

Base de Dados

IMDb

Filmes de Comédia (2005-2023)



Realizado por Grupo D:

up202001845@edu.lettras.up.pt ¹

up202009437@edu.lettras.up.pt ²

up202008196@edu.lettras.up.pt ³

Fábio Alexandre Rodrigues Nunes ¹

Marta Garranas do Novo Pinhal ²

Tiago Miguel Carvalho Vieira ³

Bases de Dados

Licenciatura em Ciência da Informação

Faculdade de Engenharia da Universidade do Porto

2022/2023

Índice

1	Introdução	5
2	Contexto e Objetivo	6
3	Metodologia e Ferramentas	7
4	Modelo de Classes	9
5	Esquema Relacional	11
6	Instâncias	15
7	Interrogações	21
7.1	Interrogações simples	21
7.2	Interrogações de média dificuldade.....	24
7.3	Interrogações complexas	28
8	SGBD	30
8.1	Formulários (Forms)	30
8.2	Relatórios (Reports)	35
9	Conclusão	39
10	Referências	40
	Anexos	41
	Interrogações.....	41

Índice de Figuras

Figura 1 – Modelo de Classes	9
Figura 2 – Esquema Relacional	11
Figura 3 – Esquema Relacional (Acess)	13
Figura 4 – Instância da tabela 'Category'	15
Figura 5 – Instância da tabela 'Crew'	16
Figura 6 – Instância da tabela 'Movie'	17
Figura 7 – Instância da tabela 'Person'	18
Figura 8 – Instância da tabela 'Principals'	19
Figura 9 – Instância da tabela 'Rating'	20
Figura 10 – Forms 'Menu'	30
Figura 11 – Forms 'Elenco'	31
Figura 12 – Forms 'Pesquisa de Elenco por Filme'	32
Figura 13 – Forms 'Informações sobre pessoas'	33
Figura 14 – Forms 'FAQ'	34
Figura 15 – Relatório 'Idade das Pessoas Envolvidas'	35
Figura 16 – Relatório 'Nr de Nascimentos por ano'	36
Figura 17 – Relatório 'Rating dos Filmes'	37
Figura 18 – Relatório 'Top 245 Filmes'	38
Figura 19 – Query 1 (Nome_pessoas_envolvidas_filmes)	41
Figura 20 – Query 2 (Idade_Falecidos)	42
Figura 21 – Query 3 (Person_profession)	42
Figura 22 – Query 4 (Filme_Mais_Minutos)	43
Figura 23 – Query 5 (Filme_Menos_Minutos)	43
Figura 24 – Query 6 (Rating_Filme)	44
Figura 25 – Query 7 (TOP245_Filmes)	45
Figura 26 – Query 8 (Media_rating_filmes)	45
Figura 27 – Query 9 (MilaKunis_Ano_Nascimento+Idade)	46
Figura 28 – Query 10 (Filmes_MilaKunis)	46
Figura 29 – Query 11 (Filme_ScaryMovie_Pessoas)	47
Figura 30 – Query 12 (Ano_mais_filmes)	47
Figura 31 – Query 13 (Filmes_José_Valdebenito_diretor)	48
Figura 32 – Query 14 (N_personagens_cada_filme)	48
Figura 33 – Query 15 (Filme_maior_n_votos)	49

Figura 34 – Query 16 (Filme_maior_rating).....	49
Figura 35- Query 17 (Filme_menor_rating).....	49
Figura 36 – Query 18 (Idade_Pessoas_Vivas)	50
Figura 37 – Query 19 (Ator_+velho)	50
Figura 38 – Query 20 (Produtor_c+_filmes)	51
Figura 39 – Query 21 (Ator_ c+_filmes).....	51
Figura 40 – Query 22 (Filme_Rating>7,2_filmes_depois_2006_+55000votos)	52
Figura 41- Query 23 (NrNascimento_por_ano).....	52
Figura 42- Query 24 (Papel_Elenco)	53
Figura 43 – Query 25 (Nome_filme+peessoa_em_nome_personagem_Beauty).....	53
Figura 44- Query 26 (People_Idades)	54
Figura 45 – Query 27 (5_anos_melhor_média_pontuação_users_entre_2006_2022).....	55
Figura 46- Query 28 (Idade_ator+64).....	56

1 Introdução

No âmbito da unidade curricular de Bases de Dados, que se enquadra no 2º Semestre do 3º Ano da Licenciatura em Ciência da Informação, da Faculdade de Letras e Faculdade de Engenharia, realizamos este relatório, que diz respeito ao trabalho final desta unidade curricular. Posto isto foi escolhido um conjunto de dados (dataset), pelo que o grupo optou pelo da plataforma online do IMDb, por forma a criar um exemplar de uma base de dados para o conjunto escolhido (filmes de comédia entre os anos 2005 a 2023, que consta com um total de 1203 dados), utilizando a ferramenta Microsoft Access.

O principal objetivo do trabalho é a criação e apresentação da criação uma base de dados, perguntas e análise, para tal foram usadas ferramentas, metodologias e recursos.

Este relatório encontra-se dividido da seguinte forma: [Contexto e Objetivo](#) do trabalho realizado, seguido pela exposição da [Metodologia e Ferramentas](#) utilizadas, onde são mostrados passos de obtenção e normalização dos dados. De seguida, são apresentados o [Modelo de Classes](#) (UML) e o [Esquema Relacional](#) desenvolvidos no Draw.io e no Microsoft Access, juntamente com a descrição detalhada de cada um deles. Na próxima secção do relatório, apresentamos as [Instâncias](#) introduzidas no Access, seguida pelas [Interrogações](#) feitas à base de dados. Por fim, mostramos a utilização do Sistema de Gestão de Base de Dados ([SGBD](#)).

Para o presente trabalho, utilizamos o dataset disponibilizado pelo IMDb, no entanto, é importante notar que existiram algumas complicações com a utilização desses dados, de modo a não comprometer a integridade dos mesmos, o que, por sua parte dificultou, mas não impediu a realização do trabalho.

2 Contexto e Objetivo

O IMDb é uma base de dados online que abrange informações e estatísticas sobre filmes, programas televisivos e vídeo jogos, bem como dados de atores, diretores e outros profissionais da indústria cinematográfica. Tais informações podem compreender listas de elenco e equipa técnica, datas de estreia e dados de bilheteria, sinopses de enredo, trailers, biografias de atores e diretores, e outras curiosidades pertinentes (Lavery, 2017).

O grupo realizou o trabalho com base em 6 tabelas do IMDb presentes no website (IMDb, 2023) contêm informação variada sobre os objetos audiovisuais, bem como ratings e informação sobre as pessoas envolvidas na produção e realização dos filmes. Estas tabelas contêm milhares de linhas, havendo um excesso de dados para normalizar e importar para o Microsoft Access foi realizada uma filtração de dados por ano de 2005 a 2023, bem como de género de filme, sendo que o grupo se focou nos filmes de comédia.

Uma vez que certas tabelas podem ser de difícil leitura devido à utilização de IDs de filmes ou de pessoas em vez dos seus nomes correspondentes, optou-se por trabalhar nesses dados. O objetivo é desenvolver interrogações e formulários que auxiliem o utilizador na realização de pesquisas. O trabalho visa, assim, facilitar o acesso à informação no domínio do cinema, através da criação e desenvolvimento de várias expressões em SQL, bem como de uma interface adequada.

O grupo decidiu selecionar o website do IMDb como fonte de dados, uma vez que este é um dos melhores sites de classificação cinematográfica disponíveis na internet. A escolha do tema decorreu do fato de ser de interesse comum para a maioria da população, uma vez que as pessoas quando tem tempo livre em casa passam algum do seu tempo a relaxar e ver filmes. Assim, é cada vez mais importante facilitar a escolha de visualização de filmes, seja devido à presença de um determinado ator no elenco, à duração, etc.

Em suma, o objetivo principal do trabalho é viabilizar a recuperação de informações relativas à matéria audiovisual e seus respetivos elencos, a fim de fomentar pesquisas associadas à cultura cinematográfica.

3 Metodologia e Ferramentas

As tabelas mencionadas do IMDb consistem em 7 com os seguintes nomes originais: 'name.basics', 'title.akas', 'title.basics', 'title.crew', 'title.episode', 'title.principals' e 'title.ratings', todas em formato ficheiro (.tsv) (IMDb, 2023).

Contudo para o trabalho foram apenas usadas 6 tabelas com os seguintes nomes: "Movie", "Crew", "Principals", "Ratings", "Person" e "Category". Sendo que a tabela category estava inserida nos principals no IMDb, mas o grupo achou melhor separar e criar uma tabela à parte.

As tabelas excluídas devido a um extenso número de dados e também devido ao ficheiro ser demasiado pesado para realização da conversão do mesmo foi a tabela 'title.akas', quanto à tabela 'title.episode' foi excluída pelo grupo visto que foi restringido o tipo audiovisual apenas para filme.

Tendo em conta as tabelas escolhidas mencionadas acima também dispõem de um excesso de dados, assim para ser possível a importação destes para o Microsoft Access o grupo teve de arranjar forma de filtrar as tabelas. Dado a dimensão destas a forma mais eficaz que o grupo encontrou foi importar as tabelas para outra ferramenta da Microsoft, ou seja, o PowerBI, bem como o Microsoft Excel (sendo usada a funcionalidade: Power Query) desta forma foi possível trabalhar os dados, reduzindo-os no período de tempo (ano) entre 2005 a 2023, por género sendo este apenas Comédia (comedy) e tipo de objeto audiovisual como filme.

No Power BI para a realização da filtragem, o grupo optou por reduzir os dados na tabela 'Movie' e, em seguida, através dos IDs (tanto de título como de pessoas) foi utilizada a intercalação das consultas e os INNER JOINS, e com isto comparou as informações das tabelas com esta, filtrando, assim, os dados de igual forma.

Quanto à passagem do PowerBI para Excel o número de linhas com dados era o seguinte em cada tabela:

- Movies – 1203;
- Principals – 11 196;
- Crew – 1203;
- Ratings – 1097;
- Person – 11 196;
- Category – 1 012 933.

Depois de efetuada a transformação, as tabelas foram copiadas para o Microsoft Excel, onde se procedeu à respetiva modificação, de forma a torná-las compatíveis com a plataforma de destino. Esta modificação teve por base o modelo de classes e o esquema relacional previamente criados, os quais serão descritos de seguida.

Após a utilização da funcionalidade Power Query do Microsoft Excel e de termos efetuado a eliminação dos dados duplicados, o número de linhas com dados era o seguinte em cada tabela:

- Movies – 1203;
- Principals – 11 196;
- Crew – 1203;
- Ratings – 1097;
- Person – 10 472;
- Category – 10 472.

Após a conclusão do procedimento descrito anteriormente, foi possível importar as tabelas para o Access e criar o esquema pretendido nesta ferramenta. A partir desse ponto, o grupo trabalhou os dados e construiu o restante da base de dados de acordo com o planeado inicialmente. A maioria dos dados está na forma de "Texto Breve" e "Número Grande", pois esses foram os tipos que permitiram manter a integridade dos dados exportados.

4 Modelo de Classes

O grupo realizou um modelo de classes baseado no dataset do IMDb, onde representa a informação que vai ser trabalhada e da sua relação entre classes, como se pode verificar na figura demonstrada abaixo.

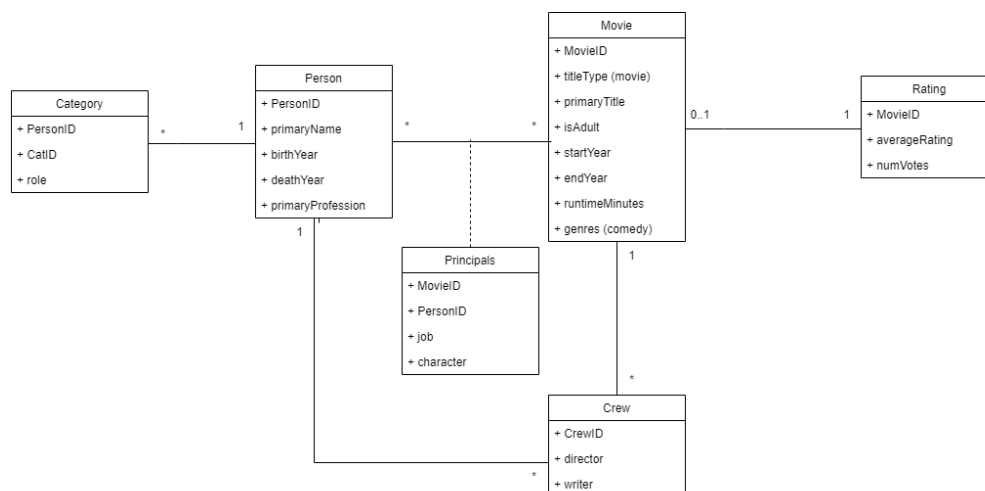


Figura 1 – Modelo de Classes

Nesta seção, serão priorizadas as relações entre as classes e o conteúdo geral constituinte do modelo, sendo que na próxima seção, haverá uma descrição minuciosa do conteúdo de cada tabela do esquema relacional correspondente às classes. Desta forma, iniciando a leitura do modelo da direita para a esquerda, temos a classe do “Rating” que consiste na atribuição de uma avaliação a cada um dos filmes presentes no dataset, filmes estes que se encontram presentes na classe “Movie”. Visto que nem todos os filmes possuíam uma avaliação foi dada então uma cardinalidade de 0 ou 1 avaliações na relação entre as classes “Rating” e “Movie”. Em relação agora a classe do “Movie”, dado o volume excessivo de dados foram impostas algumas condições em certos atributos de modo a filtrar os resultados e reduzir esse volume elevado. Uma das restrições foi no atributo “titleType”, ou seja, no tipo do conteúdo audiovisual presente no dataset, ao qual o grupo definiu que podia ser apenas do tipo “movie”, eliminando todos os outros tipos. Além desta restrição também foi imposta outra ao atributo “startYear”, ou ano de lançamento, que só seriam contabilizados filmes que estivessem compreendidos no período de tempo entre o ano de 2005 até ao ano 2023, por fim, foi imposta também uma restrição ao atributo “genres” que só podia corresponder ao género de comédia. Ainda conectado a classe “Movie”, temos a classe “Crew” que define o diretor que realiza o filme, ao qual foi atribuída uma cardinalidade de muitos para 1, uma vez que um filme pode ser realizado por vários diretores, diretores esses que fazem parte do conjunto de pessoas presentes na classe “Person”.

A colaborar com vários “Movies” estão várias pessoas, classe “Person”, sendo que entres esta existe uma classe associativa chamada de “Principals” onde contém os ID’s da classe “Person” e da classe “Movie” e ainda como atributos a função da pessoa no filme e respetivas personagens.

Posto isto, encontra-se a classe “Category”, onde cada pessoa está ligada a várias categorias de acordo com o tipo de trabalho realizado.

De forma sucinta, a classe “Rating” fornece-nos uma média das avaliações e o número de votos por parte dos utilizadores; a classe “Movie” tem a informação geral sobre os filmes, tal como o título, se é para adulto, ano de lançamento, duração em minutos; a classe “Crew” demonstra quais os diretores presentes em cada um dos filmes, cada filme pode ter vários diretores; a classe “Principals” contém informação sobre o a função da pessoa no filme e o nome das personagens que esta desempenha; a classe “Person” que contém o nome, ano de nascimento, ano de falecimento e profissões desempenhadas até então; por fim, temos a classe “Category” que informação sobre o tipo de trabalho realizado pelas pessoas.

5 Esquema Relacional

Numa primeira fase, procedeu-se à conceção do esquema relacional que iria servir de base à criação da Base de Dados, tendo sido elaborado com base no modelo de classes UML previamente apresentado. No que diz respeito ao modelo relacional, é crucial que se proceda ao desenho integral do esquema da base de dados, o que engloba a definição dos esquemas das tabelas e das restrições de integridade. As principais restrições utilizadas no âmbito deste trabalho foram a chave primária, que identifica de forma única uma entidade, e a integridade referencial, que impõe a existência do valor referido na Base de Dados, sendo, assim, uma chave externa.

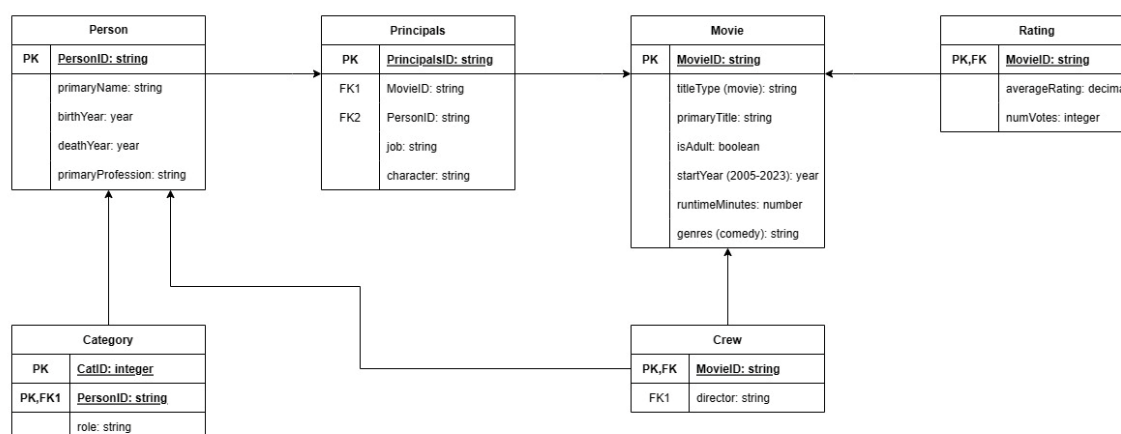


Figura 2 – Esquema Relacional

As tabelas usadas contêm a seguinte informação:

- **Movie** - Contains the following information for titles:
 - MovieID (string)- identificador de cada filme
 - titleType (string) – (movie)- tipo de título/filme, restringido para apenas filmes
 - primaryTitle (string) – título principal do filme
 - isAdult (boolean)- 0: sem conteúdo adulto; 1: com conteúdo adulto
 - StartYear (large number)- (2005-2023) – data de lançamento, restringida entre 2005 e 2023
 - RuntimeMinutes (number) – duração do conteúdo, em minutos
 - genres (string) – (comedy)- género do conteúdo, restringido para comédia
- **Crew** – Contém o identificador do diretor para cada filme:
 - MovieID (string)- identificador de cada filme
 - FK – MovieID (Movie)

- director (string)- identificador do diretor do filme
 - FK1 – PersonID (Person)
- Principals – Contém os principais elementos colaboradores de cada filme:
 - PrincipalsID (number)- identificador de cada elemento
 - MovieID (string)- combinação de caracteres, que identifica cada filme
 - FK1 – MovieID (Movie)
 - PersonID (string)- identificador da pessoa
 - FK2 – PersonID (Person)
 - job (string)- função da pessoa no filme, se aplicável
 - character (string)- nome da personagem que a pessoa representa, se aplicável
- Rating – Contém informação sobre a avaliação dos filmes:
 - MovieID (string)- identificador de cada filme
 - FK – MovieID (Movie)
 - averageRating – média dos valores de avaliação atribuídos pelos utilizadores
 - numVotes- número de votos
- Person – Contém informação sobre as pessoas:
 - PersonID (string)- identificador da pessoa
 - primaryName (string)– nome principal pelo qual a pessoa é conhecida
 - BirthYear (string) – ano de nascimento da pessoa
 - DeathYear (string) – ano da morte da pessoa, se aplicável
 - primaryProfession (string) – 3 principais profissões da pessoa
- Category – Contém informação sobre o trabalho das pessoas:
 - CatID (number) – identificador da categoria
 - PersonID (string)- identificador da pessoa
 - FK – PersonID (Person)
 - role (string)- tipo de trabalho realizado pela pessoa

Após a importação para o Microsoft Access, foram criadas relações entre as tabelas tendo em conta as relações de cardinalidade que foram definidas no modelo de classes (UML), bem como as relações de integridade referencial com chaves externas (FK).

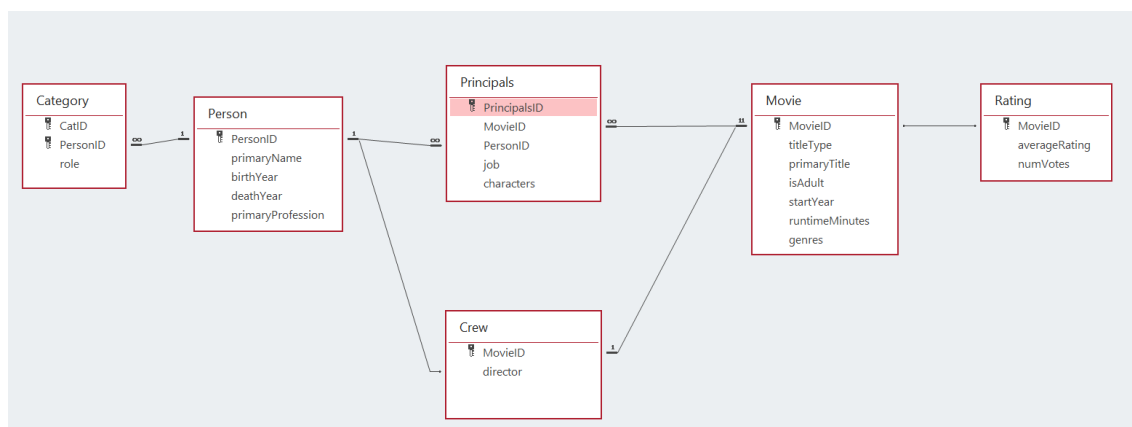


Figura 3 – Esquema Relacional (Acess)

O esquema relacional manteve-se, com se pode observar na FIGURA, apenas com algumas alterações no tipo de dados, devido a incoerências nos dados disponibilizados pelo IMDb, de modo a não violar os mesmos.

As alterações foram as seguintes:

- Movie
 - isAdult- mudança do tipo de dados “boolean” para “large number”, porque ao mudar para “yes/no”, dados eram eliminados por incompatibilidade
 - startYear- mudança do tipo de dados “year” para “large number”, porque ao mudar para “date/time”, dados eram eliminados por incompatibilidade
- Principals
 - PrincipalsID- mudança do tipo de dados “string” para “number”, porque era possível a especificação
- Rating
 - averageRating- mudança do tipo de dados “decimal” para “short text”, porque ao usar o tipo de dados “decimal”, todos os valores mudavam para 0 (zero), logo, os dados eram perdidos
 - NumVotes- mudança do tipo de dados “integer” para “number- double”, porque ao usar o tipo de dados “number- integer”, os dados eram eliminados por incompatibilidade
- Person

- birthYear, deathYear- mudança do tipo de dados “year” para “short text”, porque ao mudar para “date/time”, dados eram eliminados por incompatibilidade
- Category
 - CatID- mudança do tipo de dados “integer” para “number- double”, porque ao usar o tipo de dados “number- integer”, os dados eram eliminados por incompatibilidade

É importante referir também a mudança do tipo de dados “string” para “short text”, isto porque, no Microsoft Access, “short text” é utilizado para o tipo de dados “string”

Quanto às chaves externas a relação foi criada através da funcionalidade “Lookup Wizard”, selecionando o campo que era necessário. Foram realizadas as seguintes chaves externas das tabelas:

- ‘Principals’
 - MovieID com MovieID (Tabela Movie);
 - PersonID com PersonID (Tabela Person);
- ‘Crew’
 - Director com PersonID (Tabela Person).

6 Instâncias

Como os dados e as tabelas foram trabalhadas antes da migração para a Base de dados, quando se realizou a migração não foram necessárias mais simplificações, todas as modificações estão descritas na secção 3 ([Metodologia e Ferramentas](#)). Após a importação das tabelas foi realizada a cadeia de relações, onde as associações automáticas estavam corretas pelo programa, apenas foi necessário atribuir chaves primárias e chaves secundárias às tabelas.

Posto isto, nas figuras abaixo são apresentados excertos de instâncias utilizadas para a criação da Base de Dados. Primeiramente apresenta-se na [Figura 3](#) a tabela “Category”, que possui 10472 dados, que apresenta o ID da categoria, o ID das pessoas e o papel correspondente desta.

Category				
CatID	PersonID	role	Click to Add	
1	nm0929093	actor		
2	nm0813219	actor		
3	nm1245343	actor		
4	nm0620847	actor		
5	nm0046033	actor		
6	nm0073162	actor		
7	nm0038932	actor		
8	nm0210690	actor		
9	nm0285109	actor		
10	nm0079343	actor		
11	nm1525688	actor		
12	nm0043609	actor		
13	nm0614971	actor		
14	nm0096542	actor		
15	nm0217015	actor		
16	nm0176444	actor		
17	nm0846159	actor		
18	nm0315195	actor		
19	nm0103977	actor		
20	nm1237063	actor		
21	nm0764769	actor		

Record: 1 of 10472 No Filter Search

Figura 4 – Instância da tabela ‘Category’

Relativamente à tabela “Crew” apresentada na [Figura 5](#) esta tabela contém 1203 dados e apresenta o ID dos filmes e os diretores associados aos respetivos filmes.

Crew			
	MovieID	director	Click to Add
+	tt0111414	nm0398271	
+	tt0119231	nm0818538	
+	tt0159790	nm0311780	
+	tt0179803	nm0485548	
+	tt0199602	nm0953395	
+	tt0227671	nm0372932	
+	tt0238992	nm0189236	
+	tt0243246	nm0865379	
+	tt0244521	nm0340037	
+	tt0258901	nm0101218	
+	tt0285252	nm0899854	
+	tt0288190	nm1028004	
+	tt0294654	nm0116145	
+	tt0299215	nm1069270	
+	tt0300007	nm0023852	
+	tt0302498	nm0589056	
+	tt0304876	nm0667041	
+	tt0310334	nm0871456	
+	tt0322999	nm0415148	
+	tt0337584	nm0886684	

Record: 1 of 1203 No Filter Search

Figura 5 – Instância da tabela ‘Crew’

Na tabela principal Movie que é apresentada na [Figura 6](#) que contém 1203 dados onde apresenta o ID do filme, o respetivo título do filme, se este filme é só para adultos ou não com números binários de 0 ou 1 (em que 0 é para todos e 1 é só para adultos), o ano de lançamento do filme, o tempo do filme e o género do filme (sendo que neste foi adicionada uma restrição para só obtermos filmes de comédia).

MovieID	titleType	primaryTitle	isAdult	startYear	runtimeMini	genres	Click to Add
tt0111414	movie	A Thin Life	0	2018	75	Comedy	
tt0119231	movie	The Junior Defenders	0	2007	83	Comedy	
tt0159790	movie	The Third Vehicle	0	2005		Comedy	
tt0179803	movie	Angels with Angles	0	2005	87	Comedy	
tt0199602	movie	Holy Hollywood	0	2021	82	Comedy	
tt0227671	movie	Tortilla Heaven	0	2007	98	Comedy	
tt0238992	movie	Fortunes	0	2005	91	Comedy	
tt0243246	movie	Vultures	0	2005		Comedy	
tt0244521	movie	Funny Money	0	2006	98	Comedy	
tt0258901	movie	The Luau	0	2005	90	Comedy	
tt0285252	movie	Bachelor Trip	0	2012	100	Comedy	
tt0288190	movie	El reino de los cielos	0	2009		Comedy	
tt0294654	movie	Hollywood Dot Com	0	2007	87	Comedy	
tt0299215	movie	Wheelmen	0	2005	93	Comedy	
tt0300007	movie	The Man Who Would Be Queen	0	2007		Comedy	
tt0302498	movie	Raising Flagg	0	2006	103	Comedy	
tt0304876	movie	Live Wire	0	2009	81	Comedy	
tt0310334	movie	Waitin' to Live	0	2006	95	Comedy	
tt0322999	movie	Ze film	0	2005	105	Comedy	
tt0337584	movie	Backseat	0	2005	80	Comedy	

Figura 6 – Instância da tabela 'Movie'

A tabela Person, como demonstra a [Figura 7](#) contém 10472 dados onde apresenta o ID das pessoas, o Nome, o ano de nascimento e ano de morte (caso este tenha morrido, caso não tenha este está representado com uma célula vazia) e por fim a sua profissão.

PersonID	primaryName	birthYear	deathYear	primaryProfession	Click to Add
nm0000080	Orson Welles	1915	1985	actor,director,writer	
nm0000092	John Cleese	1939		actor,writer,producer	
nm0000111	Matthew Broderick	1962		actor,soundtrack,director	
nm0000114	Steve Buscemi	1957		actor,producer,director	
nm0000117	Neve Campbell	1973		actress,producer,soundtrack	
nm0000155	Whoopi Goldberg	1955		actress,producer,writer	
nm0000156	Jeff Goldblum	1952		actor,soundtrack,producer	
nm0000157	Linda Hamilton	1956		actress,soundtrack	
nm0000167	Elizabeth Hurley	1965		actress,producer	
nm0000168	Samuel L. Jackson	1948		actor,producer,soundtrack	
nm0000190	Matthew McConaughey	1969		actor,producer,soundtrack	
nm0000199	Al Pacino	1940		actor,producer,director	
nm0000205	Parker Posey	1968		actress,soundtrack,writer	
nm0000224	Alicia Silverstone	1976		actress,producer,soundtrack	
nm0000236	Jennifer Tilly	1958		actress,soundtrack,writer	
nm0000251	Timothy Dowling	1973		actor,writer,script_department	
nm0000257	Alan Alda	1936		actor,miscellaneous,soundtrack	
nm0000266	Ursula Andress	1936		actress,soundtrack	
nm0000273	Alan Arkin	1934		actor,soundtrack,director	
nm0000275	Rosanna Arquette	1959		actress,producer,director	

Record: 1 of 10472 No Filter Search

Figura 7 – Instância da tabela 'Person'

Quanto à tabela dos “Principals”, possui 11196 dados como se pode observar na [Figura 8](#) em que se pode observar o ID dos “principals”, o ID do filme, o ID da pessoa, o trabalho e personagem que esta representa, sendo que se não tiver informação sobre o mesmo o campo fica vazio.

Principals						
PrincipalsID	MovieID	PersonID	job	characters	Click to Add	
1	tt0111414	nm0398271				
2	tt0119231	nm0315993				
3	tt0119231	nm0641168				
4	tt0119231	nm0818538				
5	tt0119231	nm0765769				
6	tt0159790	nm0311780				
7	tt0179803	nm2021987				
8	tt0179803	nm2020797				
9	tt0179803	nm0277811				
10	tt0199602	nm0615773				
11	tt0199602	nm0953395				
12	tt0199602	nm1163215				
13	tt0199602	nm0113423				
14	tt0227671	nm0372932				
15	tt0238992	nm0949711				
16	tt0238992	nm0189236				
17	tt0238992	nm0759538				
18	tt0238992	nm0819855				
19	tt0238992	nm0258303				
20	tt0238992	nm0504303				

Record: 1 of 11196 No Filter Search

Figura 8 – Instância da tabela ‘Principals’

Por fim quanto à tabela do Rating, contém 1097 dados como se observa na [Figura 9](#) devido a alguns filmes não possuírem classificação, sendo que nesta tabela demonstra o ID do filme, a média da classificação e o número de votos.

Rating				
	MovieID	averageRatir	numVotes	Click to Add
+	tt0119231	4,6	187	
+	tt0179803	5,0	152	
+	tt0199602	6,6	35	
+	tt0227671	5,3	450	
+	tt0238992	4,3	166	
+	tt0244521	5,2	3155	
+	tt0258901	5,4	111	
+	tt0285252	3,6	275	
+	tt0288190	5,4	10	
+	tt0294654	6,1	23	
+	tt0299215	4,2	118	
+	tt0300007	5,8	35	
+	tt0302498	5,3	370	
+	tt0304876	5,4	82	
+	tt0310334	5,9	94	
+	tt0322999	5,4	277	
+	tt0337584	6,9	82	
+	tt0343603	3,1	130	
+	tt0348333	6,7	100972	
+	tt0353765	5,7	61	

Record: 1 of 1097 No Filter Search

Figura 9 – Instância da tabela 'Rating'

7 Interrogações

Após a importação de dados das tabelas, procedeu-se à construção de interrogações à base de dados em SQL, colocando em prática os conhecimentos adquiridos na unidade curricular.

Foram realizadas 28 queries, em que o grupo decidiu dividir por três tipos de interrogações: simples, de média dificuldade e complexas. Em anexo, é possível visualizar as impressões dos resultados das queries ([Interrogações](#)).

7.1 Interrogações simples

Foram realizadas 14 queries de interrogações simples que foram as seguintes:

- **Query 1 (Nome_pessoas_envolvidas_filmes): Qual o nome de todas as pessoas?**

```
SELECT p.primaryName AS Name, p.primaryProfession AS Profession  
  
FROM Person AS p;
```

- **Query 2 (Idade_Falecidos): Quais as idades das pessoas falecidas?**

```
SELECT primaryName AS Name, (deathyear-birthyear) AS Age  
  
FROM Person  
  
WHERE (deathyear-birthyear) is not null;
```

- **Query 3 (Person_profession): Quais são as profissões correspondentes às pessoas?**

```
SELECT p.primaryName AS Name, p.primaryProfession AS Profession  
  
FROM Person AS p;
```

- **Query 4 (Filme_Mais_Minutos): Qual o filme com maior duração?**

```
SELECT TOP 1 m.primaryTitle AS Movie, m.runtimeMinutes AS Runtime  
  
FROM Movie AS m  
  
WHERE m.runtimeMinutes Is Not Null  
  
ORDER BY m.runtimeMinutes DESC;
```

– Query 5 (Filme_Menos_Minutos): Qual o filme com menor duração?

```
SELECT TOP 1 m.primaryTitle AS Movie, m.runtimeMinutes AS Runtime
FROM Movie AS m
WHERE m.runtimeMinutes is not null
ORDER BY m.runtimeMinutes;
```

– Query 6 (Rating_Filme): Qual o Rating de cada filme?

```
SELECT m.primaryTitle AS Movie, r.averageRating AS Rating
FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID
ORDER BY r.averageRating DESC;
```

– Query 7 (TOP245_Filmes): Qual o top 245 de filmes por ordem descendente de rating?

```
SELECT TOP 245 m.primaryTitle AS Movie, m.startYear AS [Year], r.averageRating AS Rating,
r.numVotes AS Votes
FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID
ORDER BY r.averageRating DESC;
```

– Query 8 (Media_rating_filmes): Qual a Média do Rating?

```
SELECT Avg(averageRating) AS Total_AVG
FROM Rating;
```

– Query 9 (MilaKunis_Ano_Nascimento+Idade): Qual a idade e ano de nascimento da Mila Kunis?

```
SELECT primaryName AS Name, birthYear AS Birth_Year, (2023-birthYear) AS Age
FROM Person
WHERE primaryName Like "**Mila Kunis**";
```

– Query 10 (Filmes_MilaKunis): Em que filmes participa a Mila Kunis?

```
SELECT p.primaryName AS Name, m.primaryTitle AS Title

FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN
Movie AS m ON pr.MovieID=m.MovieID

WHERE p.primaryName="Mila Kunis"

ORDER BY m.primaryTitle;
```

– Query 11 (Filme_ScaryMovie_Pessoas): Quais foram as pessoas envolvidas no filme Scary Movie e respetivas personagens?

```
SELECT p.primaryName AS Name, pr.characters AS [Character]

FROM (Movie AS m INNER JOIN Principals AS pr ON m.MovieID=pr.MovieID) INNER JOIN
Person AS p ON pr.PersonID=p.PersonID

WHERE m.primaryTitle Like "*Scary Movie*";
```

– Query 12 (Ano_mais_filmes): Qual o ano com mais filmes?

```
SELECT TOP 1 startYear AS [Year], COUNT(startYear) AS NrMovies

FROM Movie

WHERE titleType LIKE 'movie'

GROUP BY startYear

ORDER BY COUNT(startYear) DESC;
```

- **Query 13 (Filmes_José_Valdebenito_diretor): Qual/Quais foi/foram o(s) filme(s) que o José Valdebenito realizou?**

```
SELECT m.primaryTitle AS Title

FROM (Movie AS m INNER JOIN Crew AS c ON m.MovieID=c.MovieID) INNER JOIN Person
AS p ON c.director=p.PersonID

WHERE p.primaryName='José Valdebenito' AND c.MovieID=m.MovieID;
```

- **Query 14 (N_personagens_cada_filme): Quantas personagens tem cada filme?**

```
SELECT m.primaryTitle AS Movie, count(p.characters) AS Characters

FROM Movie AS m INNER JOIN Principals AS p ON m.MovieID=p.MovieID

GROUP BY m.primaryTitle;
```

7.2 Interrogações de média dificuldade

Foram realizadas mais 11 queries de interrogações de média dificuldade que foram as seguintes:

- **Query 15 (Filme_maior_n_votos): Qual o filme com maior número de votos?**

```
SELECT Movie.primaryTitle AS Title, Rating.numVotes AS NrVotes

FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID

WHERE numVotes =

(SELECT MAX(numVotes) FROM Rating);
```

- **Query 16 (Filme_maior_rating): Qual o filme com maior rating?**

```
SELECT Movie.primaryTitle AS Title, Rating.averageRating AS Rating

FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID

WHERE averageRating =

(SELECT MAX(averageRating) FROM Rating);
```


– Query 17 (Filme_menor_rating): Qual o filme com menor rating?

```
SELECT Movie.primaryTitle AS Title, Rating.averageRating AS Rating  
  
FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID  
  
WHERE averageRating =  
  
(SELECT MIN(averageRating) FROM Rating);
```

– Query 18 (Idade_Pessoas_Vivas): Quais as idades das pessoas vivas?

```
SELECT Ages.primaryName AS Name, Ages.Age AS Age  
  
FROM (SELECT primaryName, birthyear, (2023-birthyear) AS Age FROM Person WHERE  
deathyear is null AND birthyear is not null) AS Ages  
  
ORDER BY Ages.Age;
```

– Query 19 (Ator_+velho): Qual o ator com mais idade vivo?

```
SELECT TOP 1 Ages.primaryName AS Name, Ages.Age AS Age  
  
FROM (SELECT PersonID, primaryName, birthyear, (2023-birthyear) AS Age FROM Person  
WHERE deathyear is null AND birthyear is not null) AS Ages  
  
ORDER BY Ages.Age DESC;
```

– Query 20 (Produtor_c+_filmes): Qual/Quais o(s) produtor(es) com mais filmes?

```
SELECT TOP 1 p.primaryName AS Producer, count(m.MovieID) AS Movies  
  
FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN  
Movie AS m ON pr.MovieID=m.MovieID  
  
WHERE p.primaryProfession Like "producer"  
  
GROUP BY p.primaryName  
  
ORDER BY count(m.MovieID) DESC;
```

– Query 21 (Ator_c+_filmes): Qual o ator com mais filmes?

```
SELECT TOP 1 p.primaryName AS Actor, count(m.MovieID) AS Movie

FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN
Movie AS m ON pr.MovieID=m.MovieID

WHERE p.primaryProfession="actor"

GROUP BY p.primaryName

ORDER BY count(m.MovieID) DESC;
```

– Query 22 (Filme_Rating>7,2_filmes_depois_2006_+55000votos): Qual/Quais é/são o(s) nome(s) do filme(s) que tem rating superior a 7,2, realizados depois de 2006 e que contenham um número de votos por parte dos utilizadores superior a 55000? Apresentar os resultados por ordem decendente.

```
SELECT m.primaryTitle AS Title, r.averageRating AS Rating

FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID

WHERE (r.averageRating > "7,2") AND (m.startYear>2006) AND (r.numVotes>55000)

ORDER BY r.averageRating DESC;
```

– Query 23 (NrNascimento_por_ano): Qual o número de nascimentos por ano?

```
SELECT birthYear AS Birth_Year, count(PersonID) AS NrBirths

FROM Person

WHERE birthYear is not null

GROUP BY birthYear

ORDER BY count(PersonID) DESC , birthYear DESC;
```

- Query 24 (Papel_Elenco): Por ID de título, quais são as funções de cada pessoa e que personagens são interpretadas?

```
SELECT DISTINCT Movie.MovieID AS ID, Movie.primaryTitle AS Title, Person.primaryName  
AS Name, Category.role AS Role, Principals.characters AS [Character]
```

```
FROM Movie INNER JOIN (Principals INNER JOIN (Category INNER JOIN Person ON  
Category.PersonID = Person.PersonID) ON Principals.PersonID = Person.PersonID) ON  
Principals.MovieID = Movie.MovieID;
```

- Query 25 (Nome_filme+persona_em_nome_personagem_Beauty): Nome do filme, nome da pessoa em que o nome da personagem é igual a "Beauty"?

```
SELECT m.primaryTitle AS Title, p.primaryName AS Name, l.characters AS Characters
```

```
FROM (Movie AS m INNER JOIN Principals AS l ON m.MovieID=l.MovieID) INNER JOIN  
Person AS p ON l.PersonID=p.PersonID
```

```
WHERE l.characters=["Beauty"];
```

7.3 Interrogações complexas

Por fim, foram realizadas umas 3 queries adicionais de interrogações complexas que foram as seguintes:

- **Query 26 (People_Idades): Qual o nome, data de nascimento, data de falecimento (caso tenha ocorrido), idade das pessoas (no caso de terem morrido, a idade que tinham) e as profissões das pessoas da Base de Dados?**

```
(SELECT primaryName AS Name, birthyear AS Birth_Year, deathyear AS Death_Year,
(deathyear-birthyear) AS Age, primaryProfession AS Profession

FROM Person

WHERE (deathyear-birthyear) is not null

UNION  (SELECT  Ages.primaryName,  Ages.birthyear,  Ages.deathyear,  Ages.Age,
Ages.primaryProfession

FROM (SELECT primaryName, birthyear, deathyear, (deathyear-birthyear) AS Idade_Morte,
(2023-birthyear) AS Age, primaryProfession FROM Person

WHERE (deathyear-birthyear) is null) AS Ages)

ORDER BY Age DESC;
```

- **Query 27 (5_anos_melhor_média_pontuação_users_entre_2006_2022): Quais são os 5 anos com melhor média em termos de pontuação por parte dos utilizadores entre os anos 2006 e 2022?**

```
SELECT TOP 5 Movie.startYear AS [Year], Avg(Rating.averageRating) AS Average

FROM Rating INNER JOIN Movie ON Rating.MovieID = Movie.MovieID

WHERE (((Movie.startYear)>2006 And (Movie.startYear)<2022))

GROUP BY Movie.startYear

ORDER BY Avg(Rating.averageRating) DESC;
```

- Query 28 (Idade_ator+64): Qual a idade do(s) ator(es) cuja idade seja superior a 64 anos, que participe no filme que contenha o número 2 no seu titulo, o respetivo rating, o número de votos, que o trabalho seja nulo e que o seu CatID esteja compreendido entre 799 e 7000?

```
SELECT c.CatID AS ID, p.primaryName AS Name, (deathyear-birthyear) AS Age, c.role AS
Role, l.job AS Job, m.primaryTitle AS Title, r.averageRating AS Rating, sum(r.numVotes) AS Votes

FROM (((Category AS c INNER JOIN Person AS p ON c.PersonID=p.PersonID) INNER JOIN
Principals AS l ON p.PersonID=l.PersonID) INNER JOIN Movie AS m ON l.MovieID=m.MovieID)
INNER JOIN Rating AS r ON m.MovieID=r.MovieID

WHERE c.CatID between 799 AND 7000

AND (deathyear-birthyear) > 64

AND c.role='actor'

AND l.job is null

AND m.primaryTitle Like "2*"

GROUP BY c.CatID, p.primaryName, (deathyear-birthyear), c.role, l.job, m.primaryTitle,
r.averageRating;
```

8 SGBD

As Interfaces criadas estão divididas em dois grupos: os formulários e os relatórios.

8.1 Formulários (Forms)

Quanto aos formulários criados dividem-se em dois grupos:

– Formulário de Navegação

O formulário de navegação é notável pela sua funcionalidade e simplicidade, permitindo a criação de ambientes de navegação utilizando recursos das bases de dados, os quais são organizados de maneira coerente e intuitiva. Em virtude disso, foi desenvolvido o "Menu de Navegação", conforme apresentado na FIGURA X, que possibilita o acesso às diversas partes da base de dados, facilitando a pesquisa pelo utilizador. O menu inclui 5 botões de pesquisa que dão acesso a formulários e relatórios que serão descritos posteriormente:

- FAQ (Frequently Asked Questions);
- Pesquisa de elenco por filme;
- Idade das pessoas;
- Rating dos Filmes;
- Informações sobre pessoas;
- Top 245 Filmes.



Figura 10 – Forms 'Menu'

– Os formulários de apresentação de dados ao utilizador

Os formulários criados de apresentação de dados ao utilizador foram:

- Papel Elenco;
- Idades das pessoas;
- Elenco do filme;
- Papel Elenco;
- FAQ.

Os formulários criados possuem um botão para regressar ao menu de navegação, com exceção do ‘Papel_Elenco’.

O forms ‘Papel Elenco’ foi criado apenas para fazer o forms Elenco filme para desta forma ser possível fazer a pesquisa de elenco por filme. O papel elenco apresenta o nome da pessoa que interpreta o filme, o titulo do filme, a personagem e o seu papel no filme.

The image shows a web form titled 'IMDb Elenco' with a purple header. The form is divided into two columns. The left column has a 'Name' field with the value 'Tommy Dysart' and a 'Title' field with the value 'A Thin Life'. The right column has a 'Role' field with the value 'actor' and a 'Character' field with the value '["The Man"]'. At the bottom of the form, there are four navigation buttons: 'Inicio', 'Anterior', 'Seguinte' (which is highlighted with a dashed border), and 'Fim'.

Name	Role
Tommy Dysart	actor

Title	Character
A Thin Life	["The Man"]

Inicio Anterior **Seguinte** Fim

Figura 11 – Forms ‘Elenco’

O forms 'Elenco Filme' foi criado através de um subforms com o 'Papel Elenco' para que desta forma seja possível o utilizador poder fazer a pesquisa do filme e no resultado conseguir observar o elenco. Com isto o utilizador poderá pesquisar o título do filme que tenha curiosidade em saber qual o elenco do filme em questão.

Name	Role
Tommy Dysart	actor


Title	Character
A Thin Life	["The Man"]

Inicio Anterior Seguinte Fim

Record: 1 of 2 No Filter Search


Figura 12 – Forms 'Pesquisa de Elenco por Filme'

O form 'Idades das pessoas' foi criado para saber a informação sobre as pessoas, este dispõe de informação sobre o nome, data de nascimento, data de morte (caso exista), idade e a sua profissão. Este forms foi criado com o intuito do utilizador poder pesquisar sobre um ator/realizador/etc que tenha curiosidade em saber alguma informação.



Informações sobre Pessoas

Voltar



Pesquisar

Name

Birth_Year

Death_Year

Age


Profession

Fechar

Name	Birth_Year	Death_Year	Age	Profession
María Cristina Camilo	1918		105	actress
C.Y. Lee	1915	2018	103	writer
Renée Le Calm	1918	2019	101	actress
Danielle Darrieux	1917	2017	100	actress,soundtrack
A.K. Hangal	1914	2012	98	actor

Figura 13 – Forms 'Informações sobre pessoas'

Quanto ao form FAQ foi desenvolvido e implementado com o objetivo de disponibilizar aos utilizadores as perguntas mais frequentes da base de dados de forma rápida e fácil. Este formulário é composto por doze opções: ‘Rating dos filmes’, ‘Filme com menor rating’, ‘Filme com maior rating’, ‘Pessoas envolvidas no filme Scary Movie 4’, ‘Média de rating de todos os filmes’, ‘Idade Mila Kunis’, ‘Número de nascimentos por ano’, ‘Produtor com mais filmes’, ‘Filme com menos minutos’, ‘Filme com mais minutos’, ‘Ator com mais idade’, ‘Ator com mais filmes realizados’.



The image shows a web form titled 'FAQ' (Frequently asked questions) with the IMDb logo. The form contains 12 buttons, each representing a different query:

- Rating dos Filmes
- Filme com menor rating
- Filme com maior rating
- Pessoas envolvidas no filme Scary Movie 4
- Média de rating de todos os filmes
- Idade Mila Kunis
- Número de nascimentos por ano
- Produtor com mais filmes
- Filme com menos minutos
- Filme com mais minutos
- Ator com mais idade
- Ator com mais filmes realizados

Figura 14 – Forms ‘FAQ’

Após a apresentação dos formulários, é relevante mencionar que a maioria dos campos, conforme explicado anteriormente, encontram-se protegidos (*locked*), impedindo a modificação ou inserção desnecessária de dados. O único campo que não se encontra locked é a barra de pesquisa para possibilitar a pesquisa aos utilizadores.

8.2 Relatórios (Reports)

Quanto aos relatórios foram criados 4 sendo estes:

- Idade das pessoas vivas;
- Nascimentos por ano;
- Rating dos Filmes;
- Top 245 filmes.

Os relatórios criados possuem um botão para regressar ao menu de navegação.

O relatório 'Idade das pessoas vivas' possibilita ao utilizador visualizar a informação do nome e respetiva idade da pessoa.

IMDb Idade das Pessoas Envolvidas		7 de abril de 2023 17:22:17	Voltar
Name		Age	
Kayden Bryce		20	
Victoria Leigh		22	
Mason Cook		23	
Finley Bernsen		25	
Abhinay Berde		26	
Jimmy Bennett		27	
Daniel Deli		28	
Nicolas Jouxte		28	
Nam Ji-hyun		28	
Logan Paul		28	
Natalia Pujszo		29	
Elijah Runcorn		29	
Myrtle Sarrosa		29	

Figura 15 – Relatório 'Idade das Pessoas Envolvidas'

O relatório 'Nr de Nascimentos por ano' possibilita ao utilizador visualizar a informação do ano e quantos nascimentos das pessoas existem nesse ano.

IMDb Nr de Nascimentos por ano		7 de abril de 2023 17:22:37	Voltar
Birth_Year	NrBirths		
1972	154		
1971	152		
1973	145		
1977	138		
1976	133		
1964	127		
1970	126		
1968	125		
1974	121		
1975	121		
1979	117		
1969	115		
1965	113		

Figura 16 – Relatório 'Nr de Nascimentos por ano'

O relatório intitulado 'rating dos filmes' possibilita ao utilizador visualizar o rating atribuído a cada filme. O grupo optou por ordenar as avaliações de forma decrescente, facilitando a identificação rápida do filme com melhor classificação.

 Rating dos Filmes		7 de abril de 2023 17:23:01	Voltar
Movie	Rating		
A Most Particularly Peculiar Bank Heist	9,6		
Generation Wrecks	9,1		
Unsung	9,0		
Love Lessons	9,0		
Sweet Tessie and Bags	9,0		
Mass Romantic	8,9		
Stan	8,9		
Paperdolls	8,9		
Filmic Achievement	8,8		
Living with Uncle Ray	8,7		
Sunny & Share Love You	8,7		
The Bonnie Situation	8,7		
The Cleaning Lady	8,6		
Touch Wood	8,6		
The Agonist	8,6		
Fellowship	8,6		
The Alpha Geek	8,6		
Motel, Glimpse	8,6		

Figura 17 – Relatório 'Rating dos Filmes'

Por fim quanto ao relatório 'Top 245 Filmes' possibilita ao utilizador visualizar 245 resultados dos melhores filmes contendo o nome do filme, ano de lançamento e respetivo rating. Adicionalmente, foi disponibilizado o número de votos associado a cada avaliação de filme, de forma a permitir uma análise mais precisa por parte do utilizador acerca do conteúdo em questão, tendo em conta que a classificação por vezes pode ser enganadora.

IMDb Top 245 Filmes			
7 de abril de 2023 17:23:25			Voltar
Movie	Year	Rating	Votes
A Most Particularly Peculiar Bank Heist	2005	9,6	8
Generation Wrecks	2021	9,1	137
Unsung	2005	9,0	23
Love Lessons	2005	9,0	13
Sweet Tessie and Bags	2008	9,0	23
Mass Romantic	2008	8,9	20
Stan	2011	8,9	34
Paperdolls	2006	8,9	11
Filmic Achievement	2005	8,8	66
Living with Uncle Ray	2006	8,7	27
Sunny & Share Love You	2007	8,7	73
The Bonnie Situation	2005	8,7	16
The Cleaning Lady	2005	8,6	15
Touch Wood	2010	8,6	21
The Agonist	2007	8,6	10
Fellowship	2005	8,6	34
The Alpha Geek	2009	8,6	24

Figura 18 – Relatório 'Top 245 Filmes'

As funcionalidades da interface são passíveis de ver na seguinte hiperligação:

- <https://www.youtube.com/watch?v=jrtofgEpMol>

9 Conclusão

O trabalho iniciou-se com a escolha da base de dados IMDb, tendo em conta a sua dimensão, com isto os conhecimentos foram aprofundados quanto à cultura cinematográfica, mais precisamente sobre os filmes de comédia entre os anos 2005 a 2023.

Através do desenvolvimento do trabalho prático foi possível aplicar conhecimentos de conteúdos lecionados na unidade curricular e ainda de forma autónoma pesquisar mais informação que se complementa à aprendizagem do grupo.

Ao longo do trabalho foram colocadas em prática as capacidades de criação do modelo de classes, bem como o respetivo modelo relacional, após a conclusão de ambos os modelos procedeu-se à filtragem das tabelas de Excel com o uso da ferramenta PowerBI, por forma a ser possível a importação das mesmas para o Microsoft Access.

Numa fase seguinte foram elaboradas questões com os vários níveis de dificuldade em SQL à base de dados criada, por forma a recuperar informação relevante de acordo com a necessidade de informação dos nossos utilizadores. Foi ainda criado 5 formulários com o intuito de tornar o acesso à informação mais intuitivo por parte do utilizador. Finalizamos o trabalho com a criação de 4 relatórios exemplos, que continham informações que qualquer utilizador que utilize este tipo de base de dados, poderia fazer.

Em suma, este trabalho sintetiza a prática da aprendizagem da unidade curricular, realizando várias etapas importantes que futuramente serão úteis no mercado de trabalho.

10 Referências

IMDb. (2023). *DataSets*. Obtido de IMDb: <https://datasets.imdbws.com/>

IMDb. (2023). *IMDb Datasets*. Obtido de IMDb: <https://www.imdb.com/interfaces/>

Lavery, T. (2017). *Internet Movie Database (IMDb)*. Obtido de TechTarget:
<https://www.techtarget.com/whatis/definition/Internet-Movie-Database-IMDb>

Anexos

Interrogações

Aqui vão ser apresentados os prints das interrogações realizadas.

- Query 1 (Nome_pessoas_envolvidas_filmes): Qual o nome de todas as pessoas?

```
SELECT p.primaryName AS Name, p.primaryProfession AS Profession
```

```
FROM Person AS p;
```



Name	Profession
Orson Welles	
John Cleese	
Matthew Broderick	
Steve Buscemi	
Neve Campbell	
Whoopi Goldberg	
Jeff Goldblum	
Linda Hamilton	
Elizabeth Hurley	
Samuel L. Jackson	
Matthew McConaughey	
Al Pacino	
Parker Posey	
Alicia Silverstone	
Jennifer Tilly	
Timothy Dowling	
Alan Alda	
Ursula Andress	
Alan Arkin	
Rosanna Arquette	
Linda Blair	

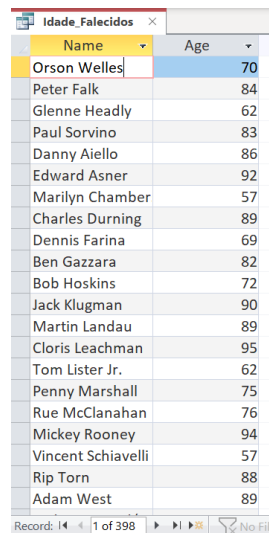
Figura 19 – Query 1 (Nome_pessoas_envolvidas_filmes)

– Query 2 (Idade_Falecidos): Quais as idades das pessoas falecidas?

SELECT primaryName AS Name, (deathyear-birthyear) AS Age

FROM Person

WHERE (deathyear-birthyear) is not null;



Name	Age
Orson Welles	70
Peter Falk	84
Glenne Headly	62
Paul Sorvino	83
Danny Aiello	86
Edward Asner	92
Marilyn Chamber	57
Charles Durning	89
Dennis Farina	69
Ben Gazzara	82
Bob Hoskins	72
Jack Klugman	90
Martin Landau	89
Cloris Leachman	95
Tom Lister Jr.	62
Penny Marshall	75
Rue McClanahan	76
Mickey Rooney	94
Vincent Schiavelli	57
Rip Torn	88
Adam West	89

Figura 20 – Query 2 (Idade_Falecidos)

– Query 3 (Person_profession): Quais são as profissões correspondentes às pessoas?

SELECT p.primaryName AS Name, p.primaryProfession AS Profession

FROM Person AS p;



Name	Profession
Orson Welles	actor,director,writer
John Cleese	actor,writer,producer
Matthew Broderick	actor,soundtrack,director
Steve Buscemi	actor,producer,director
Neve Campbell	actress,producer,soundtrack
Whoopi Goldberg	actress,producer,writer
Jeff Goldblum	actor,soundtrack,producer
Linda Hamilton	actress,soundtrack
Elizabeth Hurley	actress,producer
Samuel L. Jackson	actor,producer,soundtrack
Matthew McConaughey	actor,producer,soundtrack
Al Pacino	actor,producer,director
Parker Posey	actress,soundtrack,writer
Alicia Silverstone	actress,producer,soundtrack
Jennifer Tilly	actress,soundtrack,writer
Timothy Dowling	actor,writer,script_department
Alan Alda	actor,miscellaneous,soundtrack
Ursula Andress	actress,soundtrack
Alan Arkin	actor,soundtrack,director
Rosanna Arquette	actress,producer,director
Linda Blair	actress,soundtrack,producer

Figura 21 – Query 3 (Person_profession)

– Query 4 (Filme_Mais_Minutos): Qual o filme com maior duração?

```
SELECT TOP 1 m.primaryTitle AS Movie, m.runtimeMinutes AS Runtime
FROM Movie AS m
WHERE m.runtimeMinutes Is Not Null
ORDER BY m.runtimeMinutes DESC;
```

Filme_Mais_Minutos	
Movie	Runtime
Ti Dete To Deto Te Detat Saglech Detat Shivya	197
*	0

Figura 22 – Query 4 (Filme_Mais_Minutos)

– Query 5 (Filme_Menos_Minutos): Qual o filme com menor duração?

```
SELECT TOP 1 m.primaryTitle AS Movie, m.runtimeMinutes AS Runtime
FROM Movie AS m
WHERE m.runtimeMinutes is not null
ORDER BY m.runtimeMinutes;
```

Filme_menos_minutos	
Movie	Runtime
Homicidal Hollywood Babes	45
*	0

Figura 23 – Query 5 (Filme_Menos_Minutos)

– Query 6 (Rating_Filme): Qual o Rating de cada filme?

```
SELECT m.primaryTitle AS Movie, r.averageRating AS Rating

FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID

ORDER BY r.averageRating DESC;
```

Movie	Rating
A Most Particularly Peculiar Bank Heist	9,6
Generation Wrecks	9,1
Unsung	9,0
Love Lessons	9,0
Sweet Tessie and Bags	9,0
Mass Romantic	8,9
Stan	8,9
Paperdolls	8,9
Filmic Achievement	8,8
Living with Uncle Ray	8,7
Sunny & Share Love You	8,7
The Bonnie Situation	8,7
The Cleaning Lady	8,6
Touch Wood	8,6
The Agonist	8,6
Fellowship	8,6
The Alpha Geek	8,6
Motel, Glimpse	8,6
Repetition	8,6
Ella at Five	8,6
Jackrabbit Sky	8,6

Figura 24 – Query 6 (Rating_Filme)

- Query 7 (TOP245_Filmes): Qual o top 245 de filmes por ordem descendente de rating?

```
SELECT TOP 245 m.primaryTitle AS Movie, m.startYear AS [Year], r.averageRating AS Rating,
r.numVotes AS Votes
```

```
FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID
```

```
ORDER BY r.averageRating DESC;
```

Movie	Year	Rating	Votes
A Most Particularly Peculiar Bank Heist	2005	9,6	8
Generation Wrecks	2021	9,1	137
Unsung	2005	9,0	23
Love Lessons	2005	9,0	13
Sweet Tessie and Bags	2008	9,0	23
Mass Romantic	2008	8,9	20
Stan	2011	8,9	34
Paperdolls	2006	8,9	11
Filmic Achievement	2005	8,8	66
Living with Uncle Ray	2006	8,7	27
Sunny & Share Love You	2007	8,7	73
The Bonnie Situation	2005	8,7	16
The Cleaning Lady	2005	8,6	15
Touch Wood	2010	8,6	21
The Agonist	2007	8,6	10
Fellowship	2005	8,6	34
The Alpha Geek	2009	8,6	24
Motel, Glimpse	2005	8,6	20
Repetition	2005	8,6	51
Ella at Five	2006	8,6	45
Jackrabbit Sky	2009	8,6	14

Figura 25 – Query 7 (TOP245_Filmes)

- Query 8 (Media_rating_filmes): Qual a Média do Rating?

```
SELECT Avg(averageRating) AS Total_AVG
```

```
FROM Rating;
```

Total_AVG
5,79115770282588

Figura 26 – Query 8 (Media_rating_filmes)

- Query 9 (MilaKunis_Ano_Nascimento+Idade): Qual a idade e ano de nascimento da Mila Kunis?

```
SELECT primaryName AS Name, birthYear AS Birth_Year, (2023-birthYear) AS Age
```

```
FROM Person
```

```
WHERE primaryName Like "*Mila Kunis*";
```

MilaKunis_Ano_Nascimento+Idade		
Name	Birth_Year	Age
Mila Kunis	1983	40
*		

Figura 27 – Query 9 (MilaKunis_Ano_Nascimento+Idade)

- Query 10 (Filmes_MilaKunis): Em que filmes participa a Mila Kunis?

```
SELECT p.primaryName AS Name, m.primaryTitle AS Title
```

```
FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN  
Movie AS m ON pr.MovieID=m.MovieID
```

```
WHERE p.primaryName="Mila Kunis"
```

```
ORDER BY m.primaryTitle;
```

Filmes_MilaKunis	
Name	Title
Mila Kunis	Tom Cool
*	

Figura 28 – Query 10 (Filmes_MilaKunis)

- Query 11 (Filme_ScaryMovie_Pessoas): Quais foram as pessoas envolvidas no filme Scary Movie e respetivas personagens?

```
SELECT p.primaryName AS Name, pr.characters AS [Character]

FROM (Movie AS m INNER JOIN Principals AS pr ON m.MovieID=pr.MovieID) INNER JOIN
Person AS p ON pr.PersonID=p.PersonID

WHERE m.primaryTitle Like "*Scary Movie*";
```

Name	Character
David Zucker	
Craig Mazin	
Jim Abrahams	
Pat Proft	
Marlon Wayans	
Shawn Wayans	
Anna Faris	["Cindy"]
Regina Hall	["Brenda"]
Craig Bierko	["Tom"]
Bill Pullman	["Henry Hale"]

Figura 29 – Query 11 (Filme_ScaryMovie_Pessoas)

- Query 12 (Ano_mais_filmes): Qual o ano com mais filmes?

```
SELECT TOP 1 startYear AS [Year], COUNT(startYear) AS NrMovies

FROM Movie

WHERE titleType LIKE 'movie'

GROUP BY startYear

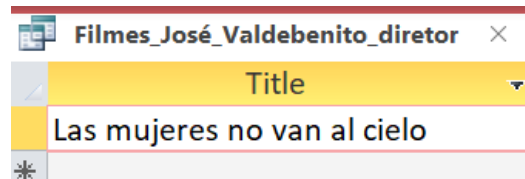
ORDER BY COUNT(startYear) DESC;
```

Year	NrMovies
2005	384

Figura 30 – Query 12 (Ano_mais_filmes)

- Query 13 (Filmes_José_Valdebenito_diretor): Qual/Quais foi/foram o(s) filme(s) que o José Valdebenito realizou?

```
SELECT m.primaryTitle AS Title
FROM (Movie AS m INNER JOIN Crew AS c ON m.MovieID=c.MovieID) INNER JOIN Person
AS p ON c.director=p.PersonID
WHERE p.primaryName='José Valdebenito' AND c.MovieID=m.MovieID;
```



Title
Las mujeres no van al cielo

Figura 31 – Query 13 (Filmes_José_Valdebenito_diretor)

- Query 14 (N_personagens_cada_filme): Quantas personagens tem cada filme?

```
SELECT m.primaryTitle AS Movie, count(p.characters) AS Characters
FROM Movie AS m INNER JOIN Principals AS p ON m.MovieID=p.MovieID
GROUP BY m.primaryTitle;
```

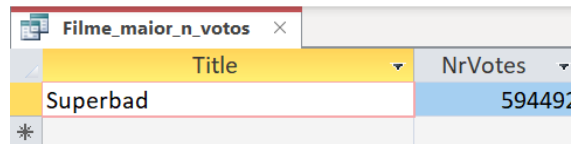


Movie	Characters
...e se domani	4
¡Soy un pelele!	4
10,000 Apologies	3
100 Kada 10	0
13th Grade	7
1409. Afera na zamku Bartenstein	4
15 Lakh Kado Aauga	0
18.15 Uhr ab Ostkreuz	4
1800 gramów	4
2 Fingers Honey	7
20 Something	4
2002 sudak odyssey	8
29 Reasons to Run	4
29th & Gay	4
3 Friends	4
4-4-2 - Il gioco più bello del mondo	4
6 Month Rule	4
7 More Days in the Tank	1
99	4
A Bug and a Bag of Weed	4
A Couple of Days and Nights	4

Figura 32 – Query 14 (N_personagens_cada_filme)

– Query 15 (Filme_maior_n_votos): Qual o filme com maior número de votos?

```
SELECT Movie.primaryTitle AS Title, Rating.numVotes AS NrVotes
FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID
WHERE numVotes =
(SELECT MAX(numVotes) FROM Rating);
```

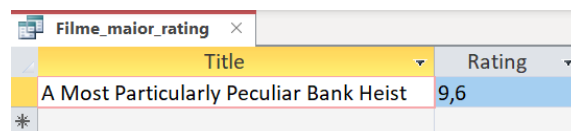


Title	NrVotes
Superbad	594492

Figura 33 – Query 15 (Filme_maior_n_votos)

– Query 16 (Filme_maior_rating): Qual o filme com maior rating?

```
SELECT Movie.primaryTitle AS Title, Rating.averageRating AS Rating
FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID
WHERE averageRating =
(SELECT MAX(averageRating) FROM Rating);
```

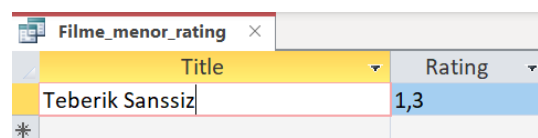


Title	Rating
A Most Particularly Peculiar Bank Heist	9,6

Figura 34 – Query 16 (Filme_maior_rating)

– Query 17 (Filme_menor_rating): Qual o filme com menor rating?

```
SELECT Movie.primaryTitle AS Title, Rating.averageRating AS Rating
FROM Rating INNER JOIN Movie ON Rating.MovieID=Movie.MovieID
WHERE averageRating =
(SELECT MIN(averageRating) FROM Rating);
```



Title	Rating
Teberik Sanssiz	1,3

Figura 35 - Query 17 (Filme_menor_rating)

– Query 18 (Idade_Pessoas_Vivas): Quais as idades das pessoas vivas?

```
SELECT Ages.primaryName AS Name, Ages.Age AS Age

FROM (SELECT primaryName, birthyear, (2023-birthyear) AS Age FROM Person WHERE
deathyear is null AND birthyear is not null) AS Ages

ORDER BY Ages.Age;
```

Idade_Pessoas_Vivas	
Name	Age
Kayden Bryce	20
Victoria Leigh	22
Mason Cook	23
Finley Bernsen	25
Abhinav Berde	26
Jimmy Bennett	27
Daniel Deli	28
Nicolas Jouxte	28
Nam Ji-hyun	28
Logan Paul	28
Natalia Pujso	29
Elijah Runcorn	29
Myrtle Sarrosa	29
Ahmet Dursun	30
Fanna Simó	30
EJ Jallorina	30
Vincent Lacoste	30
Alexandre Nachi	30
Daryl Sabara	31
Henry Bernsen	31
Scott Beaudin	31

Figura 36 – Query 18 (Idade_Pessoas_Vivas)

– Query 19 (Ator_+velho): Qual o ator com mais idade vivo?

```
SELECT TOP 1 Ages.primaryName AS Name, Ages.Age AS Age

FROM (SELECT PersonID, primaryName, birthyear, (2023-birthyear) AS Age FROM Person
WHERE deathyear is null AND birthyear is not null) AS Ages

ORDER BY Ages.Age DESC;
```

Ator_+velho	
Name	Age
María Cristina Camilo	105

Figura 37 – Query 19 (Ator_+velho)

– Query 20 (Produtor_c+_filmes): Qual/Quais o(s) produtor(es) com mais filmes?

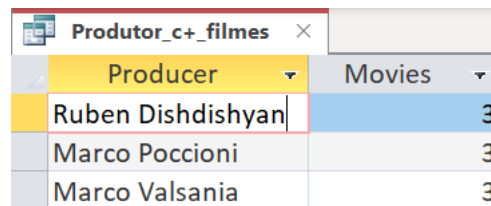
```
SELECT TOP 1 p.primaryName AS Producer, count(m.MovieID) AS Movies

FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN
Movie AS m ON pr.MovieID=m.MovieID

WHERE p.primaryProfession Like "producer"

GROUP BY p.primaryName

ORDER BY count(m.MovieID) DESC;
```



Producer	Movies
Ruben Dishdishyan	3
Marco Poccioni	3
Marco Valsania	3

Figura 38 – Query 20 (Produtor_c+_filmes)

– Query 21 (Ator_c+_filmes): Qual o ator com mais filmes?

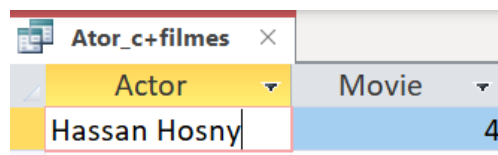
```
SELECT TOP 1 p.primaryName AS Actor, count(m.MovieID) AS Movie

FROM (Person AS p INNER JOIN Principals AS pr ON p.PersonID=pr.PersonID) INNER JOIN
Movie AS m ON pr.MovieID=m.MovieID

WHERE p.primaryProfession="actor"

GROUP BY p.primaryName

ORDER BY count(m.MovieID) DESC;
```

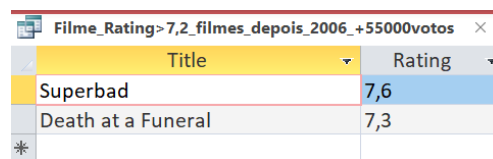


Actor	Movie
Hassan Hosny	4

Figura 39 – Query 21 (Ator_c+_filmes)

- Query 22 (Filme_Rating>7,2_filmes_depois_2006_+55000votos): Qual/Quais é/são o(s) nome(s) do filme(s) que tem rating superior a 7,2, realizados depois de 2006 e que contenham um número de votos por parte dos utilizadores superior a 55000? Apresentar os resultados por ordem decendente.

```
SELECT m.primaryTitle AS Title, r.averageRating AS Rating
FROM Movie AS m INNER JOIN Rating AS r ON m.MovieID=r.MovieID
WHERE (r.averageRating > "7,2") AND (m.startYear>2006) AND (r.numVotes>55000)
ORDER BY r.averageRating DESC;
```

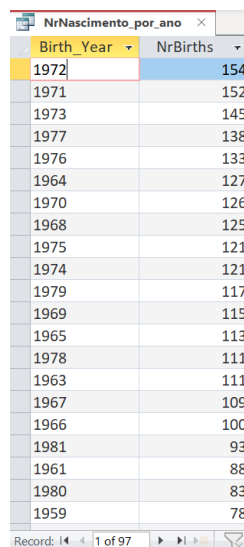


Title	Rating
Superbad	7,6
Death at a Funeral	7,3

Figura 40 – Query 22 (Filme_Rating>7,2_filmes_depois_2006_+55000votos)

- Query 23 (NrNascimento_por_ano): Qual o número de nascimentos por ano?

```
SELECT birthYear AS Birth_Year, count(PersonID) AS NrBirths
FROM Person
WHERE birthYear is not null
GROUP BY birthYear
ORDER BY count(PersonID) DESC , birthYear DESC;
```



Birth_Year	NrBirths
1972	154
1971	152
1973	145
1977	138
1976	133
1964	127
1970	126
1968	125
1975	121
1974	121
1979	117
1969	115
1965	113
1978	111
1963	111
1967	109
1966	100
1981	93
1961	88
1980	83
1959	78

Figura 41 - Query 23 (NrNascimento_por_ano)

- Query 24 (Papel_Elenco): Por ID de título, quais são as funções de cada pessoa e que personagens são interpretadas?

```
SELECT DISTINCT Movie.MovieID AS ID, Movie.primaryTitle AS Title, Person.primaryName
AS Name, Category.role AS Role, Principals.characters AS [Character]
```

```
FROM Movie INNER JOIN (Principals INNER JOIN (Category INNER JOIN Person ON
Category.PersonID = Person.PersonID) ON Principals.PersonID = Person.PersonID) ON
Principals.MovieID = Movie.MovieID;
```

ID	Title	Name	Role	Character
tt0111414	A Thin Life	Barry Porter-Rob	producer	
tt0111414	A Thin Life	Tommy Dysart	actor	["The Man"]
tt0119231	The Junior Defenders	Ally Sheedy	actress	["Jill Fields"]
tt0119231	The Junior Defenders	Brian O'Halloran	actor	
tt0119231	The Junior Defenders	Brook Wetzel	producer	
tt0119231	The Junior Defenders	Brooke Ciardelli	producer	
tt0119231	The Junior Defenders	Cabot Orton	producer	
tt0119231	The Junior Defenders	Jason David Frank	actor	["Tommy Keen"]
tt0119231	The Junior Defenders	Justin Henry	actor	["Bobbi"]
tt0119231	The Junior Defenders	Keith Spiegel	self	
tt0119231	The Junior Defenders	Mark Sasahara	cinematograph	
tt0119231	The Junior Defenders	Rick Giachino	actor	
tt0159790	The Third Vehicle	Ernie Gehr	self	
tt0179803	Angels with Angles	David Proval	actor	["Howie Gold"]
tt0179803	Angels with Angles	Frank Gorshin	actor	["George Burns", "Shelleen"]
tt0179803	Angels with Angles	Holly Fink	self	
tt0179803	Angels with Angles	Jamie Mitchell	writer	
tt0179803	Angels with Angles	Julie Carmen	actress	["Graciella"]
tt0179803	Angels with Angles	Mark Pietri	writer	
tt0179803	Angels with Angles	Melissa Wegman	producer	
tt0179803	Angels with Angles	Paul Greenberg	director	

Figura 42 - Query 24 (Papel_Elenco)

- Query 25 (Nome_filme+peessoa_em_nome_personagem_Beauty): Nome do filme, nome da pessoa em que o nome da personagem é igual a "Beauty"?

```
SELECT m.primaryTitle AS Title, p.primaryName AS Name, l.characters AS Characters
FROM (Movie AS m INNER JOIN Principals AS l ON m.MovieID=l.MovieID) INNER JOIN
Person AS p ON l.PersonID=p.PersonID
WHERE l.characters='["Beauty"]';
```

Title	Name	Characters
Frank & Fearless	Khanyi Mbau	["Beauty"]
*		

Figura 43 – Query 25 (Nome_filme+peessoa_em_nome_personagem_Beauty)

- Query 26 (People_Idades): Qual o nome, data de nascimento, data de falecimento (caso tenha ocorrido), idade das pessoas (no caso de terem morrido, a idade que tinham) e as profissões das pessoas da Base de Dados?

```
(SELECT primaryName AS Name, birthyear AS Birth_Year, deathyear AS Death_Year,
(deathyear-birthyear) AS Age, primaryProfession AS Profession
```

```
FROM Person
```

```
WHERE (deathyear-birthyear) is not null)
```

```
UNION (SELECT Ages.primaryName, Ages.birthyear, Ages.deathyear, Ages.Age,
Ages.primaryProfession
```

```
FROM (SELECT primaryName, birthyear, deathyear, (deathyear-birthyear) AS Idade_Morte,
(2023-birthyear) AS Age, primaryProfession FROM Person
```

```
WHERE (deathyear-birthyear) is null) AS Ages)
```

```
ORDER BY Age DESC;
```

Name	Birth_Year	Death_Year	Age	Profession
María Cristina Camilo	1918		105	actress
C.Y. Lee	1915	2018	103	writer
Renée Le Calm	1918	2019	101	actress
Danielle Darrieux	1917	2017	100	actress,soundtrack
A.K. Hangal	1914	2012	98	actor
Laura Cardoso	1927		96	actress
Cloris Leachman	1926	2021	95	actress,soundtrack
Fulvio Lucisano	1928		95	producer,writer,production_manager
Asunción Balaguer	1925	2019	94	actress
Franciszek Pieczka	1928	2022	94	actor
Fyvush Finkel	1922	2016	94	actor,soundtrack
Guillermo Murray	1927	2021	94	actor,writer,director
James Hong	1929		94	actor,producer,writer
Maria Charles	1929		94	actress
Mickey Rooney	1920	2014	94	actor,soundtrack,producer
Miklós Jancsó	1921	2014	93	director,writer,actor
Paul Vecchiali	1930	2023	93	director,writer,editor
Robert Wagner	1930		93	actor,producer,soundtrack
Safa Önal	1930		93	writer,director,producer
Stanislaw Brudny	1930		93	actor
Claire Bloom	1931		92	actress

Figura 44 - Query 26 (People_Idades)

- Query 27 (5_anos_melhor_média_pontuação_users_entre_2006_2022): Quais são os 5 anos com melhor média em termos de pontuação por parte dos utilizadores entre os anos 2006 e 2022?

```
SELECT TOP 5 Movie.startYear AS [Year], Avg(Rating.averageRating) AS Average
```

```
FROM Rating INNER JOIN Movie ON Rating.MovieID = Movie.MovieID
```

```
WHERE (((Movie.startYear)>2006 And (Movie.startYear)<2022))
```

```
GROUP BY Movie.startYear
```

```
ORDER BY Avg(Rating.averageRating) DESC;
```

Year	Average
2016	7,1
2015	6,6
2021	6,45
2011	6,34285714285714
2018	6,04

Figura 45 – Query 27 (5_anos_melhor_média_pontuação_users_entre_2006_2022)

- Query 28 (Idade_ator+64): Qual a idade do(s) ator(es) cuja idade seja superior a 64 anos, que participe no filme que contenha o número 2 no seu titulo, o respetivo rating, o número de votos, que o trabalho seja nulo e que o seu CatID esteja compreendido entre 799 e 7000?

```
SELECT c.CatID AS ID, p.primaryName AS Name, (deathyear-birthyear) AS Age, c.role AS
Role, l.job AS Job, m.primaryTitle AS Title, r.averageRating AS Rating, sum(r.numVotes) AS Votes

FROM (((Category AS c INNER JOIN Person AS p ON c.PersonID=p.PersonID) INNER JOIN
Principals AS l ON p.PersonID=l.PersonID) INNER JOIN Movie AS m ON l.MovieID=m.MovieID)
INNER JOIN Rating AS r ON m.MovieID=r.MovieID

WHERE c.CatID between 799 AND 7000

AND (deathyear-birthyear) > 64

AND c.role='actor'

AND l.job is null

AND m.primaryTitle Like "2*"

GROUP BY c.CatID, p.primaryName, (deathyear-birthyear), c.role, l.job, m.primaryTitle,
r.averageRating;
```

ID	Name	Age	Role	Job	Title	Rating	Votes
1868	Koço Devole	75	actor		2 Fingers Honey	4,0	658

Figura 46 - Query 28 (Idade_ator+64)