

Databases

Introdução ao SQL

Bachelor in Informatics Engineering
Department of Informatics Engineering
University of Coimbra
2024/2025





Outline

- Introdução a SQL
- Consultas simples

Register your presence at UCStudent!

Tabelas

- As tabelas de uma base de dados são essencialmente semelhantes às que usamos em *papel* ou numa *folha de cálculo*
 - Têm algumas características específicas que veremos mais tarde
- 4 conceitos importantes:
 - Tabela
 - Linha ou Registo
 - Coluna
 - Campo

← Linha ou Registo

→ Campo

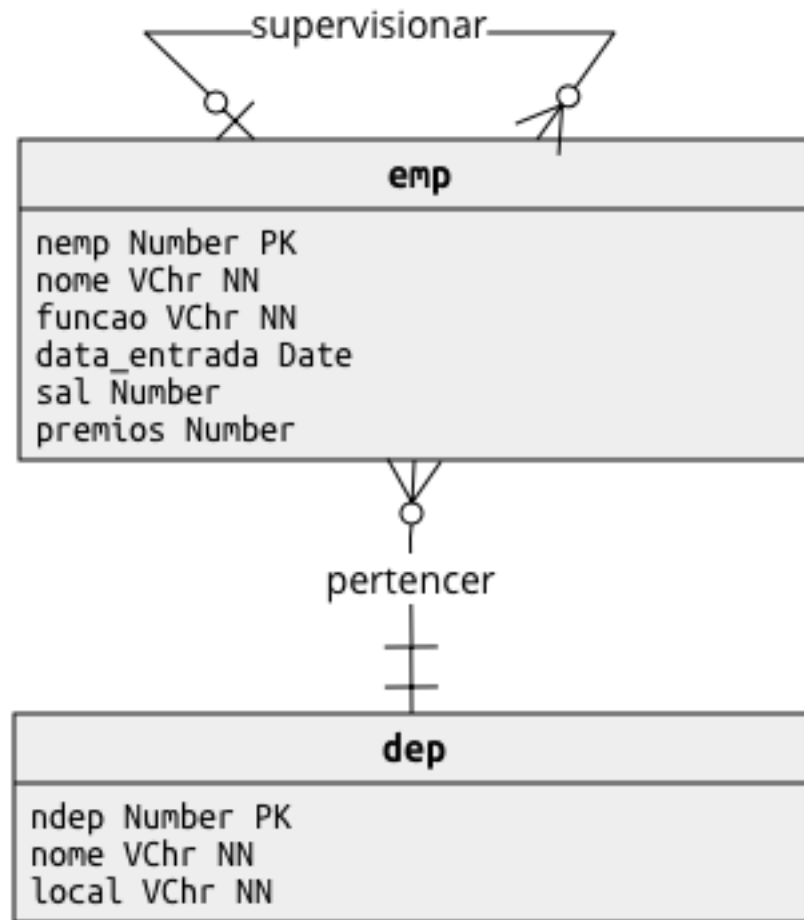
↑
Coluna



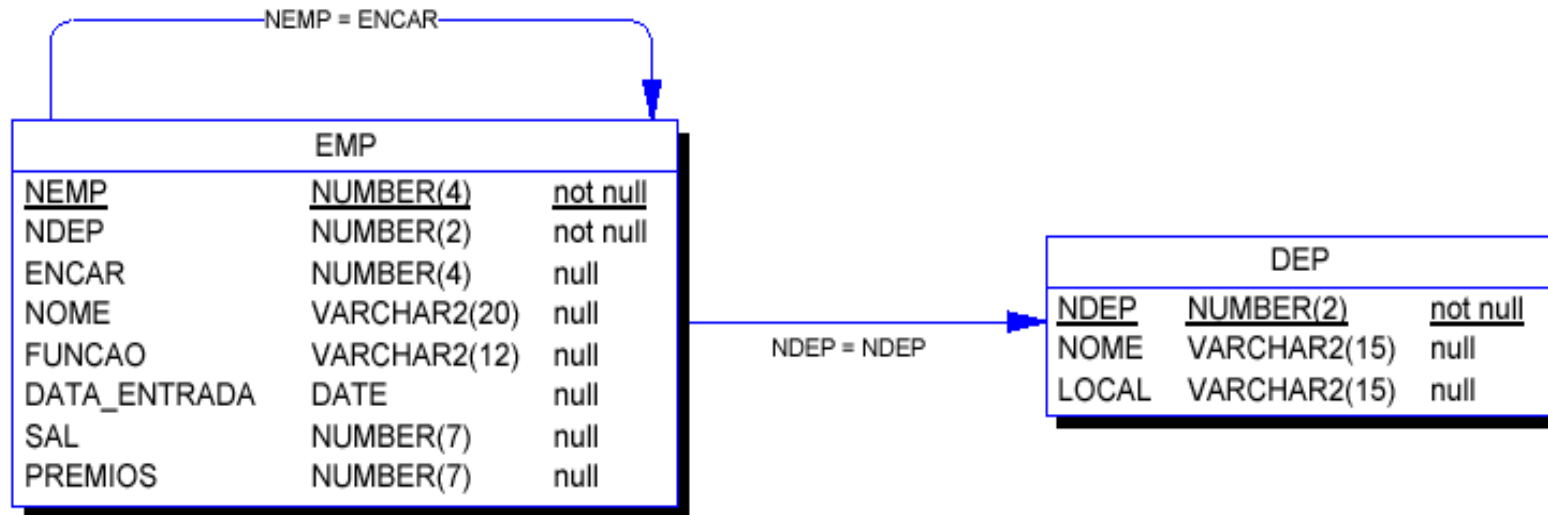
O problema

- Uma empresa é composta por vários departamentos
- Cada empregado trabalha num departamento e pode supervisionar vários outros empregados
 - Mesmo que trabalhem noutros departamentos
- Um empregado é supervisionado no máximo por um outro empregado
- O Presidente desta empresa não tem nenhum supervisor

Modelo conceptual



Modelo físico



Manipulação e utilização de dados relacionais

- **Definição de dados** (criar e alterar tabelas, vistas, etc)
- **Integridade** (definir regras de integridade dos dados)
- **Interrogação** (pesquisar dados em tabelas)
- **Manipulação de dados** (inserir, alterar, apagar dados)
- **Autorizações e segurança** (definir privilégios, perfis, etc)
- **Controlo de transações** (iniciar e terminar transações)

SQL
(Structured
Query
Language)

SQL é um standard ANSI e ISO desde 1986



SQL

- Linguagem não procedimental adaptada à língua inglesa
- Disponibiliza comandos para extração, inserção e manipulação de dados
- Disponibiliza também comandos para alteração da base de dados
- O utilizador não sabe ou não precisa de saber o formato ou a localização dos dados
- O utilizador também não precisa de saber como os dados são obtidos (algoritmo usado, etc.)



O comando SELECT

- O comando SELECT é o mais importante e mais complexo de todos os comandos de SQL
- Serve para seleccionar dados
- Para tal, pode aplicar vários tipos de relações às tabelas existentes na base de dados
 - Restrição, projecção, produto, junção, união, intersecção e diferença



Estrutura do Comando SELECT

- Estrutura (simplificada):

```
SELECT <lista de colunas>  
  FROM <lista de tabelas>  
 WHERE <restrições a aplicar>  
 ORDER BY <colunas a usar para o ordenamento>
```

Projeção

- Quando não queremos ver todas as colunas
- Operação que permite selecionar uma ou mais colunas de uma tabela e criar uma nova tabela (resultado)

```
SELECT nome, funcao  
FROM emp;
```

Mostra apenas os atributos
nome e **funcao** da tabela **emp**

Mostra todos os registos
existentes na tabela pois
não há qualquer restrição
(i.e., não especificámos
uma cláusula *where*).

NOME	FUNCAO
-----	-----
Jorge Sampaio	Presidente
Augusto Reis	Encarregado
Duarte Guedes	Encarregado
Silvia Teles	Encarregado
Maria Dias	Analista
Catarina Silva	Analista
Joana Mendes	Vendedor
Nelson Neves	Vendedor
Ana Rodrigues	Vendedor
Manuel Madeira	Vendedor
Tome Ribeiro	Continuo
Rita Pereira	Continuo
Olga Costa	Continuo
Antonio Silva	Continuo

14 rows selected.

Projeção e valores únicos

```
SELECT funcao  
FROM emp;
```

FUNCAO

Presidente
Encarregado
Encarregado
Encarregado
Analista
Analista
Vendedor
Vendedor
Vendedor
Vendedor
Continuo
Continuo
Continuo
Continuo

```
SELECT DISTINCT funcao  
FROM emp;
```

FUNCAO

Presidente
Encarregado
Continuo
Vendedor
Analista

Restrição

- Operação que permite seleccionar registos de uma tabela que satisfazem uma dada condição.

```
SELECT *  
FROM emp  
WHERE ndep = 10;
```

Mostra todos os atributos (i.e., todas as colunas) da tabela **emp**

Mas apenas para os registos que verificam a condição de **ndep = 10**

Resultado:

NEMP	NOME	FUNCAO	ENCAR	DATA_ENT	SAL	PREMIOS	NDEP
1839	Jorge Sampaio	Presidente		84.02.11	890000		10
1782	Silvia Teles	Encarregado	1839	86.11.03	279450		10
1934	Olga Costa	Continuo	1782	86.06.22	68300		10

Restrição: Operadores (1)

- É possível também usar outros operadores para além do ‘=’

Operador	Descrição
=	Igual a
<>	Diferente de
>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
BETWEEN	Dentro de um intervalo
LIKE	Igual a um padrão
IN	Dentro de uma lista de valores

```
SELECT *  
FROM emp  
WHERE ndep <> 10;
```

Restrição: Operadores (2)

- Para escrevermos cláusulas *where* válidas é necessário formatar **dados** corretamente com o uso de **apóstrofos**
- Dados numéricos não necessitam de apóstrofos, ao contrário de outros tipos de dados (e.g., character, varchar, date, timestamp, etc.)
- Em alguns SGBDs podemos também usar os operadores '<' e '>' para comparar datas

```
SELECT *  
FROM emp  
WHERE ndep = 10;
```

```
SELECT *  
FROM emp  
WHERE nome = 'Manuel Madeira';
```

Restrição: Operadores (3)

- Utilização do operador **IN**

```
SELECT *  
FROM emp  
WHERE ndep IN (10, 30)
```

- O operador **LIKE** é usado para pesquisar um padrão
- Os padrões definem-se com alguns caracteres especiais
- Para especificar zero ou mais letras desconhecidas no padrão usa-se o caracter %
- O que fará a seguinte query?

```
SELECT *  
FROM emp  
WHERE nome LIKE '%Silva'
```


Restrição: Operadores lógicos

- Podem ser usadas condições múltiplas usando os operadores **AND, OR e NOT**

```
SELECT *  
FROM emp  
WHERE ndep = 10 OR ndep = 30;
```

NEMP	NOME	FUNCAO	ENCAR	DATA_ENT	SAL	PREMIOS	NDEP
1839	Jorge Sampaio	Presidente		11/2/84	890000		10
1698	Duarte Guedes	Encarregado	1839	25/11/91	380850		30
1782	Silvia Teles	Encarregado	1839	3/11/86	279450		10
1499	Joana Mendes	Vendedor	1698	4/10/84	145600	56300	30
1521	Nelson Neves	Vendedor	1698	27/2/83	212250	98500	30
1654	Ana Rodrigues	Vendedor	1698	17/12/90	221250	81400	30
1844	Manuel Madeira	Vendedor	1698	21/4/85	157800	0	30
1900	Tome Ribeiro	Continuo	1698	5/3/94	56950		30
1934	Olga Costa	Continuo	1782	22/6/86	68300		10

- Os operadores podem ser combinados em condições mais complexas (use parêntesis para agrupar as sub-condições)

Restrição: Verificação de valores nulos

- Exemplo: queremos listar todos os empregados que recebem prémios. Solução imediata:

```
SELECT *  
FROM emp  
WHERE premios >= 0
```

- Mas... como listar todos os empregados que **não** recebem prémios?
- Por outras palavras, queremos obter todas as linhas onde a coluna **premios** esteja vazia (**nula**), isto é, tenha o valor especial **NULL**

```
SELECT *  
FROM emp  
WHERE premios IS NULL
```

- Valores **NULL** são tratados de forma diferente dos restantes
- NULL não é equivalente a zero** (não são comparáveis)

Restrição: Verificação de valores não nulos

- Solução alternativa para o exemplo (listar todos os empregados que recebem prémios):

```
SELECT *  
FROM emp  
WHERE premios IS NOT NULL
```

Restrição e projeção

- Podemos combinar as operações de restrição e de projeção.

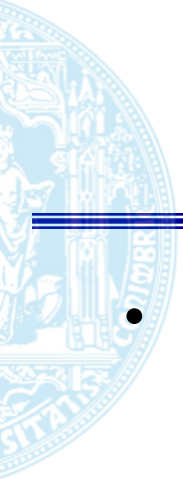
```
SELECT nome, funcao, sal
FROM emp
WHERE ndep = 10 AND sal > 200;
```

Projeção: mostra apenas os atributos **nome**, **funcao** e **sal** da tabela **emp**

Restrição: Mostra apenas os registros que satisfazem a condição de ser do departamento 10 e ter salário maior do que 200.

Resultado:

NOME	FUNCAO	SAL
-----	-----	-----
Jorge Sampaio	Presidente	890000
Silvia Teles	Encarregado	279450
Olga Costa	Continuo	68300



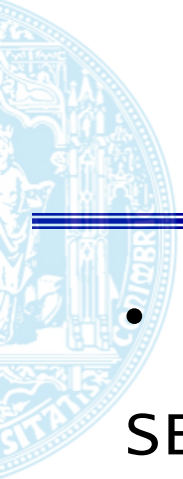
Cláusulas SELECT e FROM

- A cláusula SELECT especifica uma lista de nomes de colunas das tabelas separadas por vírgulas
- Permite ainda fazer operações aritméticas, de *strings*, e de datas sobre os valores selecionados
- Permite também renomear as colunas de dados através de pseudónimos

```
SELECT nome, nemp, sal  
FROM emp;
```

```
SELECT nome, sal * 12  
FROM emp;
```

NOME	sal * 12
Jorge Sampaio	10680000
Augusto Reis	5411700
...	...



Cláusulas SELECT e FROM

- Uso de pseudónimos (*alias*):

```
SELECT nome, sal * 12 "Remuneracao anual"  
FROM emp;
```

- Concatenação de colunas:

```
SELECT nome || funcao "Nome e funcao"  
FROM emp;
```

- Tratamento de valores nulos:

```
SELECT nome, COALESCE(premios, 0)  
FROM emp;
```

NOME	Remuneracao anual
Jorge Sampaio	10680000
Augusto Reis	5411700
...	...

Cláusula WHERE

- Permite restringir, através de uma **condição**, as linhas que fazem parte do resultado

```
SELECT *  
FROM emp  
WHERE ndep = 10;
```

NOME	FUNCAO	SAL	NDEP
Jorge Sampaio	Presidente	890000	10
Silvia Teles	Encarregado	279450	10
Olga Costa	Continuo	68300	10

- Podem usar-se condições sobre uma ou mais tabelas
- Para além das restrições, é comum relacionar colunas de várias tabelas
 - Voltaremos a estas funcionalidades em futuras aulas

Cláusula ORDER BY

- Serve para especificar uma determinada ordem
 - Os registos não estão ordenados na base de dados

```
SELECT nome, funcao, sal  
FROM emp  
ORDER BY ndep, sal DESC, nome;
```

NOME	FUNCAO	SAL
Jorge Sampaio	Presidente	890000
Silvia Teles	Encarregado	279450
Olga Costa	Continuo	68300
Maria Dias	Analista	565000
Augusto Reis	Encarregado	450975
Catarina Silva	Analista	435000
Antonio Silva	Continuo	70800
Rita Pereira	Continuo	65100
Duarte Guedes	Encarregado	380850
Ana Rodrigues	Vendedor	221250
Nelson Neves	Vendedor	212250
Manuel Madeira	Vendedor	157800
Joana Mendes	Vendedor	145600
Tome Ribeiro	Continuo	56950

- A cláusula ORDER BY aparece no final
- Os resultados são ordenados de forma ASCendente ou DESCendente
- Podem ser usadas colunas que não foram especificadas no SELECT

Tabela emp

- Dados dos Empregados
- Colunas
 - nome
 - funcao
 - sal
 - ndep
- *Nota: a tabela usada nos exercícios tem mais colunas*
- Vai servir de base para os próximos exercícios

NOME	FUNCAO	SAL	NDEP
Jorge Sampaio	Presidente	890000	10
Augusto Reis	Encarregado	450975	20
Duarte Guedes	Encarregado	380850	30
Silvia Teles	Encarregado	279450	10
Maria Dias	Analista	565000	20
Catarina Silva	Analista	435000	20
Joana Mendes	Vendedor	145600	30
Nelson Neves	Vendedor	212250	30
Ana Rodrigues	Vendedor	221250	30
Manuel Mad...	Vendedor	157800	30
Tome Ribeiro	Continuo	56950	30
Rita Pereira	Continuo	65100	20
Olga Costa	Continuo	68300	10
Antonio Silva	Continuo	70800	20



O Comando SELECT (revisitado)

- Para consultar a informação presente nas diversas tabelas da base de dados podemos usar o comando SELECT:

SELECT	que atributos (i.e., colunas) queremos ver
FROM	em que tabela(s) se encontra a informação
WHERE	qual o(s) critério(s) para a escolha dos dados (e.g., queremos ver apenas 5 registos, ou apenas registos com características especiais)

Exemplo simples do comando SELECT

- Ver todos os dados de uma tabela

```
SQL> select * from emp;
```

NEMP	NOME	FUNCAO	ENCAR	DATA_ENT	SAL	PREMIOS	NDEP
1839	Jorge Sampaio	Presidente		84.02.11	890000		10
1566	Augusto Reis	Encarregado	1839	85.02.13	450975		20
1698	Duarte Guedes	Encarregado	1839	91.11.25	380850		30
1782	Silvia Teles	Encarregado	1839	86.11.03	279450		10
1788	Maria Dias	Analista	1566	82.11.07	565000		20
1902	Catarina Silva	Analista	1566	93.04.13	435000		20
1499	Joana Mendes	Vendedor	1698	84.10.04	145600	56300	30
1521	Nelson Neves	Vendedor	1698	83.02.27	212250	98500	30
1654	Ana Rodrigues	Vendedor	1698	90.12.17	221250	81400	30
1844	Manuel Madeira	Vendedor	1698	85.04.21	157800	0	30
1900	Tome Ribeiro	Continuo	1698	94.03.05	56950		30
1876	Rita Pereira	Continuo	1788	96.02.07	65100		20
1934	Olga Costa	Continuo	1782	86.06.22	68300		10
1369	Antonio Silva	Continuo	1902	96.12.22	70800		20

14 rows selected.



Q&A



Databases

Introdução ao SQL

Bachelor in Informatics Engineering
Department of Informatics Engineering
University of Coimbra
2024/2025

