



BASE DE DADOS

Trabalho Prático n.º 2

Docente: Davide Dias

Grupo 14: Andre Pereira (a40627)
Diogo Coelho (a40708)

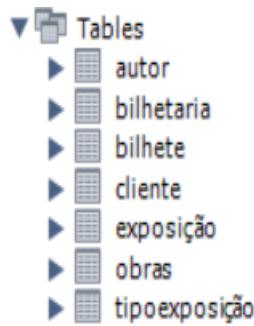
Bernardo Nogueira (a47334)
Bernardo Ferreira (a48524)

1 - Create Table

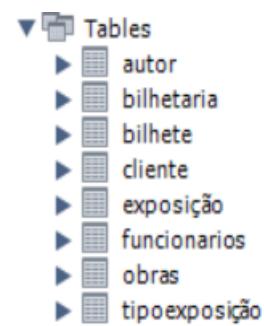
```
CREATE TABLE IF NOT EXISTS funcionarios (
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(45) NOT NULL);
```

Nesta query é criada uma tabela **funcionarios**, caso ainda não exista, com os atributos: **id**, como chave primária, com tipo de dados inteiro(INT), não nulo (NOT NULL) e **nome**, com tipo de dados varchar de 45 caracteres (VARCHAR(45)), não nulo(NOT NULL).

Antes:



Depois:



2 - Alter Table

```
ALTER TABLE funcionarios MODIFY nome VARCHAR(30);
```

Nesta query é realizada a edição de uma tabela já existente, **funcionarios**, onde se altera o tamanho do atributo **nome**, tornando-o um varchar de 30 caracteres (VARCHAR(30)).

Antes:

Table: **funcionarios**

Columns:

id	int(11) AI PK
nome	varchar(45)

Depois:

Table: **funcionarios**

Columns:

id	int(11) AI PK
nome	varchar(30)

3 - Insert Into

```
INSERT INTO `autor` (`idAutor`, `nomeAutor`) VALUES ('06', 'Joaquim Alberto');
```

Nesta query é adicionado um novo registo à tabela **autor**.

	idAutor	NomeAutor
1	Joaquim Andrade	
2	Andre Pereira	
3	Bernardo Noqueira	
4	Diooo Coelho	
5	Pedro Montanha	
NULL	NULL	

	idAutor	NomeAutor
1	Joaquim Andrade	
2	Andre Pereira	
3	Bernardo Noqueira	
4	Diooo Coelho	
5	Pedro Montanha	
6	Joaquim Alberto	
NULL	NULL	

Antes: **Depois:**

4 - Select com condição

```
SELECT nome AS 'Nome Cliente', idade
FROM cliente WHERE (idade >= 18 and idade <30);
```

Nesta query é realizada um seleção dos clientes entre os 18 e 30 anos(idade >= 18 and idade <30) , retornando o seu nome e idade .

Output esperado:

	Nome Cliente	idade
	Joao ribeira	21
	Helder Ribeiro	20
	Bruna Fonseca	25

5 - Select com um Join de duas tabelas

```
SELECT exposição.idExposição AS 'idExposição', exposição.Sala AS 'Sala ', tipoexposição.idTipoExposição AS 'idTipoExposição', tipoexposição.Tipo
```

```
AS 'Tipo' FROM exposição LEFT JOIN tipoexposição ON exposição.idTipoExposição = tipoexposição.idTipoExposição;
```

Nesta query une-se a tabela exposição com a tabela tipoexposição, conseguindo assim saber através do idexposição o tipo de exposição a que pertence.

Output esperado:

	<u>idExposição</u>	Sala	<u>idTipoExposição</u>	Tipo
	20	215	15	Arte rural
	50	216	16	Arte urbana
	69	217	17	Arte realismo
	70	218	18	Arte futurista
	100	220	19	Arte surrealismo

6 - Select com join de 3 tabelas

```
SELECT bilhete.idBilhete AS 'idBilhete', bilhetaria.idcliente AS 'idCliente', bilhete.CC as 'CC' FROM bilhete
RIGHT JOIN bilhetaria ON bilhetaria.idCliente = bilhete.idCliente LEFT JOIN cliente ON bilhete.CC =
cliente.CC;
```

Com esta query é nos dado acesso ao idBilhete , idCliente e o CC das Pessoas que compraram um bilhete.

	<u>idBilhete</u>	<u>idCliente</u>	CC
	2222	200	30711833
	2500	300	4035678
	2700	500	80934567
	2900	700	12345678
	4000	1000	23496543

Output esperado:

7 - Select com join de 3 tabelas com ordenação de dados

```
SELECT bilhete.idBilhete AS 'idBilhete', bilhetaria.idcliente AS 'idCliente', bilhete.CC as 'CC' FROM bilhete
RIGHT JOIN bilhetaria ON bilhetaria.idCliente = bilhete.idCliente LEFT JOIN cliente ON bilhete.CC =
cliente.CC order by CC;
```

Com esta query é nos dado acesso ao idBilhete , idCliente e o CC das Pessoas que compraram um bilhete.

Esta informação está ordenada pelo CC.

Output esperado:

	idBilhete	idCliente	CC
	2900	700	12345678
	4000	1000	23496543
	2222	200	30711833
	2500	300	40356787
	2700	500	80934567

9 - Select com uma subquery

```
SELECT bilhetaria.idCliente AS 'idCliente', bilhetaria.Preço as 'Preço' FROM bilhetaria WHERE
bilhetaria.Preço = any(SELECT bilhetaria.Preço FROM bilhetaria WHERE Preço >= 6.00);
```

Esta query mostra-nos o idcliente e o preço do bilhete desse cliente de acordo com o preço que ele pagou.

Output esperado:

	idCliente	Preço
	200	9.99
	500	6.00
	700	7.00
	NULL	NULL