INSTITUTO DO EMPREGO
E FORMAÇÃO PROFISSIONAL

AEP

cesae digital

Centro para o Desenvolvimento
de Competências Digitais

PROGRAMAÇÃO SQL

DDL | DML | DQL

Helder Rodrigo Pinto

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DDL
- DML
- DQL

SQL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- A Linguagem SQL (Structured Query Language) é essencialmente uma linguagem de pesquisa (query) associada aos SGBD.
- É também uma linguagem declarativa (o programador especifica aquilo que quer, em vez de especificar como se pode obter aquilo que se quer).
- Muitas das características originais da SQL foram inspiradas na álgebra relacional.

ÂMBITO DA LINGUAGEM SQL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DDL (Data Definition Language) – Definição de Dados
- DML (Data Manipulation Language) – Linguagem de manipulação de dados
- DQL (Data Query Language) – Linguagem de Pesquisa de Dados

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DDL
- DML
- DQL

DDL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- CREATE cria um objeto (DATABASE, TABLE, INDEX, VIEW, …).
- DROP apaga um objeto da base de dados.
- ALTER permite alterar um objeto, por exemplo, adicionando uma coluna a uma tabela existente.

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma Base de Dados

```
CREATE DATABASE NOME;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Eliminar uma Base de Dados

```
DROP DATABASE NOME;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela

MySQL

```
CREATE TABLE TipoUtilizador(
   idTipo INT AUTO_INCREMENT PRIMARY KEY,
   descricao VARCHAR(50)
);
```

SQLServer

```
   idTipo INT IDENTITY(1,1) PRIMARY KEY,
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Eliminar uma tabela

```
DROP TABLE TipoUtilizador;
```

TIPOS DE DADOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- **SMALINT:** número inteiro desde -32768 até 32767 (2bytes)
- **INTEGER, INT:** valores inteiros desde -2147483648 até 2147483647 (4bytes)
- **BIGINT:** número inteiro com desde -9.223.372.036.854.775.808 até 9.223.372.036.854.775.807 (8bytes)

- **FLOAT:** número fracionário de 4 bytes
- **DOUBLE:** número fracionário de 8 bytes
- **DECIMAL, DEC, NUMERIC:** Número fracionário desempacotado: DECIMAL(M,D)

TIPOS DE DADOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- **VARCHAR:** caracteres alfanuméricos
  - Implica que se indique o número máximo de caracteres que terá esse atributo. Ex: VARCHAR(50)

- **DATE:** Valores de Data
- **DATETIME:** Valores de Data e Hora
- **TIMESPAN:** Duração

- **BIT ou BOOL:** um número inteiro que pode ser 0 ou 1.

ATRIBUTOS DOS CAMPOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- **NOT NULL** – Não permite que o valor de um campo seja nulo ou inexistente
- **UNIQUE** – Não permite que o valor de um campo seja repetido
- **DEFAULT** *valor* – Define um valor por omissão que é adicionado mas que pode ser alterado
- **CHECK** – Faz validação aos valores de um campo de acordo com uma expressão
- **AUTO INCREMENT** – Incremento automático do MySQL (começa em 1 e é acrescentado sempre +1)
- **IDENTITY(1,1)** – Incremento automático do SQLServer (começa em 1 e é acrescentado sempre +1)
- **PRIMARY KEY** – Usado para identificar um campo único como chave primária.
- **FOREIGN KEY** – Usado para identificar um campo como chave estrangeira – tem de referenciar a tabela mãe.

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT PRIMARY KEY,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT
   );
```

APENAS PODE SER USADO PARA CASOS DE TABELAS COM CHAVE PRIMÁRIA CONSTITUIDA POR UM SÓ CAMPO

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT,
   PRIMARY KEY (idUtilizador)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária definindo nome para a chave

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária e chave estrangeira

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT REFERENCES TipoUtilizador (idTipo),
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador)
);
```

Não compatível em MySQL

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária e chave estrangeira

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT FOREIGN KEY REFERENCES TipoUtilizador
   (idTipo),
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador)
);
```

Não compatível em MySQL

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária e chave estrangeira

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   FOREIGN KEY (tipo) REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com chave primária e chave estrangeira

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50),
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   CONSTRAINT FK_Utilizador_Tipo FOREIGN KEY (tipo)
   REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com restrições

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50) UNIQUE,
   nome VARCHAR(50),
   password VARCHAR(30),
   tipo INT,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   CONSTRAINT FK_Utilizador_Tipo FOREIGN KEY (tipo)
   REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com restrições

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50) UNIQUE,
   nome VARCHAR(50),
   password VARCHAR(30) NOT NULL,
   tipo INT,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   CONSTRAINT FK_Utilizador_Tipo FOREIGN KEY (tipo)
   REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com restrições

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50) UNIQUE,
   nome VARCHAR(50),
   password VARCHAR(30) NOT NULL,
   tipo INT DEFAULT 1,
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   CONSTRAINT FK_Utilizador_Tipo FOREIGN KEY (tipo)
   REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar uma tabela com restrições

```
CREATE TABLE Utilizador(
   idUtilizador INT AUTO_INCREMENT,
   eMail VARCHAR(50) UNIQUE,
   nome VARCHAR(50),
   password VARCHAR(30) NOT NULL,
   tipo INT DEFAULT 1 CHECK(tipo > 0),
   CONSTRAINT PK_Utilizador PRIMARY KEY (idUtilizador),
   CONSTRAINT FK_Utilizador_Tipo FOREIGN KEY (tipo)
   REFERENCES TipoUtilizador (idTipo)
);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Alterar um campo de uma tabela

MySQL

```
ALTER TABLE  Utilizador
MODIFY COLUMN password VARCHAR(50);
```

SQLServer

```
ALTER TABLE  Utilizador
ALTER COLUMN password VARCHAR(50);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Adicionar um campo a uma tabela

```
ALTER TABLE Cliente
ADD dataNascimento DATE;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Apagar um campo a uma tabela

```
ALTER TABLE Cliente
DROP COLUMN dataNascimento;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Alterar um campo de uma tabela (adicionar uma chave primária posteriormente)

```
ALTER TABLE  Utilizador
ADD CONSTRAINT PK_Utilizador
                PRIMARY KEY (idUtilizador);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Alterar um campo de uma tabela (apagar chave primária posteriormente)

MySQL

```
ALTER TABLE  Utilizador
DROP PRIMARY KEY;
```

SQLServer

```
ALTER TABLE  Utilizador
DROP CONSTRAINT PK_Utilizador;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Alterar um campo de uma tabela (adicionar uma chave estrangeira posteriormente)

```
ALTER TABLE  Utilizador
ADD CONSTRAINT FK_Utilizador_Tipo
        FOREIGN KEY (tipo) REFERENCES
                TipoUtilizador (idTipo);
```

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DDL
- DML
- DQL

DML

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- INSERT é usada para inserir um registo numa tabela.
- UPDATE para alterar os valores de um registo de uma tabela existente.
- DELETE permite remover registos de uma tabela.

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Inserir dados numa tabela

```
INSERT INTO TipoUtilizador (descricao)
  VALUES ('Administrador');
INSERT INTO TipoUtilizador (descricao)
  VALUES ('Utilizador');
INSERT INTO Utilizador (eMail, nome, password, tipo)
  VALUES ('helder@mail.pt', 'Helder', 'helder', 1);
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Atualizar dados de um registo numa tabela

```
UPDATE Cliente
SET telefone = '912345678', cidade= 'Porto'
WHERE codCliente=1;
```

- Se não existir condição (WHERE) atualiza todos os registos da tabela

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Eliminar um registo de uma tabela

```
DELETE FROM Cliente
WHERE codCliente='1';
```

- Se não existir condição (WHERE) remove todos os registos da tabela

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Eliminar todos os registos de uma tabela

```
DELETE *
FROM Cliente;
```

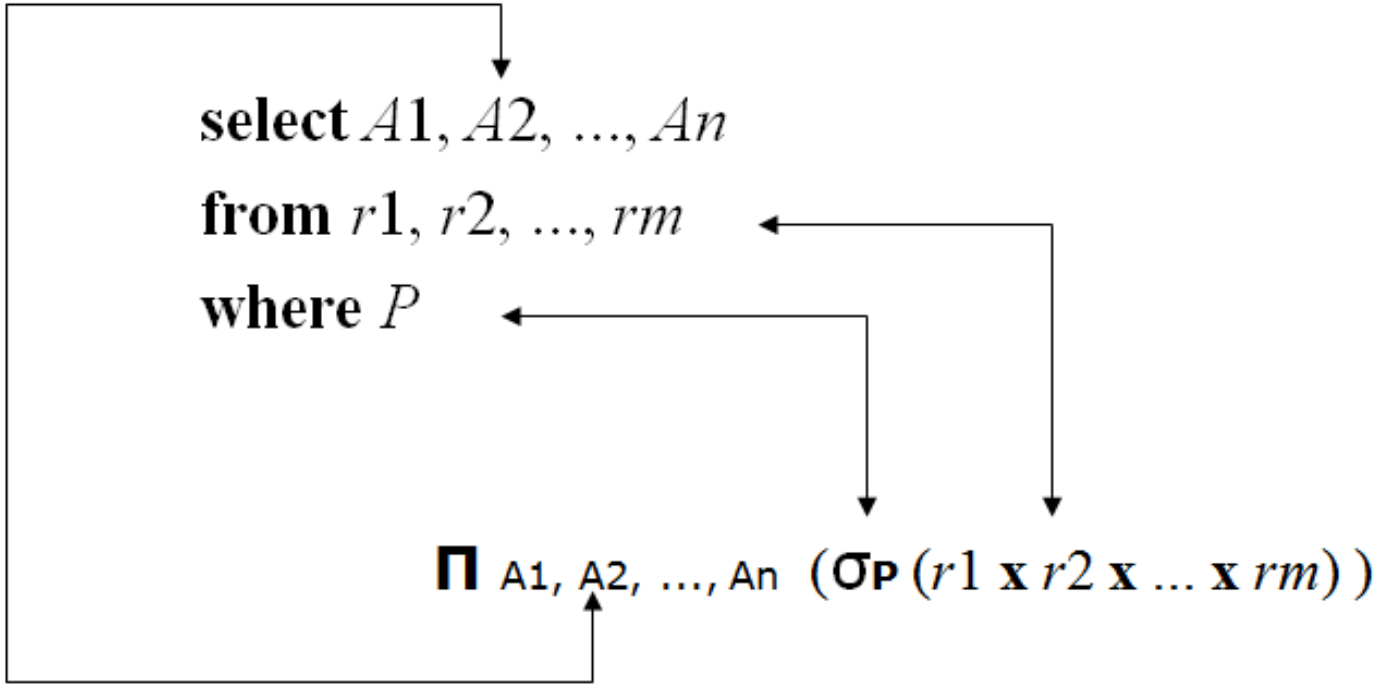
OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DDL
- DML
- DQL

DQL

- O SELECT corresponde a uma expressão de álgebra relacional, envolvendo seleções, projeções e produtos cartesianos.



DQL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- ESTRUTURA

```
SELECT [ ALL | DISTINCT ] [ AVG | SUM | COUNT | MAX | MIN | TOP ] <select_list>
FROM <table_list>
[WHERE <where_expression>]
[GROUP BY <groupby_list>]
[HAVING <having_expression_for_groupby>]
[ORDER BY <orderby_list> [DESC|ASC]]
```

DQL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- SELECT seleciona os atributos a serem mostrados na pesquisa.
- FROM faz o produto cartesiano de todas as tabelas.
- A expressão da cláusula WHERE tem de ser verdadeira para um determinado registo (obtido do produto cartesiano) ser devolvido no resultado.
- A cláusula GROUP BY define o número máximo de linhas do resultado final. Esse número é igual ao número de todos os valores distintos para as colunas da groupby_list. O HAVING pode ser usado agregado ao GROUP BY para definir uma condição.
- A cláusula ORDER BY permite ordenar registos.

DQL

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- DISTINCT força a eliminação de duplicados.
- AVG é uma função que permite o cálculo da média dos valores de um atributo de uma relação.
- SUM é uma função que permite o cálculo da soma de um atributo numérico.
- COUNT é uma função que permite contar quantos valores existem.
- MIN e MAX são funções que retornam o menor ou maior valor existente, respetivamente.
- TOP devolve um único valor, o mais a cima do resultado.

OPERADORES WHERE

cesae digital
Centro para o Desenvolvimento de Competências Digitais

OPERADOR	DESCRIÇÃO
=	Igual (exatamente igual)
<> Ou !=	Diferente
>	Maior que
<	Menor que
>=	Maior ou igual que
<=	Menor ou igual que

OPERADORES WHERE

cesae digital
Centro para o Desenvolvimento de Competências Digitais

OPERADOR	DESCRIÇÃO
LIKE	Igual (de tal forma que)
IN	Especifica valores múltiplos para essa coluna
BETWEEN	Entre
NOT	Não
AND	E
OR	OU
IS [NOT] NULL	Valida se o valor está (ou não) vazio
ALL/ANY	Todos/Pelo menos um

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Fazer pesquisas à Base de Dados

```
SELECT * FROM Utilizador;
```

O * representa todos os atributos.

```
SELECT eMail, nome
FROM Utilizador
WHERE tipo=1;
```

EXERCÍCIO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

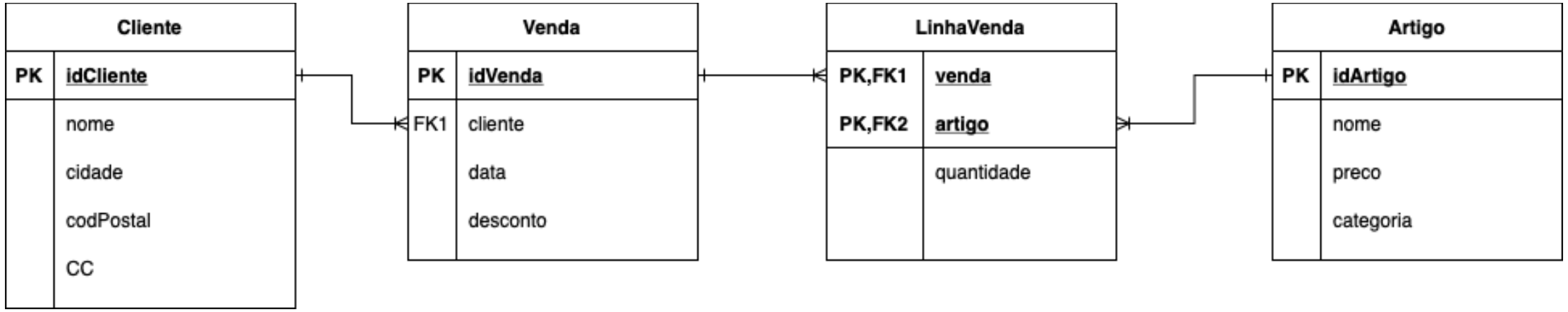
- Considere uma base de dados (âmbito académico) de artigos que são vendidos a clientes.

- Tabelas
  - Cliente
  - Venda
  - LinhaVenda
  - Artigo

EXERCÍCIO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Considere uma base de dados (âmbito académico) de artigos que são vendidos a clientes.



CONSULTA 1

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve todos os artigos da tabela Artigo
  - O * representa todos os atributos

```
SELECT *
FROM Artigo
```

CONSULTA 2

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostrar o idArtigo do Artigo, o nome deste e o preco da tabela Artigo dos artigos mais baratos que 650€

```
SELECT idArtigo, nome, preco
FROM Artigo
WHERE preco<650
```

CONSULTA 3

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra o Artigo e Preço dos artigos cujo preço está entre 100 e 150

```
SELECT nome, preco
FROM Artigo
WHERE preco>=100 AND preco<=150
```

CONSULTA 4

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra o Artigo e Preço dos artigos cujo preço está entre 500 e 1500 (inclusive)

```
SELECT nome, preco
FROM Artigo
WHERE preco BETWEEN 500 AND 1500
```

CONSULTA 5 e 6

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- O OR representa OU e o NOT BETWEEN refere que não está entre...

```
SELECT nome, preco
FROM Artigo
WHERE preco<150 OR preco>250

SELECT nome, preco
FROM Artigo
WHERE preco NOT BETWEEN 150 AND 250
```

CONSULTA 7

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra os artigos e Preço dos Artigos cujo Preço é 100 ou 250

```
SELECT nome, preco
FROM Artigo
WHERE preco IN (100,250)
```

CONSULTA 8

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra todos os artigos cujo nome começa por M
  - O % substitui zero ou vários caracteres
  - O _ substitui um só e só um caracter
  - Apenas funciona em campos varchar

```
SELECT *
FROM Artigo
WHERE nome LIKE 'M%'
```

CONSULTA 9, 10 e 11

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Consultas Ordenadas por preço de Venda (ASC, POR OMISSÃO)

```
SELECT * FROM Artigo
ORDER BY preco ASC

SELECT * FROM Artigo
ORDER BY preco DESC

SELECT * FROM Artigo
WHERE preco >500
ORDER BY preco DESC
```

CONSULTA 12

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Conta quantos artigos existem
  - O AS permite mudar o nome (temporariamente) de uma tabela ou coluna

```
SELECT COUNT(nome) AS TotalArtigos
FROM Artigo
```

CONSULTA 13

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Faz a soma de todos os preços dos Artigos

```
SELECT SUM(preco) AS SomaPV
FROM Artigo
```

CONSULTA 14

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra a média e todos os preços dos Artigos

```
SELECT AVG(preco) AS MediaPV
FROM Artigo
```

CONSULTA 15 e 16

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Apresenta o máximo e o mínimo, respetivamente, Preço de todos os artigos

```
SELECT MAX(preco) AS MaximoPV
FROM Artigo
```

```
SELECT MIN(preco) AS MinimoPV
FROM Artigo
```

CONSULTA 17

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- O DISTINCT retira informação repetida quando esta existe, e neste caso mostra apenas as várias cidades onde existem clientes, não mostrando o nome das cidades repetido

```
SELECT DISTINCT cidade
FROM Cliente
```

CONSULTA 18

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Apresenta para cada Cliente: o seu nome nome e a informação sobre as Vendas registadas: data e artigos
  - **EQUI JOIN**: cruza informação de várias tabelas. Usa-se ligando os campos chave primária e chave estrangeira coincidentes

```
SELECT Cliente.nome, Venda.data, Artigo.nome
FROM Artigo, Cliente, Venda, LinhaVenda
WHERE Cliente.idCliente = Venda.cliente AND
      Venda.idVenda = LinhaVenda.venda AND
      LinhaVenda.artigo = Artigo.idArtigo
```

CONSULTA 19

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Podem ser usadas letras para identificar uma tabela (apenas uma por tabela)

```
SELECT c.nome, v.data, a.nome
FROM Artigo a, Cliente c, Venda v, LinhaVenda l
WHERE c.idCliente = v.cliente AND
      v.idVenda = l.venda AND
      l.artigo = a.idArtigo
```

CONSULTA 20

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- O EQUI JOIN ou INNER JOIN são usados para relacionar informação correspondente de várias tabelas

```
SELECT c.nome, v.data
FROM Cliente c
INNER JOIN Venda v
ON c.idCliente = v.cliente
```

JOIN

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- EQUI JOIN / INNER JOIN (Equivalentes)
  - Relaciona os dados cruzados de ambas tabelas

- FULL JOIN
  - Relaciona os dados cruzados de ambas tabelas e acrescente ambas as tabelas

JOIN

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- LEFT JOIN
  - Relaciona os dados cruzados de ambas tabelas e acrescente a tabela da esquerda

- RIGHT JOIN
  - Relaciona os dados cruzados de ambas tabelas e acrescente a tabela da direita

CONSULTA 21

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Calcula o total de uma Venda de um cliente
- Devolve apenas uma linha por data de venda de cada cliente

```
SELECT c.nome, v.data, SUM(l.quantidade*a.preco) AS Total
FROM Artigo a, Cliente c, Venda v, LinhaVenda l
WHERE c.idCliente = v.cliente AND
      v.idVenda = l.venda AND
      l.artigo = a.idArtigo
GROUP BY c.nome, v.data
```

CONSULTA 22

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas uma linha com a última venda registada

```
SELECT TOP(1) c.nome, v.data
FROM Cliente c, Venda v
WHERE c.idCliente = v.cliente
ORDER BY v.data DESC
```

CONSULTA 22

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas uma linha com a última venda registada

```
SELECT c.nome, v.data
FROM Cliente c, Venda v
WHERE c.idCliente = v.cliente
ORDER BY v.data DESC
LIMIT 1
```

MySQL

CONSULTA 23

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas três linhas com as últimas três vendas registadas

```
SELECT TOP(3) c.nome, v.data
FROM Cliente c, Venda v
WHERE c.idCliente = v.cliente
ORDER BY v.data DESC
```

CONSULTA 23

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas três linhas com as últimas três vendas registadas

```
SELECT c.nome, v.data
FROM Cliente c, Venda v
WHERE c.idCliente = v.cliente
ORDER BY v.data DESC
LIMIT 3
```

MySQL

CONSULTA 24

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas os códigos postais que aparecem mais que duas vezes

```
SELECT c.codPostal
FROM Cliente c
GROUP BY c.codPostal
HAVING COUNT(c.codPostal) > 2;
```

CONSULTA 24 ERRADO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- O **comando está incorrecto** porque a cláusula WHERE é testada linha a linha e numa linha não é possível obter o total de códigos postais.

```
SELECT codPostal
FROM Cliente
WHERE COUNT(codPostal)>2
```

CONSULTA 25

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve para cada código postal, a quantidade daqueles que estão associados a mais do que um cliente

```
SELECT c.codPostal, COUNT(c.codPostal)
FROM Cliente c
GROUP BY c.codPostal
HAVING COUNT(c.codPostal) > 2;
```

CONSULTA 26

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Conta quantos artigos existem numa dada categoria
- Sempre que existe uma função de agregação na cláusula SELECT, todos os restantes atributos da cláusula têm que estar incluídos no GROUP BY.

```
SELECT a.categoria, COUNT(a.nome) AS TotalArtigos
FROM Artigo a
GROUP BY a.categoria
```

- Não seria possível acrescentar o atributo nome à cláusula SELECT.

CONSULTA 27

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve apenas os nomes dos clientes que nunca fizeram compras

```
SELECT c.nome
FROM Cliente c
WHERE c.idCliente NOT IN (SELECT cliente
                          FROM Venda)
```

CONSULTA 28

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve para cada cliente o número de compras associadas

```
SELECT nome, (SELECT COUNT(*)
              FROM Venda
              WHERE Venda.cliente = Cliente.idCliente)
FROM Cliente
```

CONSULTA 29 e 29.1

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Devolve o(s) nome(s) do(s) artigo(s) mais caro(s)

```
SELECT nome
FROM Artigo
WHERE preco >= ALL (SELECT preco
                    FROM Artigo WHERE categoria=”automóvel”)
```

- Devolve o(s) nome(s) do(s) artigo(s) cujo preço não é o menor de todos

```
SELECT nome
FROM Artigo
WHERE preco > ANY (SELECT preco
                   FROM Artigo WHERE categoria=”automóvel”))
```

CONSULTA 30

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- O UNION é usado para cruzar informação de duas ou mais pesquisas

```
SELECT c.cidade FROM Cliente c
UNION
SELECT f.cidade FROM Fornecedor f
```

cesae digital

Centro para o Desenvolvimento
de Competências Digitais

INSTITUTO DO EMPREGO
E FORMAÇÃO PROFISSIONAL

AEP

www.cesae.pt