 Universidade de Trás-os-Montes e Alto Douro

Fase 2 – Protocolo E

Licenciatura em Engenharia Informática

Base de Dados

Paulo Martins – [pmartins@utad.pt](mailto:pmartins@utad.pt)

Daniel Alexandre – [daniel@utad.pt](mailto:daniel@utad.pt)

**Autores**

Eduardo Manuel Afonso Chaves - 70611

João Henrique Constâncio Rodrigues - 70579

Luís André de Marques Pimenta - 70827

Vila Real, Junho de 2021

Índice

[Índice 2](#_Toc72762058)

[1. Introdução 3](#_Toc72762059)

[2. Enquadramento teórico 4](#_Toc72762060)

[2.1. Linguagem SQL 4](#_Toc72762061)

[3. Desenvolvimento do trabalho 5](#_Toc72762062)

[3.1. Inserir registos em cada tabela 5](#_Toc72762063)

[3.2. Resposta ás questões propostas em SQL 9](#_Toc72762064)

[4. Conclusão 11](#_Toc72762065)

[5. Bibliografia 12](#_Toc72762066)

# Introdução

Neste relatório, iremos introduzir certos conceitos de modo a apresentar os assuntos retratados na unidade curricular de Base de Dados, com um objetivo inicial de apresentar todo o conhecimento adquirido em Base de Dados. Conceitos esses relacionados com a Normalização, Arquitetura de base de dados e linguagem SQL.

Antes de iniciarmos a 2ª fase, fizemos algumas alterações no código enviado na 1ª fase de entrga tendo em vista os erros apontados pelo professor (por este motivo enviamos as tabelas que sofreram alterações). Na segunda fase do relatório, introduzimos mais um pouco de linguagem SQL, implementando assim dados e algoritmos para a construção de uma base de dados mais complexa e robusta. Primeiramente, começamos por popular a base de dados para termos informação suficiente para que seja possível testar a base de dados. De seguida, implementamos vários algoritmos em linguagem SQL de modo a especificar e a entender como funciona realmente uma base de dados.

# Enquadramento teórico

## Linguagem SQL

Para entendermos um pouco do que se trata a linguagem SQL vamos partir dos comandos mais básicos que foram usados na primeira e segunda parte do projeto. Em primeiro são criadas a base de dados e as suas tabelas de forma a estruturar o esqueleto do nosso projeto. É dado tipos aos dados que serão colocados nas tabelas (ex: **INTEGER**, **CHAR**, **DATE** e **TIME**), além disso temos que especificar se as colunas obtêm ou não valores nulos. Comandos necessários:

* **CREATE** - Comando necessário para a construção de uma base de dados (**CREATE** **DATABASE**) ou criar uma tabela (**CREATE** **TABLE**);
* **DROP** - Permite remover uma base de dados (**DROP** **DATABASE**) ou remover uma tabela (**DROP** **TABLE**).

De seguida depois de tudo criado temos que inserir valores, neste caso denominados de registos, para podermos testar a base de dados. Isso só será possível a partir dos comandos **INSERT** **INTO** para especificar a tabela e **VALUES** para especificar os resultados. Com as tabelas criadas podemos a qualquer momento aceder aos dados das tabelas, criando assim tabelas novas a partir de outras já existentes, recorrendo ao **SELECT**. Assim de formamos uma base de dados no contexto do trabalho atribuído e de forma geral esses são os comandos mais utilizados nestas duas fases do trabalho

Existe também outros comandos que temos de ter em consideração como o **UPDATE** que permite alterar os valores já existentes nos campos de uma única tabela; o **WHERE** para selecionar um conjunto de linhas; o **AS** serve para renomear uma coluna ou uma tabela ; **MAX/MIN/TOP** para devolver o maior ou menor valor ou o valor do topo da coluna; **COUNT** serve para devolver o números de linhas na qual foi especificado; **GROUP** **BY** serve para agregar funções; **ORDER** **BY** para ordenar de forma ascendente (**ASC**) ou descendente (**DESC**); o **CAST** é uma função que converte um valor de qualquer tipo em um tipo especifico; o **GETDATE** para recebermos a data e hora no exato momento em que corremos a função em formato 'YYYY-MM-DD hh:mm:ss.mmm'; o **DATEDIFF** retorna a diferença entre duas datas.

Por último, iremos falar da SubQuerys que no fundo é uma query dentro de uma query maior, ou de certa forma um **SELECT** dentro de outro. As SubQuerys intervem de modo a tornar certas consultas mais simples que de outra forma seriam completamente complicadas ou impossíveis.

# Desenvolvimento do trabalho

## Inserir registos em cada tabela

use master;

USE Presidenciais;

-------- Inserir registos em cada tabela --------

SELECT \* FROM Pessoas;

INSERT INTO Pessoas(numero\_eleitor, nome, apelido, nacionalidade, data\_nascimento)

VALUES (1, 'Nome1', 'Apelido1', 'Espanhol', '1958-03-07'),--Candidatos

(2, 'Nome2', 'Apelido2', 'Americano', '1969-10-14'),--Candidatos

(3, 'Nome3', 'Apelido3', 'Alemão', '1988-08-22'),--Candidatos

(4, 'Nome4', 'Apelido4', 'Italiano', '1975-01-29'),--Candidatos

(5, 'Nome5', 'Apelido5', 'Brasileiro', '1960-07-02'),--Candidatos

(6, 'Nome6', 'Apelido6', 'Português', '1951-06-13'),--Candidatos

(7, 'Nome7', 'Apelido7', 'Português', '1973-09-18'),--Presidentes

(8, 'Nome8', 'Apelido8', 'Português', '1957-11-03'),--Presidentes

(9, 'Nome9', 'Apelido9', 'Português', '1970-05-12'),--Presidentes

(10, 'Nome10', 'Apelido10', 'Português', '1957-03-07'),--Vogais

(11, 'Nome11', 'Apelido11', 'Português', '1966-10-14'),--Vogais

(12, 'Nome12', 'Apelido12', 'Português', '1968-10-14'),--Vogais

(13, 'Nome13', 'Apelido13', 'Português', '1985-08-22'),--Vogais

(14, 'Nome14', 'Apelido14', 'Português', '1973-01-29'),--Vogais

(15, 'Nome15', 'Apelido15', 'Português', '1964-07-02'),--Vogais

(16, 'Nome16', 'Apelido16', 'Português', '1959-06-13'),--Mandatário

(17, 'Nome17', 'Apelido17', 'Português', '1981-09-18'),--Mandatário

(18, 'Nome18', 'Apelido18', 'Português', '1969-11-03'),--Mandatário

(19, 'Nome19', 'Apelido19', 'Português', '1959-12-06'),--Eleitor

(20, 'Nome20', 'Apelido2O', 'Português', '1981-05-26'),--Eleitor

(21, 'Nome21', 'Apelido21', 'Português', '1959-10-08'),--Eleitor

(22, 'Nome22', 'Apelido22', 'Português', '1969-02-23'),--Eleitor

(23, 'Nome23', 'Apelido23', 'Português', '1979-11-28');--Eleitor

--DELETE FROM Pessoas

SELECT \* FROM Candidatos;

INSERT INTO Candidatos(numero\_candidato)

VALUES (1),

(2),

(3),

(4),

(5),

(6);

--DELETE FROM Candidatos

SELECT \* FROM Escritorios; --Alterações nas Colunas da Tabela!!!--

INSERT INTO Escritorios(numero\_escritorio)

VALUES (1),

(2),

(3),

(4);

--DELETE FROM Escritorios

SELECT \* FROM Candidatos\_Escritorios;

INSERT INTO Candidatos\_Escritorios(numero\_candidato, numero\_escritorio)

VALUES (1, 1),

(2, 2),

(3, 3),

(4, 3),

(5, 4),

(6, 4);

--DELETE FROM Candidatos\_Escritorios

SELECT \* FROM Presidentes;

INSERT INTO Presidentes(numero\_presidente, idade)

VALUES (7, 47),

(8, 63),

(9, 51);

--DELETE FROM Presidentes

SELECT \* FROM Vogais;

INSERT INTO Vogais(numero\_vogal)

VALUES (10),

(11),

(12),

(13),

(14),

(15);

--DELETE FROM Vogais

SELECT \* FROM Descricao;

INSERT INTO Descricao(titulo, descricao)

VALUES ('Moderador', 'Pessoa que dirige uma mesa-redonda, um debate ou uma discussão em grupo.'),

('Deputado', 'Representante do povo eleito para o parlamento.'),

('Tesoureiro', 'Encarregado de efetuar as operações monetárias de um banco, de uma empresa, associação, etc.');

--DELETE FROM Descricao

SELECT \* FROM Cargos;

INSERT INTO Cargos(id\_cargos, titulo)

VALUES (1, 'Moderador'),

(2, 'Deputado'),

(3, 'Tesoureiro');

--DELETE FROM Cargos

SELECT \* FROM Mesa\_eleitoral;

INSERT INTO Mesa\_eleitoral(id\_mesa\_eleitoral, titulo, localizacao)

VALUES (1, 'Mesa eleitoral de Vila Real', 'Vila Real'),

(2, 'Mesa eleitoral do Porto', 'Porto'),

(3, 'Mesa eleitoral de Lisboa', 'Lisboa');

--DELETE FROM Mesa\_eleitoral

SELECT \* FROM Orcamento;

INSERT INTO Orcamento(id\_cargos, orcamento)

VALUES (1, 2000),

(2, 4000),

(3, 3000);

--DELETE FROM Orcamento

SELECT \* FROM Candidatura;

INSERT INTO Candidatura(id\_cargos, numero\_candidato, data\_candidatura)

VALUES (1, 1,'2021-01-16'),

(3, 1,'2021-01-18'),

(2, 2,'2021-01-23'),

(3, 2,'2021-01-24'),

(1, 3,'2021-02-03'),

(2, 4,'2021-02-11'),

(3, 4,'2021-02-12'),

(2, 5,'2021-03-22'),

(3, 6,'2021-03-14');

--DELETE FROM Candidatura

SELECT \* FROM Mandatario; --Alterações nas Colunas da Tabela!!!--

INSERT INTO Mandatario(numero\_eleitor, id\_candidatura, numero\_candidato, data\_candidatura)

VALUES (16, 1, 1,'2021-01-16'),

(17, 3, 2,'2021-01-24'),

(18, 3, 6,'2021-03-14');

--DELETE FROM Mandatario

SELECT \* FROM Local\_votar;

INSERT INTO Local\_votar(numero\_eleitor, local\_votar)

VALUES (19, 'Vila Real'),

(22, 'Vila Real'),

(23, 'Vila Real'),

(20, 'Porto'),

(21, 'Lisboa');

--DELETE FROM Local\_votar

SELECT \* FROM Votar; --Alterações nas Colunas da Tabela!!!--

INSERT INTO Votar(numero\_eleitor, id\_candidatura, numero\_candidato, data\_candidatura, data\_votar)

--Por causa do 'hoje' é necessário alterar o data\_votar para o dia de

--hoje pois não se pode colocar getdate() por causa que se o colocássemos

--todas as pessoas votariam todas ao mesmo tempo e neste caso não nos é --pedido isso.

VALUES (19, 1, 1, '2021-01-16', '2021-05-22 09:30:10'),

(22, 2, 2, '2021-01-23', '2021-05-22 10:22:27'),

(23, 2, 5, '2021-03-22', '2021-05-22 16:46:56'),

(20, 3, 4, '2021-02-12', '2021-05-22 13:24:30'),

(21, 3, 6, '2021-03-14', '2021-05-22 14:56:45');

--DELETE FROM Votar

SELECT \* FROM Numero\_votos;

INSERT INTO Numero\_votos(numero\_candidato, numero\_votos)

VALUES (1, 32),

(2, 67),

(3, 40),

(4, 27),

(5, 15),

(6, 59);

--DELETE FROM Numero\_votos

SELECT \* FROM Assumir; --Alterações nas Colunas da Tabela!!!--

INSERT INTO Assumir(id\_cargos, numero\_candidato, data\_inicio, data\_fim)

VALUES (1, 1,'2021-04-20', '2023-02-15'),

(2, 2,'2021-04-04', '2022-04-04');

INSERT INTO Assumir(id\_cargos, numero\_candidato, data\_inicio)

VALUES (3, 6,'2021-05-10');

--DELETE FROM Assumir

SELECT \* FROM Presidir; --Alterações nas Colunas da Tabela!!!--

INSERT INTO Presidir(id\_mesa\_eleitoral, numero\_presidente, data\_inicio, data\_fim)

VALUES (1, 7, '2021-04-15', '2021-04-18'),

(2, 7, '2021-04-20', '2021-04-23'),

(3, 7, '2021-04-25', '2021-04-28'),

(1, 8, '2021-05-01', '2021-05-03'),

(2, 8, '2021-05-05', '2021-06-08'),

(3, 8, '2021-05-10', '2021-05-13');

INSERT INTO Presidir(id\_mesa\_eleitoral, numero\_presidente, data\_inicio)

VALUES (3, 9, '2021-05-21');

--DELETE FROM Presidir

SELECT \* FROM Hora\_participar;

INSERT INTO Hora\_participar(data\_participar, hora\_participar)

VALUES ('2021-04-15', '09:30:00'),

('2021-04-20', '09:30:00'),

('2021-04-25', '09:30:00'),

('2021-05-01', '09:00:00'),

('2021-05-05', '09:00:00'),

('2021-05-10', '09:00:00'),

('2021-05-21', '10:00:00');

--DELETE FROM Hora\_participar

SELECT \* FROM Participar;

INSERT INTO Participar(numero\_presidente, numero\_vogal\_a, numero\_vogal\_b, id\_mesa\_eleitoral, data\_participar)

VALUES (7, 10, 11, 1, '2021-04-15'),

(7, 10, 11, 2, '2021-04-20'),

(7, 10, 11, 3, '2021-04-25'),

(8, 12, 13, 1, '2021-05-01'),

(8, 12, 13, 2, '2021-05-05'),

(8, 12, 13, 3, '2021-05-10'),

(9, 14, 15, 3, '2021-05-21');

--DELETE FROM Participar

## Resposta ás questões propostas em SQL

---------Resposta ás questões propostas em SQL---------

--2.1. Qual o último cargo assumido?

--[Cargos (título), DataInicio, Candidato(Nome)]

SELECT titulo AS Cargos, data\_inicio AS DataInicio, nome AS Candidato

FROM Cargos, Assumir, Pessoas, Candidatos

WHERE data\_inicio = (SELECT MAX(data\_inicio) FROM Assumir)

AND Cargos.id\_cargos = Assumir.id\_cargos

AND Assumir.numero\_candidato = Candidatos.numero\_candidato

AND Candidatos.numero\_candidato = Pessoas.numero\_eleitor

--2.2. Quantos vogais tem cada mesa eleitoral?

--[Mesa (Titulo), N\_Vogais]

SELECT titulo AS Mesa, (COUNT(numero\_vogal\_a) + COUNT(numero\_vogal\_b)) AS N\_Vogais

FROM Mesa\_eleitoral, Participar, (

SELECT MIN(data\_participar) AS min\_data\_participar, id\_mesa\_eleitoral

FROM Participar

GROUP BY id\_mesa\_eleitoral)SQ1

WHERE Mesa\_eleitoral.id\_mesa\_eleitoral = Participar.id\_mesa\_eleitoral

AND Participar.data\_participar = SQ1.min\_data\_participar

GROUP BY titulo

--2.3. Quais as duas primeiras pessoas a votar hoje em Vila Real?

--[Pessoas(nome)]

--Por causa do 'hoje' é necessário alterar na tabela para o dia de

--hoje pois não se pode colocar getdate() por causa que se o colocássemos --todas as pessoas votariam todas ao mesmo tempo e neste caso não nos é --pedido isso.

SELECT TOP 2 data\_votar, (nome + ' ' + apelido) AS Nome

FROM Pessoas, Votar, Local\_votar

WHERE Pessoas.numero\_eleitor = Votar.numero\_eleitor

AND Votar.numero\_eleitor = Local\_votar.numero\_eleitor

AND Local\_votar.local\_votar = 'Vila Real'

AND CAST (data\_votar AS date) = CAST (GETDATE() AS date)

ORDER BY data\_votar ASC

--2.4. Quais as pessoas que presidiram mais do que 2 mesas eleitorais nos --últimos 90 dias? Ordene-as alfabeticamente.

--[Nome e Apelido, Titulo, Data]

SELECT nome, apelido, titulo, data\_inicio--, data\_fim

FROM Pessoas, Presidentes, Presidir, Mesa\_eleitoral, (

--Conta o numero de mesas presididas por cada presidente associando-as ao --numero\_eleitor

SELECT numero\_eleitor, COUNT(id\_mesa\_eleitoral) AS count\_mesa\_eleitoral

FROM Pessoas, Presidentes, Presidir

WHERE Pessoas.numero\_eleitor = Presidentes.numero\_presidente

AND Presidentes.numero\_presidente = Presidir.numero\_presidente

GROUP BY nome, apelido, numero\_eleitor)SQ1

WHERE SQ1.count\_mesa\_eleitoral > 2

AND DATEDIFF(DD, data\_inicio, GETDATE()) < 90

AND SQ1.numero\_eleitor = Pessoas.numero\_eleitor

AND Pessoas.numero\_eleitor = Presidentes.numero\_presidente

AND Presidentes.numero\_presidente = Presidir.numero\_presidente

AND Presidir.id\_mesa\_eleitoral = Mesa\_eleitoral.id\_mesa\_eleitoral

GROUP BY nome, apelido, titulo, data\_inicio--, data\_fim

ORDER BY nome + apelido, data\_inicio ASC

--2.5. Qual é o cargo com mais candidaturas?

--[TítuloDoCargo, N\_Candidaturas]

SELECT TOP 1 COUNT(\*) AS N\_Candidaturas, titulo AS TítuloDoCargo

FROM Cargos, Candidatura

WHERE Cargos.id\_cargos = Candidatura.id\_cargos

GROUP BY titulo

ORDER BY N\_Candidaturas DESC

--2.6. Qual a nacionalidade e a que cargo se candidata o candidato mais novo?

--[Candidato (nome e apelido), Idade, Nacionalidade, CargoTitulo]

SELECT nome, apelido, (DATEDIFF(YY, data\_nascimento, GETDATE())) AS Idade, nacionalidade, titulo AS CargoTitulo

FROM Pessoas, Candidatos, Candidatura, Cargos

WHERE data\_nascimento = (SELECT MAX(data\_nascimento) FROM Pessoas)

AND Pessoas.numero\_eleitor = Candidatos.numero\_candidato

AND Candidatos.numero\_candidato = Candidatura.numero\_candidato

AND Candidatura.id\_cargos = Cargos.id\_cargos

GROUP BY nome, apelido, data\_nascimento, nacionalidade, titulo

--2.7. Qual o total dos orçamentos de todas as candidaturas de cada --candidato?

--Ordene-os por ordem crescente. [Candidato (Nome), TotalGasto]

SELECT nome, SUM(orcamento) AS TotalGasto

FROM Pessoas, Orcamento, Candidatos, Candidatura

WHERE Pessoas.numero\_eleitor = Candidatos.numero\_candidato

AND Candidatos.numero\_candidato = Candidatura.numero\_candidato

AND Candidatura.id\_cargos = Orcamento.id\_cargos

GROUP BY nome

ORDER BY TotalGasto DESC

# Conclusão

Com todos os conceitos trabalhados e tratados ao longo do relatório podemos então chegar à conclusão desta segunda etapa do trabalho.

Com a segunda fase pudemos entender como funciona de modo mais específico a linguagem SQL, linguagem usada para a construção de base de dados. Também entendemos a importância que é ter uma base de dados em um estado de integridade, ou seja, apenas com dados válidos.

# Bibliografia

* Paulo Martins, (2021). Conceção e Desenvolvimento de Bases de Dados v2
* Paulo Martins, (2021). Linguagem SQL
* Paulo Martins, (2021). Resolução Ficha Prática 5.1 e 6.1