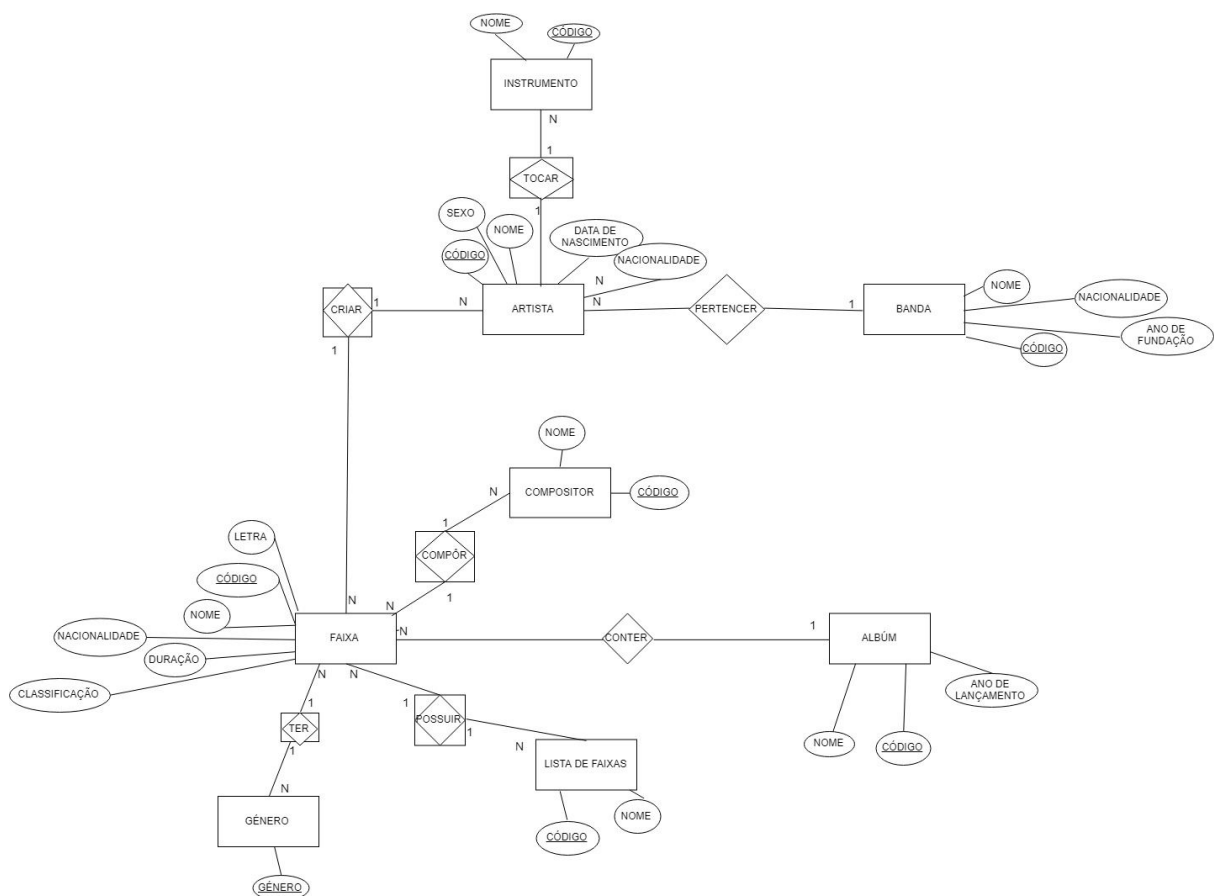


Figura X - Modelo Conceptual Arquivo MP4

### Observações à nossa visão para implementação deste modelo:

- Uma faixa pode ter vários géneros, portanto criámos uma nova entidade Género e fizemos uma ligação n para n entre Faixa e Género por meio de Faixa\_Género. O mesmo sucede para Compositor e ListaFaixas; e para Artista e Instrumento.
- Assumimos que cada faixa pertence apenas a um album, tal como cada artista pertence apenas a uma banda
- Não implementámos a opção de permita registar artistas individuais e grupo de artistas com a indicação de quais os artistas que compõem o grupo.

## **Normalização de dados**

A normalização é também uma etapa importante na criação da base de dados final. Consiste em analisar o esquema de relações/dependências entre atributos, baseados em chaves idênticas, com um conjunto de regras bem definidas com vista à eliminação dos dados.

Assim, a normalização vai permitir uma diminuição da redundância da informação (evitar repetição da informação), evitar a dependência de inconsistência, manter a integridade dos dados (uma vez que impede que quando se elimina um dado desapareçam outros), obter um armazenamento consistente e um eficiente acesso aos dados da base de dados.

Neste trabalho, a terceira forma normal foi suficiente. Aplicando a normalização ao modelo conceptual do arquivo MP4, podemos verificar a inexistência de anomalias.

## Modelo Relacional

O modelo relacional é criado após o modelo conceptual. Representa o estabelecimento de relações entre as diferentes tabelas normalizadas, de maneira a poder traduzir o modo como a informação está relacionada na realidade, e assim passar diretamente a informação necessária a implementar.

Eis o modelo relacional relativo à nossa base de dados:

Banda(cod\_banda, nome\_banda, nacionalidade\_banda, data\_fundacao\_banda)

Artista(cod\_artista, nome\_artista, nacionalidade, data\_nascimento, sexo, cod\_banda)

Album(cod\_album, nome\_album, género, ano\_lançamento)

Faixa(cod\_faixa, nome\_faixa, nacionalidade\_faixa, duracao\_faixa, letra\_faixa, classificação\_faixa, cod\_album)

Compositor(cod\_compositor, nome\_compositor)

ListaFaixas(cod\_lista, nome\_lista)

Artista\_Faixa(cod\_faixa, cod\_artista)

Faixa\_ListaFaixas(cod\_faixa, cod\_lista)

Faixa\_Compositor(cod\_faixa, cod\_compositor)

Género(cod\_genero, nome\_genero)

Faixa\_Género(cod\_faixa, cod\_genero)

Instrumento(cod\_instrumento, nome\_instrumento)

Artista\_Instrumento(cod\_artista, cod\_instrumento)

## Criação da base de dados

Abaixo segue o código utilizado para a criação da base de dados e para a criação das tabelas:

```
CREATE DATABASE MP4
```

```
USE MP4
```

```
GO
```

```
CREATE TABLE Banda(  
cod_banda INT PRIMARY KEY,  
nome_banda VARCHAR(50),  
nacionalidade_banda VARCHAR(50),  
ano_fundacao SMALLINT  
CHECK (ano_fundacao >= 0)  
)
```

```
CREATE TABLE Artista(  
cod_artista INT PRIMARY KEY,  
nome_artista VARCHAR(50),  
data_nascimento_artista DATE,  
nacionalidade_artista VARCHAR(50),  
cod_banda INT  
FOREIGN KEY(cod_banda)  
REFERENCES Banda(cod_banda)  
)
```

```
CREATE TABLE Album(  
cod_album INT PRIMARY KEY,  
nome_album VARCHAR(50),  
ano_lançamento SMALLINT,  
CHECK(ano_lançamento >= 0),  
)
```

```
CREATE TABLE Faixa(  
cod_faixa INT PRIMARY KEY,  
nome_faixa VARCHAR(50),  
nacionalidade_faixa VARCHAR(50),  
duração_faixa FLOAT,  
classificacao TINYINT,  
CHECK (classificacao BETWEEN 0 AND 10),  
letra_faixa TEXT,  
cod_album INT  
FOREIGN KEY(cod_album)
```



```
REFERENCES Album(cod_album),  
)
```

```
CREATE TABLE Compositor(  
cod_compositor INT PRIMARY KEY,  
nome_compositor VARCHAR(50)  
)
```

```
CREATE TABLE ListaFaixas(  
cod_lista INT PRIMARY KEY,  
nome_lista VARCHAR(50),  
)
```

```
CREATE TABLE Artista_Faixa(  
cod_artista INT  
    FOREIGN KEY(cod_artista)  
    REFERENCES Artista(cod_artista),  
cod_faixa INT  
    FOREIGN KEY(cod_faixa)  
    REFERENCES Faixa(cod_faixa),
```

```
PRIMARY KEY(cod_artista, cod_faixa)  
)
```

```
CREATE TABLE Faixa_ListaFaixas(  
cod_faixa INT  
    FOREIGN KEY(cod_faixa)  
    REFERENCES Faixa(cod_faixa),  
cod_lista INT  
    FOREIGN KEY(cod_lista)  
    REFERENCES ListaFaixas(cod_lista),
```

```
PRIMARY KEY(cod_faixa, cod_lista)  
)
```

```
CREATE TABLE Faixa_Compositor(  
cod_faixa INT  
    FOREIGN KEY(cod_faixa)  
    REFERENCES Faixa(cod_faixa),  
cod_compositor INT  
    FOREIGN KEY(cod_compositor)  
    REFERENCES Compositor(cod_compositor),
```

```
PRIMARY KEY(cod_faixa, cod_compositor)  
)
```

```
CREATE TABLE Género(  
genero VARCHAR(50) PRIMARY KEY  
)
```

```
CREATE TABLE Faixa_Género(  
cod_faixa INT  
    FOREIGN KEY(cod_faixa)  
    REFERENCES Faixa(cod_faixa),  
genero VARCHAR (50)  
    FOREIGN KEY(genero)  
    REFERENCES Género(genero)
```

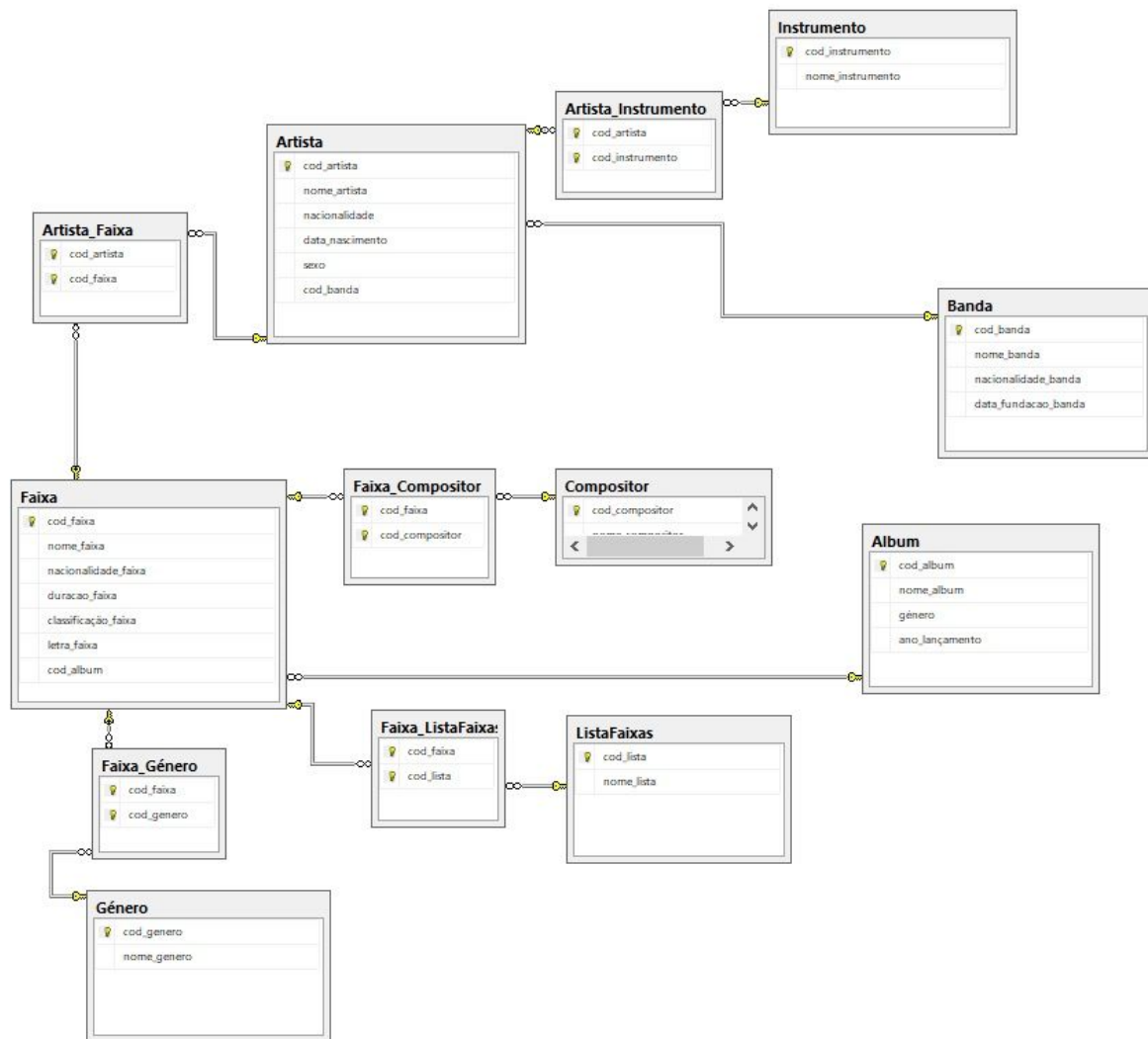
```
PRIMARY KEY(cod_faixa, genero)  
)
```

```
CREATE TABLE Instrumento(  
cod_instrumento INT PRIMARY KEY,  
nome_instrumento VARCHAR(50)  
)
```

```
CREATE TABLE Artista_Instrumento(  
cod_artista INT  
    FOREIGN KEY(cod_artista)  
    REFERENCES Artista(cod_artista),  
cod_instrumento INT  
    FOREIGN KEY(cod_instrumento)  
    REFERENCES Instrumento(cod_instrumento),
```

```
PRIMARY KEY(cod_artista, cod_instrumento)  
)
```

## Modelo Físico



Modelo Físico - Arquivo MP4

## Inserção de dados em SQL

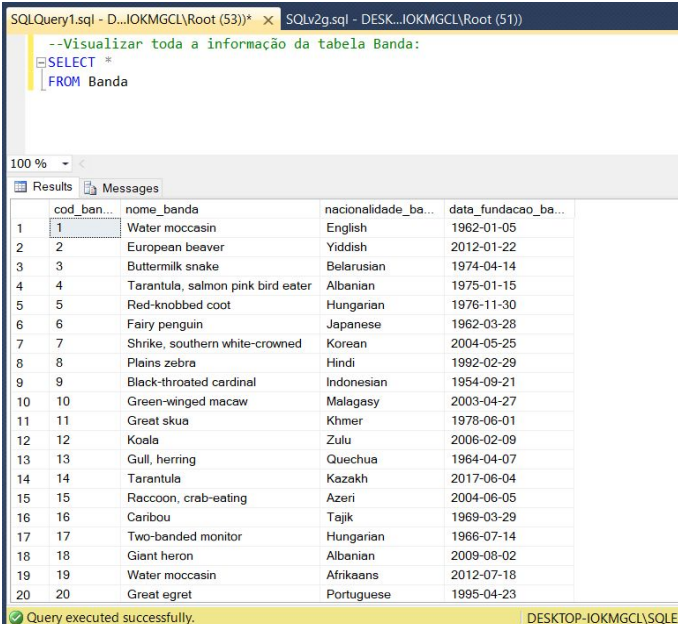
Devido à enorme extensão do código referente à inserção de dados nas tabelas, enviamos em anexo juntamente com o presente relatório, em formato de Query em SQL, pronto a ser observado diretamente no programa, incluindo também o código para a criação de tabelas.

## Consultas e subconsultas em SQL

Abaixo segue os comandos para as consultas e subconsultas que foram efetuadas, acompanhadas com um Screenshot do programa onde se podem verificar os resultados obtidos.

--Visualizar toda a informação da tabela Banda:

```
SELECT *  
FROM Banda
```



cod_ban...	nome_banda	nacionalidade_ba...	data_fundacao_ba...
1	Water moccasin	English	1962-01-05
2	European beaver	Yiddish	2012-01-22
3	Buttermilk snake	Belarusian	1974-04-14
4	Tarantula, salmon pink bird eater	Albanian	1975-01-15
5	Red-knobbed coot	Hungarian	1976-11-30
6	Fairy penguin	Japanese	1962-03-28
7	Shrike, southern white-crowned	Korean	2004-05-25
8	Plains zebra	Hindi	1992-02-29
9	Black-throated cardinal	Indonesian	1954-09-21
10	Green-winged macaw	Malagasy	2003-04-27
11	Great skua	Khmer	1978-06-01
12	Koala	Zulu	2006-02-09
13	Gull, herring	Quechua	1964-04-07
14	Tarantula	Kazakh	2017-06-04
15	Reccoon, crab-eating	Azeri	2004-06-05
16	Caribou	Tajik	1969-03-29
17	Two-banded monitor	Hungarian	1966-07-14
18	Giant heron	Albanian	2009-08-02
19	Water moccasin	Afrikaans	2012-07-18
20	Great egret	Portuguese	1995-04-23

--Visualizar colunas específicas da tabela Album:

```
SELECT nome_album, ano_lançamento  
FROM Album
```

SQLQuery1.sql - D:\IOMGCL\Root (53)) \* SQLv2g.sql - DESK\IOMGCL\Root (51))

```
--Visualizar colunas especificas da tabela Album:
SELECT nome_album, ano_lançamento
FROM Album
```

100 %

	nome_album	ano_lançame...
1	intangible	2005
2	exuding	1992
3	Persistent	1998
4	function	2004
5	structure	1996
6	composite	2009
7	orchestration	1997
8	matrix	2012
9	explicit	1990
10	website	2007
11	discrete	2006
12	Multi-tiered	2000
13	client-server	2006
14	Open-source	2004
15	groupware	1992
16	asynchronous	1993
17	contextually-based	2003
18	Self-enabling	2000
19	functionalities	2008
20	Profit-focused	1986

Query executed successfully. DESKTOP-IOMGCL\SQLEXPRESS ... DESKTOP-IOMGCL\Ro

--Visualizar os compositores existentes:

```
SELECT nome_compositor AS "Compositores"
FROM Compositor
```

SQLQuery1.sql - D:\IOMGCL\Root (53)) \* SQLv2g.sql - DESK\IOMGCL\Root (51))

```
--Visualizar os compositores existentes:
SELECT nome_compositor AS "Compositores"
FROM Compositor
```

100 %

	Compositores
1	Ferdy Kalinovich
2	Angelita Faircloth
3	Cassandra Gwynn
4	Kellen Saltern
5	Doris Jeanin
6	Maible Chalcraft
7	Esdras Broome
8	Ora Lambrook
9	Mord Shaudfurth
10	Zachariah Dyneley
11	Martino McGrowther
12	Pooh Cornelis
13	Kingston Hoble
14	Caprice Perone
15	Eb MacWilliam
16	Gerhardine Disbury
17	Sibby Zarfati
18	Becki Shimmin
19	Willdon Oleshunin
20	Saloma Tacon

Query executed successfully. DESKTOP-IOMGCL\SQLEXPRESS ... DESKTOP-IOMGCL\Ro

--Visualizar as faixas e albums correspondentes:

```
SELECT nome_faixa, nome_album
FROM Faixa INNER JOIN Album ON Faixa.cod_faixa = Album.cod_album
```

SQLQuery1.sql - D:\IOMGCL\Root (53)) \* x SQLv2g.sql - DESK...IOMGCL\Root (51))

```
--Visualizar as faixas e albums correspondentes:
SELECT nome_faixa, nome_album
FROM Faixa INNER JOIN Album ON Faixa.cod_faixa = Album.cod_album
```

100 %

Results Messages

	nome_faixa	nome_album
1	Girl of Your Dreams, The (Niña de tus ojos, La)	intangible
2	Bombardier	exuding
3	Still Life	Persistent
4	Alien from L.A.	function
5	Samson and Delilah	structure
6	Incident at Oglala	composite
7	End of the Game (Der Richter und sein Henker)	orchestration
8	Vacation from Marriage	matrix
9	Two Times Lotte	explicit
10	Beautiful Kate	website
11	Made in Britain	discrete
12	Canterville Ghost, The	Multi-tiered
13	Murphy's Romance	client-server
14	Ernest Rides Again	Open-source
15	Mad Dogs & Englishmen	groupware
16	In the Bedroom	asynchronous
17	Belles on Their Toes	contextually-based
18	Big Fan	Self-enabling
19	NULL	functionalities
20	Drats, Delmonagas	Profit-forward

Query executed successfully. DESKTOP-IOMGCL\SQLEXPRESS ... DESKTOP-IOMGCL\Root (53) master 00:00:00

--Visualizar os nomes dos 3 primeiros artistas existentes:

```
SELECT TOP (3) Artista.nome_artista
FROM Artista
```

```
--Visualizar os nomes dos 3 primeiros artistas existentes:
SELECT TOP (3) Artista.nome_artista
FROM Artista
```

100 %

Results Messages

	nome_artista
1	Ermengarde Scotchmore
2	Chrissie Rubberts
3	Danica Elen

--Saber quantas bandas existem:

```
SELECT Count(*) AS "Número de bandas"
FROM Banda
```

```
--Saber quantas bandas existem:
SELECT Count(*) AS "Número de bandas"
FROM Banda
```

100 %

Results Messages

	Número de bandas
1	501

--Visualizar o nome e a data de nascimento de todos os Artistas, do mais velho para o mais novo:

```
SELECT nome_artista, data_nascimento
FROM Artista
ORDER BY data_nascimento ASC
```

--Visualizar o nome e a data de nascimento de todos os Artistas, do mais velho para o mais novo:

```
SELECT nome_artista, data_nascimento
FROM Artista
ORDER BY data_nascimento ASC
```

30 %

Results Messages

	nome_artista	data_nascime...
1	Obie Rouby	1930-06-30
2	Erskine Yielding	1930-07-17
3	Chrissie Rubberts	1930-07-22
4	Corinne Marquet	1930-08-13
5	Zea Hardi	1930-09-15
6	Lucky Ormonde	1930-09-20
7	Delphinia Clough	1930-09-20
8	Seamus MacKimm	1930-09-25
9	Cody Torel	1930-09-28
10	Pavlov Franchioni	1930-09-30
11	Gertrude Freckingham	1930-11-12
12	Burg Fortesquieu	1930-12-05
13	Jereb Dunkerk	1930-12-20
14	Wilfrid Gibbonson	1931-01-13
15	Owen Brando	1931-01-25
16	Brunhilda Beevis	1931-02-01
17	Gillen Schimeschke	1931-04-22
18	Sabina Jenn	1931-05-02
19	Elyn Braidley	1931-06-10
20	Tiffany Firman	1931-06-17

Query executed successfully. DESKTOP-IOKMGCL\SQLEXPRESS ... DESKTOP-IOKMGCL\Root (53) master

--Visualizar informação sobre a faixa 7:

```
SELECT*
FROM Faixa
WHERE cod_faixa = 7
```

SQLQuery1.sql - D:\IOMGCL\Root (53)\* x SQLv2g.sql - DESK\IOMGCL\Root (51)

--Visualizar informação sobre a faixa 7:

```
SELECT*
FROM Faixa
WHERE cod_faixa = 7
```

100 %

Results Messages

	cod_fai...	nome_faixa	nacionalidade_f...	duracao_fa...	classificação_fa...	letra_faixa	cod_alb...
1	7	End of the Game (Der Richter und sein Henker)	Indonesian	5.2	9	rutrum ac lobortis vel	243

--Visualizar informação sobre as faixas 4-7:

```
SELECT*
FROM Faixa
WHERE cod_faixa BETWEEN 4 AND 7
```

SQLQuery1.sql - D:\IOMGCL\Root (53)) \* x SQLv2g.sql - DESK...IOMGCL\Root (51)

```
--Visualizar informação sobre as faixas 4-7:
SELECT*
FROM Faixa
WHERE cod_faixa BETWEEN 4 AND 7
```

100 %

Results Messages

	cod_fai...	nome_faixa	nacionalidade_f...	durecao_fa...	classificação_fa...	letra_faixa	cod_alb...
1	4	Alien from L.A.	Burmese	8.3	9	aliquam convallis nunc proin at turpis	76
2	5	Samson and Delilah	Gujarati	8	2	nibh quisque id justo sit amet sapien dignissim ...	207
3	6	Incident at Oglala	Georgian	3.9	7	condimentum id luctus nec molestie sed	107
4	7	End of the Game (Der Richter und sein Henker)	Indonesian	5.2	9	rutrum ac lobortis vel	243

--Visualizar todos os albúms cujo nome começa por P:

```
SELECT nome_album AS "Nome Album"
FROM Album
WHERE upper(nome_album) LIKE 'P%'
```

```
--Visualizar todos os albúms cujo nome começa por P:
SELECT nome_album AS "Nome Album"
FROM Album
WHERE upper(nome_album) LIKE 'P%'
```

100 %

Results Messages

	Nome Album
1	Persistent
2	Profit-focused
3	portal
4	product
5	Proactive
6	Pre-emptive
7	policy
8	protocol
9	Persistent
10	Phased
11	Profound
12	Public-key
13	projection
14	parallelism
15	Proactive
16	Profound
17	pricing structure
18	pricing structure
19	paradigm
20	productivity

Query executed successfully.

--Visualizar todos os artistas que não tenham a letra “u” no nome, ordenados por código:

```
SELECT nome_artista AS "Nome Artista"
FROM Artista
WHERE (nome_artista) NOT LIKE '%u%'
ORDER BY cod_artista ASC
```



```

SQLQuery1.sql - D:\IOMGCL\Root (53)* -> SQLv2g.sql - DESK_IOMGCL\Root (51)
--Visualizar todos os artistas que não tenham a letra "u" no nome, ordenados por código:
SELECT nome_artista AS "Nome Artista"
FROM Artista
WHERE (nome_artista) NOT LIKE '%u%'
ORDER BY cod_artista

```

100 %

Results Messages

	Nome Artista
1	Ermengarde Scotchmore
2	Danica Elen
3	Simmonds Coping
4	Lindsay Bingley
5	Glenn Beeres
6	Roseanna O'Sheils
7	Jodi Gimblett
8	Ardra Liepins
9	Amory Biggerstaff
10	Ivette De Few
11	Clayborne Critoph
12	Shawnee Hasney
13	Morris Donneely
14	Tiphani McAirt
15	Timothy Abram
16	Francine Thornevell
17	Bendite Seabert
18	Coral Bortolozzi
19	Javier Mcleen
20	Rosae Frammingham

Query executed successfully. DESKTOP-IOMGCL\SQLEXPRESS ... DESKTOP-IOMGCL\Root (53) master 00:00:00

--Visualizar os nomes e duração das faixas, da mais longa para a mais curta:

```

SELECT nome_faixa "Nome", duracao_faixa "Duração (min)"
FROM Faixa
ORDER BY duracao_faixa DESC

```

```

SQLQuery1.sql - D:\IOMGCL\Root (53)* -> SQLv2g.sql - DESK_IOMGCL\Root (51)
--Visualizar os nomes e duração das faixas, da mais longa para a mais curta:
SELECT nome_faixa "Nome", duracao_faixa "Duração (min)"
FROM Faixa
ORDER BY duracao_faixa DESC

```

100 %

Results Messages

	Nome	Duração (m...
1	Centerville Ghost, The	8.7
2	Midnight Movie	8.7
3	Leopard Son, The	8.7
4	Assisted Living	8.6
5	Inheritance, The (Arven)	8.6
6	Sharknado 2: The Second One	8.6
7	Vibes	8.6
8	Pinochet Case, The (Cas Pinochet, Le)	8.6
9	Shadow Riders, The	8.6
10	Treasure Island	8.6
11	Jude the Obscure	8.6
12	Jerky Boys, The	8.6
13	High Crimes	8.5
14	Sons of Katie Elder	8.5
15	Year of the Yao, The	8.5
16	Nightmare on Elm Street 5: The Dream Child, A	8.5
17	Bob le Flambeur	8.5
18	Parental Guidance	8.4
19	Numbers Station, The	8.4
20	White: The Making of the Curse	8.4

Query executed successfully. DESKTOP-IOMGCL\SQLEXPRESS ... DESKTOP-IOMGCL\Root (53) master 00:00:00 498 rows

--Visualizar quantos géneros diferentes existem

```

SELECT COUNT(DISTINCT nome_genero)
FROM Género

```

SQLQuery1.sql - D...IOKMGCL\Root (53))\* x SQLv2g.sql - DESK...IOKMGCL

```
--Visualizar quantos géneros diferentes existem
SELECT COUNT(DISTINCT nome_genero)
FROM Género
```

100 % <

Results Messages

	(No column na...
1	15

--Visualizar quantas artistas existem

SELECT COUNT(sexo) AS "Artistas femininas"

From Artista

WHERE upper(sexo) = 'Female'

SQLQuery1.sql - D...IOKMGCL\Root (53))\* x SQLv2g.sql - DESK...IOKMGCL\Root (51)

```
--Visualizar quantas artistas existem
SELECT COUNT(sexo) AS "Artistas femininas"
From Artista
WHERE upper(sexo) = 'Female'
```

100 % <

Results Messages

	Artistas feminin...
1	518

--Visualizar o número de artistas por sexo:

SELECT sexo AS "Sexo", COUNT(\*) AS "Nº"

FROM Artista

GROUP BY sexo

SQLQuery1.sql - D...IOKMGCL\Root (53))\* x SQLv2g.sql - DESK..

```
--Visualizar o número de artistas por sexo:
SELECT sexo AS "Sexo", COUNT(*) AS "Nº"
FROM Artista
GROUP BY sexo
```

100 % <

Results Messages

	Sexo	Nº
1	Male	482
2	Female	518

--Visualizar quantas faixas do album 291 existem por nome:

SELECT UPPER(nome\_faixa) "Nome", COUNT(\*) "Nº Faixas"

FROM Faixa

WHERE cod\_album = 291

The screenshot shows a SQL query window with the following text:

```
GROUP BY sexo  
  
--Visualizar quantas faixas do album 291 existem por nome:  
SELECT COUNT(nome_faixa) "Nº Faixas"  
FROM Faixa  
WHERE Faixa.cod_album = 291
```

Below the query, the 'Results' tab is active, displaying a table with one row:

	Nº Faixas
1	3

-- Visualizar as faixas com classificação maior que a faixa nº 30, ordenadas por código:

```
SELECT nome_faixa, AVG(classificação_faixa) AS "Classificação"  
FROM Faixa  
GROUP BY nome_faixa  
HAVING AVG(classificação_faixa) > (SELECT  
AVG(classificação_faixa)  
FROM Faixa  
WHERE cod_faixa = 30)
```

The screenshot shows a SQL query window with the following text:

```
-- Visualizar as faixas com classificação maior que a faixa nº 30, ordenadas por código::  
SELECT nome_faixa, AVG(classificação_faixa) AS "Classificação"  
FROM Faixa  
GROUP BY nome_faixa  
HAVING AVG(classificação_faixa) > (SELECT  
AVG(classificação_faixa)  
FROM Faixa  
WHERE cod_faixa = 30)
```

Below the query, the 'Results' tab is active, displaying a table with two columns: 'nome\_faixa' and 'Classificação'.

	nome_faixa	Classificação
1	3 Women (Three Women)	7
2	35 Up	9
3	911 in Plane Site	10
4	Against All Flags	10
5	Alien Contamination	9
6	Alien from L.A.	9
7	Alien Nation: Dark Horizon	7
8	Allnighter, The	6
9	And Soon the Darkness	7
10	Answer Man, The (a.k.a. Arlen Faber)	7
11	April in Paris	6
12	April Love	6
13	Ariel	8
14	Art of the Steal, The	9
15	Back to the Future Part III	8
16	Rev'l Family (Paha Parha)	6

At the bottom of the window, a status bar indicates: "Query executed successfully. DESKTOP-IOKMGCL\SQLEXPRESS ... DESKTOP-IOKMGCL\Root (53) master 00:00:00 2"

-- Visualizar o número de albuns que existem por ano de lançamento

```
SELECT ano_lançamento "Ano de lançamento", COUNT(*) "Quantidade"  
FROM Album  
WHERE ano_lançamento IS NOT NULL  
GROUP BY ano_lançamento
```

SQLQuery1.sql - D:\IOMGCL\Root (53))\* x SQLv2g.sql - DESK...IOMGCL\Root (51))

```
-- Visualizar o número de albuns que existem por ano de lançamento
SELECT ano_lançamento "Ano de lançamento", COUNT(*) "Quantidade"
FROM Album
WHERE ano_lançamento IS NOT NULL
GROUP BY ano_lançamento
```

100 %

Results Messages

Ano de lançamento...	Quantida...
1 1958	1
2 1962	2
3 1963	1
4 1964	1
5 1965	2
6 1967	3
7 1969	1
8 1974	1
9 1975	2
10 1976	2
11 1978	1
12 1980	2
13 1981	1
14 1983	3
15 1984	4
16 1985	4

Query executed successfully. DESKTOP-I

--Visualizar os artistas com as respectivas bandas:

```
SELECT Artista.nome_artista, Banda.nome_banda
FROM Banda INNER JOIN Artista ON Banda.cod_banda = Artista.cod_artista
```

SQLQuery1.sql - D:\IOMGCL\Root (53))\* x SQLv2g.sql - DESK...IOMGCL\Root (51))

```
--Visualizar os artistas com as respectivas bandas:
SELECT Artista.nome_artista, Banda.nome_banda
FROM Banda INNER JOIN Artista ON Banda.cod_banda = Artista.cod_artista
```

100 %

Results Messages

nome_artista	nome_banda
1 Ermengarde Scotchmore	Water moccasin
2 Chrissie Rubberts	European beaver
3 Danica Elen	Buttermilk snake
4 Ezequiel Danher	Tarantula, salmon pink bird eater
5 Lucila Atkinson	Red-knobbed coot
6 Simmonds Coping	Fairy penguin
7 Lindsay Bingley	Shrike, southern white-crowned
8 Glenn Beeres	Plains zebra
9 Roseanna O'Sheils	Black-throated cardinal
10 Jodi Gimblett	Green-winged macaw
11 Andra Liepins	Great skua
12 Amory Biggerstaff	Koola
13 Trula Sinnett	Gull, herring
14 Ivette De Few	Tarantula
15 Clayborne Critoph	Raccoon, crab-eating
16 Shawnee Hasenov	Caribou

Query executed successfully. DESKTOP-IOMGCL\SQL EXPRESS ... DESKTOP-IOMGCL\Root (53) master 00:00

--Visualizar o nº de elementos de cada banda:

```
SELECT Banda.nome_banda, COUNT(Banda.nome_banda) AS "Nº Elementos"
FROM Banda INNER JOIN Artista ON Banda.cod_banda = Artista.cod_artista
GROUP BY Banda.nome_banda
ORDER BY "Nº Elementos" DESC
```

```
--Visualizar o número de elementos de cada banda:
SELECT Banda.nome_banda, COUNT(Banda.nome_banda) AS "Nº Elementos"
FROM Banda INNER JOIN Artista ON Banda.cod_banda = Artista.cod_artista
GROUP BY Banda.nome_banda
ORDER BY "Nº Elementos" DESC
```

nome_banda	Nº Elementos
Bat, asian false vampire	3
Mallard	3
Roë deer	3
Red phalarope	2
Spider, wolf	2
Stork, greater adjutant	2
Tapir, brazilian	2
Tarantula	2
Paca	2
Barbet, crested	2
Rock dove	2
Turkey, wild	2
Turtle, eastern box	2
Unai	2
Violet-crested turaco	2
Vulture, white-rumped	2
Water moccasin	2
White-necked stork	2
Sourel, smith's bush	2

Query executed successfully.

-- Visualizar todos os artistas e as suas respetivas nacionalidades por ordem decrescente de código:

```
SELECT Artista.nome_artista "nome", Artista.nacionalidade "nacionalidade"
From Artista
ORDER BY Artista.cod_artista DESC
```

```
-- Visualizar todos os artistas e as suas respetivas nacionalidades ordenado por nome:
SELECT Artista.nacionalidade "nacionalidade", Artista.nome_artista "nome"
From Artista
ORDER BY Artista.nome_artista ASC
```

nacionalidade	nome
Swahili	Adamo Carrick
Hungarian	Addi Nayer
Gagauz	Addia Cornforth
Georgian	Addie Snewin
Romanian	Addie Garey
Icelandic	Addison Mattiuzzi
Irish Gaelic	Adelbert Munsey
Lithuanian	Adelo Lambrechts
Kazakh	Adham Doick
Icelandic	Adlai Blore
Kannada	Ado Allinson
Korean	Adolph McAleese
Spanish	Adriano Erratt
Quechua	Adriano Marten...
Assamese	Adriana Tout
Hiri Mohi	Anethe Rellam

Query executed successfully.

--Visualizar as faixas que estão na lista "Favoritos":

```
SELECT Faixa.nome_faixa
FROM Faixa INNER JOIN (Faixa_ListaFaixas INNER JOIN ListaFaixas ON
ListaFaixas.cod_lista = Faixa_ListaFaixas.cod_lista)
ON Faixa.cod_faixa = Faixa_ListaFaixas.cod_faixa
WHERE (((ListaFaixas.nome_lista) = 'Favoritos'))
```

```
--Quais as faixas que estão na lista "Favoritos"
SELECT Faixa.nome_faixa
FROM Faixa INNER JOIN (Faixa_ListaFaixas INNER JOIN ListaFaixas ON ListaFaixas.cod_lista = Faixa_ListaFaixas.cod_lista)
ON Faixa.cod_faixa = Faixa_ListaFaixas.cod_faixa
WHERE (((ListaFaixas.nome_lista) = 'Favoritos'))
```

100 %

Results Messages

	nome_faixa
1	Big Fan
2	Elle: A Modern Cinderella Tale
3	Born Reckless
4	Witless Protection
5	Joysticks
6	11'09"01 - September 11

--Visualizar a banda da qual o artista Slash pertence:

```
SELECT nome_artista, nome_banda
FROM Artista INNER JOIN Banda ON Artista.cod_banda = Banda.cod_banda
WHERE nome_artista = 'Slash'
```

```
--Visualizar a banda a que o artista Slash pertence:
SELECT nome_artista, nome_banda
FROM Artista INNER JOIN Banda ON Artista.cod_banda = Banda.cod_banda
WHERE nome_artista = 'Slash'
```

100 %

Results Messages

	nome_artista	nome_banda
1	Slash	Guns N Roses

--Visualizar os instrumentos que cada artista toca:

```
SELECT Artista.cod_artista, Artista.nome_artista, Instrumento.nome_instrumento
FROM Instrumento INNER JOIN (Artista INNER JOIN Artista_Instrumento
ON Artista.cod_artista = Artista_Instrumento.cod_artista)
ON Instrumento.cod_instrumento = Artista_Instrumento.cod_instrumento
```

```
--Visualizar os instrumento que cada artista toca:
SELECT Artista.cod_artista, Artista.nome_artista, Instrumento.nome_instrumento
FROM Instrumento INNER JOIN (Artista INNER JOIN Artista_Instrumento
ON Artista.cod_artista = Artista_Instrumento.cod_artista)
ON Instrumento.cod_instrumento = Artista_Instrumento.cod_instrumento
```

	cod_artista	nome_artista	nome_instrumento
1	1	Emengarde Scotchmore	piano
2	2	Chrissie Rubberts	flauta
3	2	Chrissie Rubberts	harpa
4	3	Danica Elen	saxofone
5	4	Ezequiel Danher	teclado
6	5	Lucila Atkinson	flauta
7	6	Simmonds Coping	bateria
8	7	Lindsay Bingley	teclado
9	8	Glenn Beeres	violino
10	9	Roseanna O'Sheils	saxofone
11	10	Jodi Gimblett	vocaís
12	11	Ardra Liepins	violino
13	12	Amory Biggerstaff	flauta
14	13	Trula Sinnett	violino
15	14	Ivette De Few	bateria
16	15	Clayborne Critoph	bateria
17	16	Shawnee Hasney	guitarra
18	17	Morris Donneely	saxofone

--Saber quais os artistas a solo (que não possuem banda) existentes

SELECT nome\_artista AS "Artistas a solo"

From Artista

WHERE cod\_banda IS NULL

```
--Saber quais os artistas a solo (que não possuem banda) existentes:
SELECT nome_artista AS "Artistas a solo"
From Artista
WHERE cod_banda IS NULL
```

	Artistas a solo
1	David Fonseca
2	Paul McCartney
3	Elvis Presley
4	Haslett Guest

--Visualizar quantos compositores possuem se chamam Lukas:

SELECT Count(Compositor.nome\_compositor) AS "Lukas"

FROM Compositor

WHERE Compositor.nome\_compositor LIKE 'Lukas%'



```

SQLQuery3.sql - D:\IOKMGCL\Root (54))* x SQLQuery1.sql - D:\IOKMGCL\Root (53))*
--Visualizar quantos compositores possuem se chamam Lukas:
SELECT Count(Compositor.nome_compositor) AS "Lukas"
FROM Compositor
WHERE Compositor.nome_compositor LIKE 'Lukas%'

```

100 %

Results Messages

	Lukas
1	3

--Visualizar o nome da música, os artistas que a interpretam e o seu compositor:

```

SELECT nome_faixa "Nome da musica", nome_artista "Nome Artista", nome_compositor
"Nome compositor"
FROM (((Artista INNER JOIN (Faixa INNER JOIN Artista_Faixa ON Artista_Faixa.cod_faixa
= Faixa.cod_faixa) ON Artista.cod_artista=Artista_faixa.cod_artista) INNER JOIN
Faixa_Compositor ON Faixa_Compositor.cod_faixa= Faixa.cod_faixa) INNER JOIN
Compositor ON Faixa_Compositor.cod_compositor= Compositor.cod_compositor)
WHERE nome_faixa = 'Santa Fe'

```

```

SQLQuery3.sql - D:\IOKMGCL\Root (54))* x SQLQuery1.sql - D:\IOKMGCL\Root (53))* SQLv2g.sql - DESK...IOKMGCL\Root (51))
--Visualizar nome da musica, artista e compositor/varias interpretações da mesma música
SELECT nome_faixa "Nome da musica", nome_artista "Nome Artista", nome_compositor "Nome compositor"
FROM (((Artista INNER JOIN (Faixa INNER JOIN Artista_Faixa ON Artista_Faixa.cod_faixa = Faixa.cod_faixa) ON Artista.cod_artista=Arti
WHERE nome_faixa = 'Santa Fe'

```

100 %

Results Messages

	Nome da musi...	Nome Artista	Nome composi...
1	Santa Fe	Cornall Whyte	Moises Egan
2	Santa Fe	Cello Keppin	Moises Egan
3	Santa Fe	Brok Blakeston	Moises Egan

Query executed successfully. DESKTOP-IOKMGCL\SQL EXPRESS ... DESKTOP-IOKMGCL\Root (54) MP4 00:00:00 3 rows

--Visualizar todas as músicas rock com acompanhamento de violino e nome do violinista:

```

SELECT Faixa.nome_faixa "nome", nome_artista "artista"
FROM Género INNER JOIN (Faixa_Género INNER JOIN (Faixa INNER JOIN (Artista_Faixa
INNER JOIN (Artista INNER JOIN (Artista_Instrumento INNER JOIN Instrumento ON
Artista_Instrumento.cod_instrumento=Instrumento.cod_instrumento) ON
Artista_Instrumento.cod_artista =Artista.cod_artista) ON

```



Artista\_Faixa.cod\_artista=Artista.cod\_artista) ON Faixa.cod\_faixa= Artista\_Faixa.cod\_faixa)  
 ON Faixa\_Gênero.cod\_faixa= Faixa.cod\_faixa) ON Gênero.cod\_genero=  
 Faixa\_Gênero.cod\_genero  
 WHERE ((Gênero.nome\_genero= 'rock') AND (Instrumento.nome\_instrumento= 'violino'))

The screenshot shows a SQL Server Enterprise Manager window with three tabs: SQLQuery3.sql, SQLQuery1.sql, and SQLv2g.sql. The active tab is SQLQuery1.sql, which contains the following SQL query:

```
--Todas as músicas rock da base de dados com acompanhamento de violino e nome do violinista
SELECT Faixa.nome_faixa "nome",nome_artista "artista"
FROM Gênero INNER JOIN (Faixa_Gênero INNER JOIN (Faixa INNER JOIN(Artista_Faixa INNER JOIN (Artista INNER JOIN(Artista_Instrumento I
WHERE ((Gênero.nome_genero= 'rock') AND (Instrumento.nome_instrumento= 'violino'))
```

The query results are displayed in a table with two columns: nome and artista. The results are as follows:

	nome	artista
1	Fireflies in the Garden	Keelby Aldcorn
2	On Guard (Bossu, Le)	Bebe Glancy
3	Ernest Goes to Jail	Gaven Le Brum
4	Jack Brooks: Monster Slayer	Jarid Curmi
5	I Could Never Be Your Woman	Christoph Soane
6	Murder by Proxy: How America Went Postal	Ebba Hairyes

The status bar at the bottom indicates that the query was executed successfully and returned 6 rows.

--Todas as músicas do gênero rap e artistas que fazem rap

SELECT Faixa.nome\_faixa "nome",nome\_artista "artista"  
 FROM Gênero INNER JOIN (Faixa\_Gênero INNER JOIN (Faixa INNER JOIN(Artista\_Faixa  
 INNER JOIN Artista ON Artista\_Faixa.cod\_artista=Artista.cod\_artista) ON Faixa.cod\_faixa=  
 Artista\_Faixa.cod\_faixa) ON Faixa\_Gênero.cod\_faixa= Faixa.cod\_faixa) ON  
 Gênero.cod\_genero= Faixa\_Gênero.cod\_genero  
 WHERE (Gênero.nome\_genero= 'rap')

with slash.sql - D:\AMAZING\César (53))" X SQLQuery3.sql - D:\AMAZING\César (52)

```
--Todas as músicas do gênero rap e artistas que fazem rap
SELECT Faixa.nome_faixa "nome", nome_artista "artista"
FROM Gênero INNER JOIN (Faixa_Gênero INNER JOIN (Faixa INNER JOIN(Artista_Faixa INNER JOIN Artista ON Artista_Faixa.cod_artista=Arti
WHERE (Gênero.nome_gênero= 'rap')
```

100 %

Results Messages

	nome	artista
8	Some Girl(s)	Madelin Tolomei
9	Ten Commandments, The	Shelden Trimby
10	Drained (O cheiro do Ralo)	Tynan Hurl
11	Some Girl(s)	Fanchon Bushell
12	Bastard Out of Carolina	Kirsten Adrian
13	Faith Like Potatoes	Ashien Bartolomucci
14	Roads to Koktebel (Koktebel)	Meaghan Gottelier
15	Numbers Station, The	Merla Topaz
16	Claire Dolan	Merla Topaz
17	Underground Comedy Movie....	Merla Topaz
18	Great White Hope, The	Westbrook Postgate
19	Skokie	Andie Sneath
20	Escape from Planet Earth	Solomon McCrackan
21	Fist of the North Star	Vasily Blune
22	Joysticks	Donta Clabbum
23	Faith Like Potatoes	Gaven Le Brum
24	Shooting Dogs (a.k.a. Beyon...	Alameda Mazzei
25	Underground Comedy Movie....	Lyn Allatt
26	Drained (O cheiro do Ralo)	Erinna Galbraith
27	Slim Carter	Jody Castlake
28	Ballad of Ramblin' Jack, The	Stavros oldev

Query executed successfully.

DAMAZING\SQLXPRESS (12.0 RTM) | DAMAZING\César (53)

## Access

Após a criação da base de dados no SQL Server Management Studio, passou-se para o Access. Foram criadas todas as tabelas:

dbo_Album			
cod_album	nome_album	gênero	ano_lançam
156	budgetary mar	blues	1988
157	moderator	punk	1997
158	exuding	eletronic	2009
159	contextually-b	hip-hop	1993
160	Optional	latino	2005
161	leading edge	rap	1997
162	Organic	alternative	2002
163	moratorium	rap	2000
164	Inverse	classical	1992
165	zero defect	rock	2000
166	Optional	metal	2006
167	middleware	rock	2011
168	multi-tasking	hip-hop	2008
169	national	classical	1969
170	infrastructure	jazz	2010
171	discrete	indie	2009
172	superstructure	rock	2010
173	foreground	pop	2006
174	Mandatory	punk	2009
175	Assimilated	latino	1995
176	grid-enabled	pop	2000
177	parallelism	indie	2007
178	zero administr	eletronic	1999
179	middleware	indie	2003
180	24/7	hip-hop	2006
181	installation	latino	2003

Tabela Album

dbo_Artista					
cod_artista	nome_artista	nacionalidade	data_nascim	sexo	cod_banda
981	Wilbur Treby	Dhivehi	1958-02-02	Male	200
982	Gabi Sines	Tsonga	1966-11-03	Male	150
983	Sheridan Wear	Kashmiri	1945-04-25	Male	313
984	Brok Blakestor	Yiddish	1979-01-16	Male	368
985	Fabio Grier	Greek	1967-09-03	Male	292
986	Chas Glashby	Lithuanian	1976-08-14	Male	327
987	Gabriella Pogo	Croatian	1940-08-05	Female	226
988	Hillary Stuchbe	Tswana	1968-02-29	Female	247
989	Angelita Jayun	Indonesian	1976-11-11	Female	200
990	Owen Brando	Portuguese	1931-01-25	Male	488
991	Anthiathia Bis	Malay	1975-09-17	Female	43
992	Brit Pinching	Somali	1966-08-19	Male	373
993	Preston Carlet	Punjabi	1991-05-02	Male	237
994	Mahmoud Pas	Somali	1962-10-10	Male	370
995	Bennie Tootal	Northern Soth	1935-02-04	Male	469
996	Rosella Barbisi	Kyrgyz	1993-07-25	Female	373
997	Jermayne Illin	Khmer	1946-09-09	Male	210
998	Nona Boase	Swedish	1988-01-04	Female	134
999	Adamo Carrick	Swahili	1947-10-05	Male	242
1000	Haslett Guest	Ndebele	1954-04-03	Male	442
1001	David Fonseca	Portuguese	1973-02-21	Male	
1002	Paul McCartne	English	1942-06-18	Male	
1003	Elvis Presley	American	1935-01-08	Male	
1004	Haslett Guest	Canadian	1959-10-04	Male	
1005	Slash	Canadian	1965-06-23	Male	501

Tabela Artista

cod_artista	cod_faixa
1	1
1	407
2	2
2	140
3	3
4	4
4	440
5	5
6	6
6	358
7	7
7	170
7	416
8	8
8	324
8	382
9	9
10	10
10	250
11	11
12	12
12	96
13	13
14	14
14	53

dbo_Artista_Instrumento	
cod_artista	cod_instrum
76	5
77	4
78	4
79	5
80	8
81	2
82	9
83	1
84	5
85	4
86	6
87	9
88	7
89	4
90	5
91	9
92	8
93	5
94	4
95	7
96	9
97	4
98	8
99	1
100	2

dbo_Banda			
cod_banda	nome_band	nacionalida	data_fundac
477	Japanese maca	Georgian	1982-08-31
478	Water monitor	Maori	2012-08-12
479	Legaan, groun	Bulgarian	2008-12-30
480	Glossy starling	Gagauz	1983-06-17
481	Nuthatch, red-	Latvian	1991-09-02
482	Gnu, brindled	Hebrew	1993-01-21
483	Owl, great hor	Arabic	1953-06-11
484	Hummingbird	Persian	1950-09-24
485	Roller, lilac-br	Kazakh	1959-10-13
486	Stork, white	Croatian	1979-06-18
487	Royal tern	Catalan	1995-01-26
488	Porcupine, cre	Georgian	1998-04-20
489	Flying fox (uni	Tok Pisin	2012-06-17
490	Deer, mule	Assamese	1970-01-01
491	Little brown de	Montenegrin	1954-10-13
492	European beav	Punjabi	2012-01-11
493	Crested buntir	Belarusian	1969-01-07
494	Owl, australiar	Guarani	1973-01-02
495	Partridge, coq	Spanish	1960-08-19
496	Dolphin, comm	Bosnian	1965-11-21
497	Bat, little brow	Macedonian	2014-11-07
498	Caracal	Italian	2006-06-26
499	Lemming, coll	Tok Pisin	1983-08-09
500	Cormorant, jav	Kannada	1966-09-04
501	Guns N Roses	American	1985-09-04

Tabelas Artista\_Faixa,Artista\_Instrumento e Banda

dbo_Faixa_Compositor	
cod_faixa	cod_compo
1	477
2	64
2	416
2	482
3	179
4	268
4	425
5	126
5	166
5	349
5	481
6	338
6	432
6	454
7	350
8	456
9	62
9	71
9	115
9	215
10	410
11	154
11	238
12	76
12	98
13	264
Registo: 1 de 999	

dbo_Compositor	
cod_compositor	nome_compositor
76	Megen Sertin
77	Debbi Lernihan
78	Gina Billett
79	Willyt McRae
80	Sheri Giovannardi
81	Zandra Lafontaine
82	Darrin Artin
83	Jauelin Sapshed
84	Brewster Gymblett
85	Jerrie Garrique
86	Sandi Tooher
87	Merrel Mahony
88	Frederic Greiswood
89	Carmina Ghidoli
90	Guntar Coleford
91	Yetty Redmell
92	Austin Dicey
93	Rhianna Gagie
94	Ring O'Mullaney
95	Opaline Ratazzi
96	Perle Spurryer
97	Archy Warcup
98	Jimmy Simonini
99	Artemis Chaney
100	Abelard Sor
101	Lynett Curcude
Registo: 7 de 500	

dbo_Faixa						
cod_faixa	nome_faixa	nacionalidad	duracao_fai	classificacão	letra_faixa	cod_album
215	One Crazy Summer	Dutch	4,4	3	condimentum curabitu	18
216	Myth, The (San wa)	Portuguese	3,5	9	augue aliquam erat vol	208
217	Werner - Gekotzt wird sp	Hungarian	6,5	7	varius integer ac leo pe	21
218	Forgiving Dr. Mengele	Arabic	5	5	elementum pellentesq	349
219	National Treasure	Tswana	1,6	6	justo nec condimentum	280
220	\$ (Dollars)	Montenegrin	2,3	2	felis fusce posuere feli	37
221	Werewolf Boy, A (Neuk-c	Catalan	2,9	5	nulla nunc purus phase	140
222	Rosetta	Latvian	3,1	5	natoque penatibus et n	370
223	Quartet	Bengali	2,7	5	neque dui bibendum i	99
224	But I'm a Cheerleader	Estonian	6,4	6	erat nulla tempus vivar	301
225	Sinners and Saints	Kazakh	0,7	4	enim in tempor turpis r	444
226	Shadow Riders, The	Hiri Motu	8,6	4	nulla elit ac nulla sed vi	391
227	Dimples	Catalan	1,9	4	aliquam convallis nunc	412
228	Red Cliff Part II (Chi Bi Xi	Hiri Motu	2,2	1	libero quis orci nullam	312
229	Children Shouldn't Play v	Norwegian	3,9	9	at ipsum ac tellus semp	163
230	Shampoo	Kashmiri	7	10	elementum in hac habi	151
231	Colonel Redl (Oberst Rec	Ndebele	6,2	9	cubilia curae donec phe	8
232	Step Up 3D	Luxembourgish	5	3	sapien cursus vestibulu	67
233	Into the Woods	Khmer	2,3	3	vestibulum ante ipsum	119
234	Castle of Clouds, The (Pil	German	1,1	10	faucibus orci luctus et u	453
235	Eraser	Tswana	4,9	4	sed accumsan felis ut a	141
236	One Flew Over the Cuck	Georgian	2,7	6	libero convallis eget eli	268
237	Bronx Tale, A	Amharic	5,5	4	justo aliquam quis turp	3
238	Elsa & Fred	Dhivehi	1,3	2	ligula in lacus curabitur	132
239	Ronja Robbersdaughter (	Swati	7,7	5	morbi porttitor lorem ii	299
240	Realms of Love, The (Wic	Swati	4,2	2	ac nulla sed vulnisi	274
Registo: 222 de 499						

Tabelas Faixa; Faixa\_Compositor; Compositor



dbo_Faixa_Gênero	dbo_Faixa_ListaFaixas	dbo_Gênero
cod_faixa	cod_faixa	cod_genero
1	106	1 rock
2	107	2 pop
3	108	3 rap
4	109	4 country
5	110	5 classical
6	111	6 latino
7	112	7 jazz
8	113	8 blues
9	114	9 hip-hop
10	115	10 indie
11	116	11 alternative
12	117	12 fado
13	118	13 eletronic
14	119	14 metal
15	120	15 punk
16	121	
17	122	
18	123	
19	124	
20	125	
21	126	
22	127	
23	128	
24	129	
25	130	
	131	

dbo_Instrumento	dbo_ListaFaixas
cod_instrumento	cod_lista
1	1 Alpha
2	2 Trippledex
3	3 Alphazap
4	4 Sonair
5	5 Transcof
6	6 Zaam-Dox
7	7 Ventosanzap
8	8 Regrant
9	9 Overhold
10	10 Veribet
	11 Cookley
	12 Tampflex
	13 Rank
	14 Flexidy
	15 Flexidy
	16 Solarbreeze
	17 Overhold
	18 Favoritos
	19 Zaam-Dox
	20 Greenlam
	21 Cardguard
	22 Zamit
	23 Rank
	24 Lotlux
	25 Temp
	26 Alpha

Tabelas Faixa\_Gênero; Faixa\_ListaFaixas; Gênero; Instrumento; ListaFaixas

## Formulários

Foram criados formulários em Access para todas as tabelas:

The image displays four screenshots of Microsoft Access forms, arranged in a 2x2 grid. Each form is designed for data entry into a specific table.

- Formulário Album:** Contains fields for `cod_album` (value: 48), `nome_album` (value: Mandatory), `gênero` (value: eletronic), and `ano_lançamento` (value: 1967). The status bar shows "Registro: 48 de 500".
- Formulário Artista:** Contains fields for `cod_artista` (value: 1005), `nome_artista` (value: Slash), `nacionalidade` (value: Canadian), `data_nascimento` (value: 1965-06-23), `sexo` (value: Male), and `cod_banda` (value: 501). The status bar shows "Registro: 1004 de 1004".
- Formulário Artista\_Faixa:** Contains fields for `cod_artista` (value: 995) and `cod_faixa` (value: 178). The status bar shows "Registro: 1485 de 1489".
- Formulário Banda:** Contains fields for `cod_banda` (value: 501), `nome_banda` (value: Guns N Roses), `nacionalidade_banda` (value: American), and `data_fundacao_banda` (value: 1985-09-04). The status bar shows "Registro: 500 de 500".

Formulários Album; Artista; Artista\_Faixa; Banda

Formulário Faixa

cod\_faixa

7

nome\_faixa

End of the Game (Der Richter und sein I

nacionalidade\_faixa

Indonesian

duracao\_faixa

5,2

classificação\_faixa

9

letra\_faixa

rutrum ac lobortis vel

cod\_album

243

Registo: 7 de 499

Sem Filtro

Procurar

Formulário Faixa\_Compositor

cod\_faixa

500

cod\_compositor

487

Formulário Faixa\_Gênero

cod\_faixa

1

cod\_genero

14

Formulário Faixa\_ListaFaixas

cod\_faixa

4

cod\_lista

4

Formulários Faixa;Faixa\_Compositor; Faixa\_Gênero; Faixa\_ListaFaixas



Formulário Género

Formulário Género

cod\_genero

nome\_genero

rock

Formulário Instrumento

Formulário Instrumento

cod\_instrumento

nome\_instrumento

vocais

Formulário ListaFaixas

Formulário ListaFaixas

cod\_lista

nome\_lista

Transcof

Formulários Género; Instrumento; ListaFaixas

## Relatórios

Foram elaborados alguns exemplos de relatórios em Access:

### Relatório Album

cod_album	nome_album	gênero	ano_lançamento
1	intangible	jazz	2005
2	exuding	alternative	1992
3	Persistent	country	1998
4	function	jazz	2004
5	structure	rap	1996
6	composite	punk	2009
7	orchestration	alternative	1997
8	matrix	fado	2012
9	explicit	eletronic	1990
10	website	jazz	2007
11	discrete	alternative	2006
12	Multi-tiered	alternative	2000
13	client-server	country	2006
14	Open-source	blues	2004
15	groupware	indie	1992

### Relatório Banda

cod_banda	nome_banda	nacionalidade_banda
2	European beaver	Yiddish
3	Buttermilk snake	Belarusian
4	Tarantula, salmon pink bird eater	Albanian
5	Red-knobbed coot	Hungarian
6	Fairy penguin	Japanese
7	Shrike, southern white-crowned	Korean
8	Plains zebra	Hindi
9	Black-throated cardinal	Indonesian
10	Green-winged macaw	Malagasy
11	Great skua	Khmer
12	Koala	Zulu
13	Gull, herring	Quechua
14	Tarantula	Kazakh
15	Raccoon, crab-eating	Azeri

### Relatórios Album e Banda

## Relatório Faixa

nome_faixa	classificação_faixa
Girl of Your Dreams, The (Niña de tus oj	3
Bombardier	5
Still Life	8
Alien from L.A.	9
Samson and Delilah	2
Incident at Oglala	7
End of the Game (Der Richter und sein I	9
Vacation from Marriage	2
Two Times Lotte	8
Beautiful Kate	8
Made in Britain	3
Canterville Ghost, The	2
Murphy's Romance	5
Ernest Rides Again	2

## Relatório Género

cod\_genero nome\_genero

1 rock

2 pop

3 rap

4 country

5 classical

6 latin

7 jazz

8 blues

9 hip-hop

10 indie

11 alternative

12 folk

13 electronic

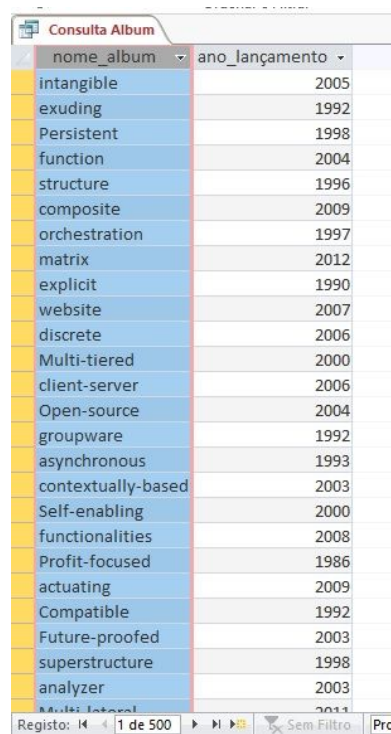
14 metal

15 punk

## Relatórios Faixa e Género

## Consultas em Access

Foram efetuadas algumas consultas em Access:



The screenshot shows an Access query window titled 'Consulta Album'. It displays a table with two columns: 'nome\_album' and 'ano\_lançamento'. The table contains 25 rows of data, alternating between yellow and light blue background colors. The bottom status bar indicates 'Registo: 1 de 500' and 'Sem Filtro'.

nome_album	ano_lançamento
intangible	2005
exuding	1992
Persistent	1998
function	2004
structure	1996
composite	2009
orchestration	1997
matrix	2012
explicit	1990
website	2007
discrete	2006
Multi-tiered	2000
client-server	2006
Open-source	2004
groupware	1992
asynchronous	1993
contextually-based	2003
Self-enabling	2000
functionalities	2008
Profit-focused	1986
actuating	2009
Compatible	1992
Future-proofed	2003
superstructure	1998
analyzer	2003
Multi-Inte...	2011

Exemplo consulta Access - Visualizar informação acerca do nome e ano de lançamento dos albuns

## Conclusão

Podemos confirmar que este projecto se tornou útil e permitiu que adquirísse-mos conceitos, informação e técnicas até então desconhecidas.

O modelo conceptual revelou ser a maior dificuldade deste trabalho (assim como a normalização do mesmo), sofreu inúmeras alterações até ao produto final, e admitimos que este modelo conceptual ainda pode ser passível de ser melhorado num futuro próspero.

É também de realçar a introdução dos dados, alguns reais que foram inseridos por nós mesmos, e outros “fictícios” (através de Data Generators), devido à enorme complexidade da base de dados relativamente a algumas entidades.

Na nossa visão pessoal deste trabalho, achámos que foi muito interessante, e salientamos que apesar de ser de o nosso conhecimento ser de um nível médio, e até dizendo que o resultado final deste trabalho se encontra também a esse nível (pois através do Access foi possível verificar erros prévios), podemos claramente afirmar que sabemos (e podemos) executar as tarefas relativas a criação de bases de dados de um modo mais eficaz, talvez até num futuro próximo.

## **Bibliografia**

<http://cms.ipbeja.pt/course/view.php?id=214>

<https://www.draw.io/>

<https://www.w3schools.com/sql/>