

EXERCÍCIO 1

a. Nesta tabela, utilizamos o **FLOAT** para declarar o salário, porque esta é uma forma de representar um número não inteiro em uma tabela. Explique os demais comandos que estão nessa query.

RESPOSTA:

CREATE TABLE: é usado para criar uma tabela

VARCHAR: strings de no máximo n caracteres

PRIMARY KEY: identificador único de cada item

NOT NULL: indica que o item não pode ser null

FLOAT: indica números não inteiros

DATE: representa uma data (YYYY-MM-DD)

b. O comando **SHOW** é bem útil para nos prover informações sobre bancos, tabelas, e mais. Utilize os comandos: **SHOW DATABASES** e **SHOW TABLES**. Explique os resultados.

RESPOSTA:

SHOW DATABASES: mostra os bancos de dados presentes no sistema

SHOW TABLES: mostra as tabelas presentes no banco de dados, (no meu teste não rodou, deu erro)

c. O comando **DESCRIBE** pode ser usado para ver estrutura de uma tabela. Utilize o comando **SHOW Actor** e explique os resultados.

RESPOSTA:

DESCRIBE é usado como um atalho para o comando SHOW COLUMNS FROM.

SHOW Actor: Não funciona por que eu deveria usar o DESCRIBE.

EXERCÍCIO 2

a. Escreva uma query que crie a atriz **Glória Pires**, com o id **002**, salário R\$1.200.000 e data de nascimento 23 de Agosto de 1963

RESPOSTA:

INSERT INTO Actor (id, name, salary, birth_date, gender)

VALUES(

“002”,

“Glória Pires”,

1200000,

“1964-08-23”,

“female”

);

b. Escreva uma query que tente adicionar um outro elemento a tabela com o mesmo id do item anterior **002**. Isso gerará um erro. Anote a mensagem de erro, traduza (pode usar o Google Tradutor diretamente) e explique porque esse erro aconteceu.

RESPOSTA:

Error Code: 1062. Entrada duplicada para a chave primária 002.

O erro ocorreu porque tentei fazer uma query com uma PK repetida, ela precisa ser única.

c. Error Code: 1136. Column count doesn't match value count at row 1

Código de erro: 1136. A contagem de colunas não corresponde à contagem de valores na linha 1

```
INSERT INTO Actor (id, name, salary, birth_date, gender)
```

```
VALUES(
```

```
“003”,
```

```
“Fernanda Montenegro”,
```

```
300000,
```

```
“1929-10-19”,
```

```
“female”
```

```
);
```

d. Error Code: 1364. Field 'name' doesn't have a default value

Código de erro: 1364. O campo 'name' não tem um valor padrão

```
INSERT INTO Actor (id, name, salary, birth_date, gender)
```

```
VALUES(
```

```
“004”,
```

```
“José Mayer”,
```

```
400000,
```

```
“1949-04-18”,
```

```
“male”
```

```
);
```

e. Error Code: 1292. Incorrect date value: '1950' for column 'birth_date' at row 1

Código do erro: 1292. Valor incorreto da data: '1950' para a coluna 'data_de_nascimento' na linha 1

```
INSERT INTO Actor (id, name, salary, birth_date, gender)
```

```
VALUES(
```

```
“005”,  
“Juliana Paes”,  
719333.33,  
“1979-03-26”,  
“female”  
);
```

```
f.  
  
INSERT INTO Actor (id, name, salary, birth_date, gender)  
VALUES(  
“006”,  
“Adriana Esteves”,  
8000000,  
“1968-02-20”,  
“female”  
);
```

```
INSERT INTO Actor (id, name, salary, birth_date, gender)  
VALUES(  
“007”,  
“Fernanda Lima”,  
1500000,  
“1983-11-30”,  
“female”  
);
```

EXERCÍCIO 3

a. Escreva uma query que retorne todas as informações das atrizes

RESPOSTA:

```
SELECT * from Actor WHERE gender = “female”;
```

b. Escreva uma query que retorne o salário do ator com o nome **Tony Ramos**

RESPOSTA:

```
SELECT salary from Actor WHERE name = “Tony Ramos”;
```

c. Escreva uma query que retorne todas as informações que tenham o **gender** com o valor **"invalid"**. Explique o resultado.

RESPOSTA:

```
SELECT * Actor WHERE gender = "invalid";
```

d. Escreva uma query que retorne o id, nome e salário de todos que tenham o salário com o valor máximo de R\$500.000

RESPOSTA:

```
SELECT id, name, salary from Actor WHERE salary >= 500000;
```

e. Tente usar as queries abaixo. Você vai reparar que elas vão gerar um erro. Anote as mensagens de erro, traduza (pode usar o Google Tradutor diretamente) e explique porque esses erros aconteceram. Por fim, corrija individualmente cada query para que funcione, teste o comando e anote-o também como resposta

```
SELECT id, nome from Actor WHERE id = "002"
```

RESPOSTA:

Error Code: 1054. Unknown column 'nome' in 'field list'

Código de erro: 1054. Não existe coluna desconhecida como 'nome' na 'lista de campos'

Esse erro aconteceu por que não há nenhuma entrada declarada como 'nome', foi declarada como 'name'.

```
SELECT id, name from Actor WHERE id = "002"
```

EXERCÍCIO 4

a. Explique com as suas palavras a query acima

```
SELECT * FROM Actor
```

```
WHERE (name LIKE "A%" OR name LIKE "J%") AND salary > 300000
```

RESPOSTA:

é uma solicitação para mostrar tudo na tabela Actor que no parâmetro 'name' começa com A ou que comece com J e tenha salário maior que 300000.

b. Escreva uma query com os atores que não comecem com a letra "A" e tenham o salário maior do que R\$350.000,00

RESPOSTA:

```
SELECT * FROM Actor
```

```
WHERE (name NOT LIKE "A%") AND salary > 350000
```

c. Escreva uma query com os atores que possuam "G" ou "g" em qualquer parte do nome.

RESPOSTA:

```
SELECT * FROM Actor
```

```
WHERE name LIKE "%G%" OR name LIKE "%g%"
```

d. Escreva uma query com os atores que tenham a letra “a” ou “A” ou “g” ou “G” no nome e o salário entre R\$350.000,00 e R\$900.000,00

RESPOSTA:

```
SELECT * FROM Actor
```

```
WHERE
```

```
(name LIKE "%g%" OR name LIKE "%G%" OR name LIKE "%a%" OR name LIKE "%A%")
```

```
AND salary BETWEEN 350000 AND 900000;
```

EXERCÍCIO 5

a. Escreva a query que cria essa tabela. Para sinopse, utilize o tipo TEXT, pesquise sobre ele se precisar. Explique a query resumidamente.

RESPOSTA:

```
CREATE TABLE Movie (  
id VARCHAR(255) PRIMARY KEY,  
title VARCHAR(255) NOT NULL UNIQUE,  
synopsis TEXT NOT NULL,  
release_Date DATE NOT NULL,  
rating INT NOT NULL  
);
```

O tipo TEXT armazena dados não Unicode de comprimento variável na página de código do servidor e com um comprimento máximo de cadeia de caracteres de $2^{31}-1$ (2.147.483.647). Ou seja, é usado para dados que podem conter um comprimento muito grande e ultrapassa o tamanho suportado pelo VARCHAR.

b. RESPOSTA:

```
CREATE TABLE Movie (  
"001",  
"Se Eu Fosse Você",  
"Cláudio e Helena são casados há muitos anos e enfrentam a rotina do casamento. Um dia eles são atingidos por um  
fenômeno inexplicável e trocam de corpos",  
"2006-01-06",  
7  
);
```

c. RESPOSTA:

```
CREATE TABLE Movie (  
"002",  
"Doce de mãe",
```

“Dona Picucha, uma animada senhora de 85 anos, sempre causa grandes confusões. A vida dela e dos seus quatro filhos sofre uma reviravolta depois que Zaida, empregada e amiga de Dona Picucha, anuncia que vai se casar e não poderá mais morar com ela”,

“2012-12-27”,

10

);

d. RESPOSTA:

```
CREATE TABLE Movie (
```

“003”,

“Dona Flor e Seus Dois Maridos”,

“Dona Flor é uma sedutora professora de culinária casada com Vadinho, que só quer saber de farras e jogatina nas boates.

A vida de abusos acaba por acarretar sua morte precoce.”,

“2017-11-02”,

8

);

e. RESPOSTA:

```
CREATE TABLE Movie (
```

“004”,

“O quatrilho”,

“Em 1910, em uma comunidade rural de imigrantes italianos no Rio Grande do Sul, dois casais dividem a mesma casa. O tempo faz com que uma das esposas se interesse pelo marido da outra. Ambos decidem fugir e recomeçar uma vida nova, deixando para trás seus parceiros...”,

“1995-10-20”,

10

);

EXERCÍCIO 6

Escreva uma query que:

a. Retorne o id, título e avaliação a partir de um id específico;

RESPOSTA:

```
SELECT id, title, rating FROM Movie WHERE id = “002”;
```

b. Retorne um filme a partir de um nome específico;

RESPOSTA:

```
SELECT * FROM Movie WHERE name = “O quatrilho”;
```

c. Retorne o id, título e sinopse dos filmes com avaliação mínima de 7

RESPOSTA:

```
SELECT id, title, synopsis FROM Movie WHERE rating > 7;
```

EXERCÍCIO 7

Escreva uma query que:

a. Retorna um filme cujo título contenha a palavra vida

RESPOSTA:

```
SELECT * FROM Movie  
WHERE title LIKE "%vida%";
```

b. Realize a pesquisa de um filme, ou seja: pesquise se o termo de busca está contido no título ou na sinopse. Utilize qualquer TERMO DE BUSCA para exemplificar.

RESPOSTA:

```
SELECT * FROM Movie  
WHERE title LIKE "%mãe%" OR  
synopsis LIKE "%quadrilho%";
```

c. Procure por todos os filmes que já tenham lançado

RESPOSTA:

```
SELECT * FROM MOVIE  
WHERE release_date < "2020-06-08";
```

d. Procure por algum filme que já tenha lançado, com o termo de busca contido no título ou sinopse e com a avaliação maior do que 7.

RESPOSTA:

```
SELECT * FROM MOVIE  
WHERE release_date < "2020-05-04" AND  
(title LIKE "%mãe%" OR  
synopsis LIKE "%casal%") AND rating > 7;
```