

Título	Ficha SQL 1 - Introdução à linguagem SQL
Contexto	Aula
Capítulo	Capítulo 2 – Manipulação de Dados
Conteúdos abordados	<ul style="list-style-type: none"> ▪ Arquitetura <i>cliente-servidor</i> no acesso a bases de dados ▪ Ligação a um servidor de base de dados remoto ▪ Aplicações <i>cliente</i> (GUI) de uma base de dados ▪ Operações CRUD em bases de dados relacionais, usando a linguagem SQL ▪ Objetos de bases de dados: TABLE

Por definição, "base de dados é todo o conjunto de dados relacionados entre si" [1], o que inclui desde uma simples folha de papel com apontamentos de uma aula até algo mais sofisticado como, por exemplo, gigantescos repositórios com milhares de milhões de registos médicos associados ao sistema de saúde de um país.

O tipo de base de dados com mais impacto no funcionamento diário das organizações é a **base de dados relacional** [2]. Este tipo de base de dados baseia-se em conceitos simples:

- os dados são organizados em tabelas bidimensionais, contendo linhas e colunas (exemplo na Figura 1).
- cada tabela armazena dados sobre determinado **conjunto** de objetos, de indivíduos, de eventos, etc., todos com características comuns entre si.
- **linha** é um grupo de valores que caracterizam determinado elemento do **conjunto**.
- **coluna** é cada característica partilhada pelas linhas da tabela.
- as linhas de uma tabela podem relacionar-se com linhas de outra(s) tabela(s), justificando o nome "bases de dados **relacionais**".

TABELA cliente

id	nome	cidade	num_telefone
1	António Freitas	Leiria	123456
2	Manuel da Silva	Lisboa	111222
...
100	Maria de Sousa	Porto	121212

coluna / atributo

linha / registo

Figura 1 – Exemplo da tabela CLIENTE numa base de dados relacional.

Os dados contidos nas bases de dados relacionais podem ser consultados/alterados/removidos utilizando uma linguagem específica criada para esse tipo de bases de dados: a linguagem SQL (*Structured Query Language*, lê-se *Siquid*) [3][4], que é utilizada por inúmeros fabricantes de servidores de bases de dados relacionais (Oracle, Microsoft, IBM, etc.). Usando comandos da linguagem SQL, as aplicações cliente (ou *GUI-Guided User Interfaces*) enviam pedidos de informação a esses servidores de bases de dados que, após os processarem, devolvem as respostas a essas mesmas aplicações cliente. A Figura 2 apresenta simplificadamente esta arquitetura, denominada *cliente-servidor*.

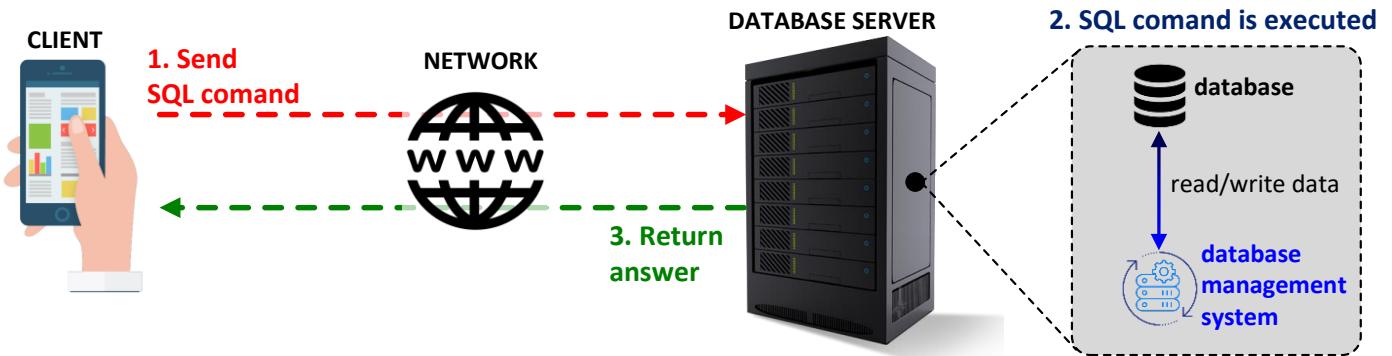


Figura 2 –Arquitetura *cliente-servidor*, a base de comunicação entre aplicações cliente e bases de dados num Sistema de Informação (SI).

Nesta ficha, começará a tomar contacto com as bases de dados relacionais, realizando as operações mais básicas associadas a esses ambientes:

- Ligação a um servidor de bases de dados.
- Criação de tabelas para armazenamento de dados.
- Inserção de dados em tabelas (inserir dados = ***populate***).
- Atualização e eliminação de dados em tabelas.
- Escrita de *queries* (***query*** = pesquisa = consulta).



**Neste momento, é importante analisar o documento anexo
Arquitetura do Sistema de Informação.**

No decurso das aulas práticas, cada estudante irá assumir o papel de :

- **DATABASE DEVELOPER**, construindo objetos da base de dados que permitam responder aos pedidos feitos pelos *end-users*.
- **GUI DEVELOPER**, alterando a GUI (criada em *java*) para que envie comandos/pedidos para a base de dados quando os *end-users* interagirem com a GUI (por exemplo, ao clicar em botões).
- **END-USER**, testando com a GUI se o SI foi implementado como esperado.



Esta instalação é desnecessária nos computadores dos laboratórios.

1. Verificar se o *software Java Developer's Kit* (JDK) está instalado no computador pessoal. Para isso, na consola em linha de comando, digitar "javac -version", **seja em Windows, LINUX ou MAC OS** (Figura 3).

Nota: a versão 21.0.4 do JDK é a mínima requerida.

```
C:\>javac -version
javac 21.0.4
C:\>-
```

Figura 3 – Como descobrir a versão do JDK instalada (neste caso, na janela de consola do Windows 10).

Se necessário, o *download* do JDK pode ser realizado em

<https://www.oracle.com/java/technologies/downloads/>



2. Descarregar e instalar o *software Oracle SQL Developer* adequado à versão de sistema operativo em

<https://www.oracle.com/tools/downloads/sqldev-downloads.html>



Preparação do cenário: configurar software de desenvolvimento instalado

- Após a instalação, aceder ao software SQL Developer para criar uma ligação (*connection*, Figura 4) para uma conta no servidor de bases de dados. Este servidor encontra-se alojado na *cloud* do Departamento de Engenharia Informática (DEI).

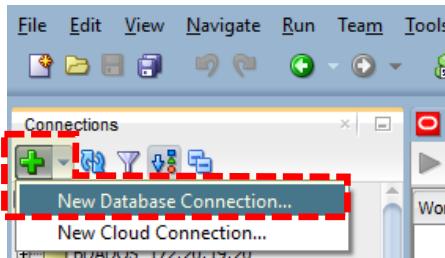


Figura 4 – Configuração de uma nova ligação (*connection*) para uma conta de base de dados.

Em alternativa, aceder a menu File ->New ->Categories/General/Connections ->Database Connection

Configurar a nova *connection* com os dados de autenticação (Figura 5):

- Name*: < qualquer um à escolha >
- Username* e *password* da conta Oracle: **solicitar ao docente da aula prática**
- Dados da base de dados: { *Hostname*=**172.22.21.58**, *Port*=**1521**, *Service name*=**bdados** }

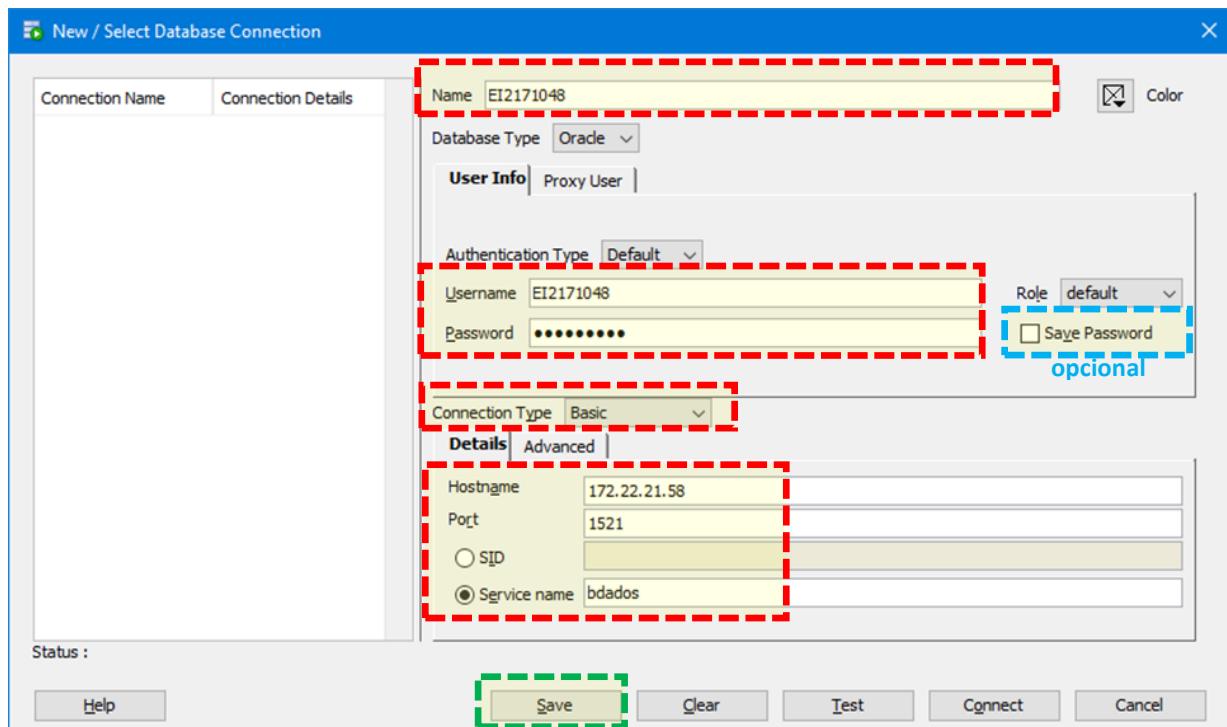


Figura 5 – Configuração de nova ligação (*connection*) para a conta de base de dados exemplo EI2171048.

Acessos fora da rede do IPLeiria

Para se aceder ao servidor Oracle fora da rede do IPLeiria (em casa, na praia, na montanha, no café da rua, etc.), é necessário estar ligado à rede interna da escola por uma VPN (*Virtual Private Network*).

Alternativas:

- Ligação por *browser*, em <https://vpn.ipleiria.pt>
- Ligação com o software *Windows Capsule* (consultar manual disponível no Moodle desta UC, tópico **Software**)
- Ligação com o software *Check Point EndPoint Security VPN* (consultar manual disponível no Moodle desta UC, tópico **Software**)
- Ligação com o software *EndPoint Security* (para sistema operativo MAC OS X, consultar manual disponível no Moodle desta UC, tópico **Software**)

Mais informações em:

[Intranet - Catálogo de Serviços - Direção de Serviços Digitais \(DSDi\) - Redes, Conetividade e Internet - \(ipleiria.pt\)](#)



Estando ligado à rede interna do IPLeiria (por wi-fi ou cabo de rede), a VPN não é necessária.

2. Quando ligado com sucesso ao servidor de bases de dados Oracle, o separador **Connections** é atualizado com a nova ligação. Nesse separador, surgirá a árvore de objetos da conta (Figura 6).

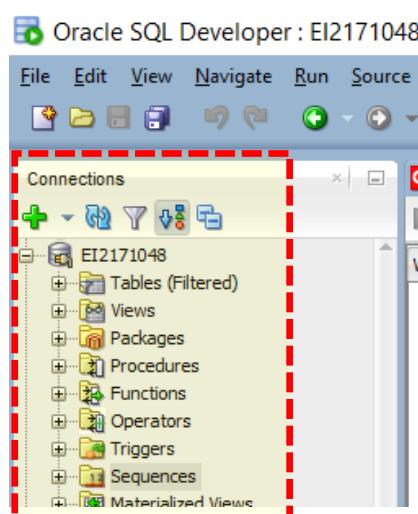


Figura 6 – Separador *Connections*, com a árvore de objetos pertencentes à conta do estudante EI2171048.

3. Para realizar os exercícios das fichas é necessária uma área de trabalho (*worksheet*, Figura 7).

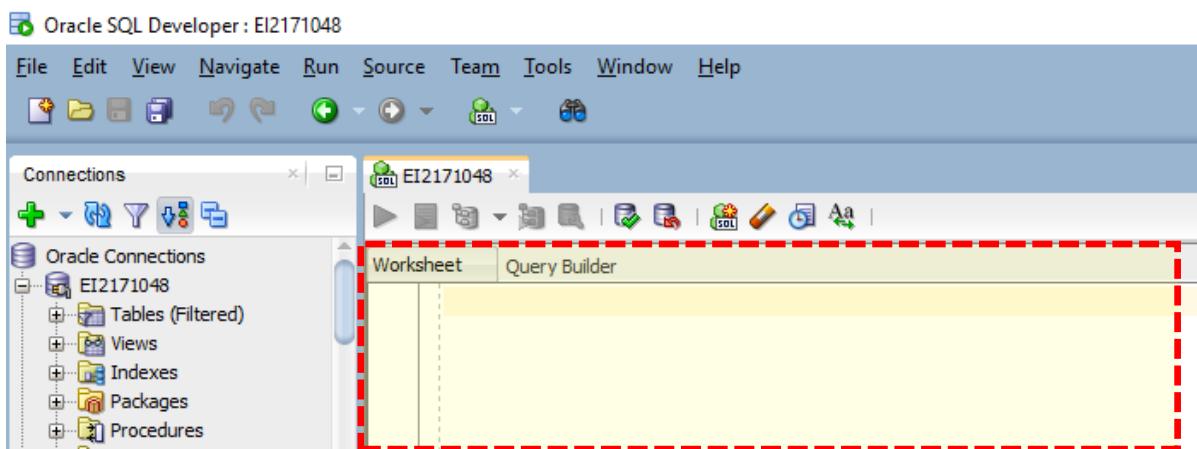


Figura 7 – Worksheet de trabalho, associada à conta do estudante.

Por cada ligação a uma base de dados, o SQL Developer *geralmente* abre uma *worksheet*. Caso tal não aconteça, pode abrir-se manualmente.

Se necessário, pode abrir-se manualmente uma *worksheet*:

- menu *Tools-> SQL Worksheet* (Figura 8).

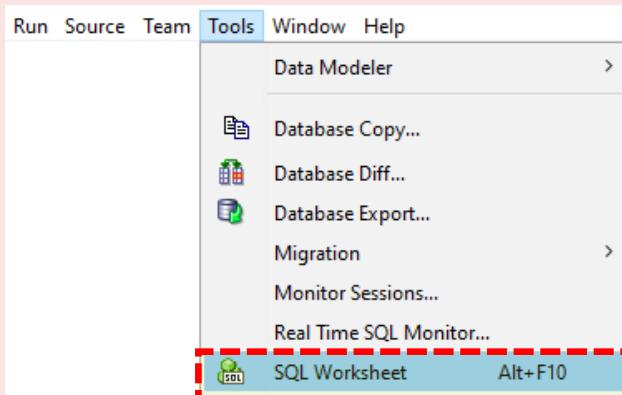


Figura 8 – Como forçar a abertura de uma nova *worksheet*.

4. Operação crítica: alteração da palavra-passe.

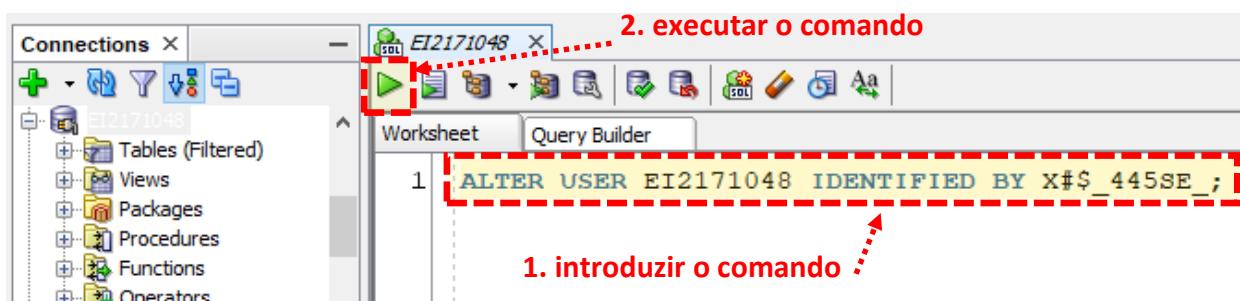


Figura 9 – Exemplo da alteração da conta exemplo EI2171048, neste caso alterando a palavra-passe para X#\$_445SE_ .

5. É importante configurar algumas opções da ferramenta cliente (acessíveis no menu **Tools-> Preferences**), nomeadamente:

- *Encoding* usado para ler/gravar ficheiros locais (Figura 10).
- *Formato por omissão* apresentado nas datas (Figura 11).
- Forma como os dados em falta (*NULL*) serão apresentados (Figura 12).

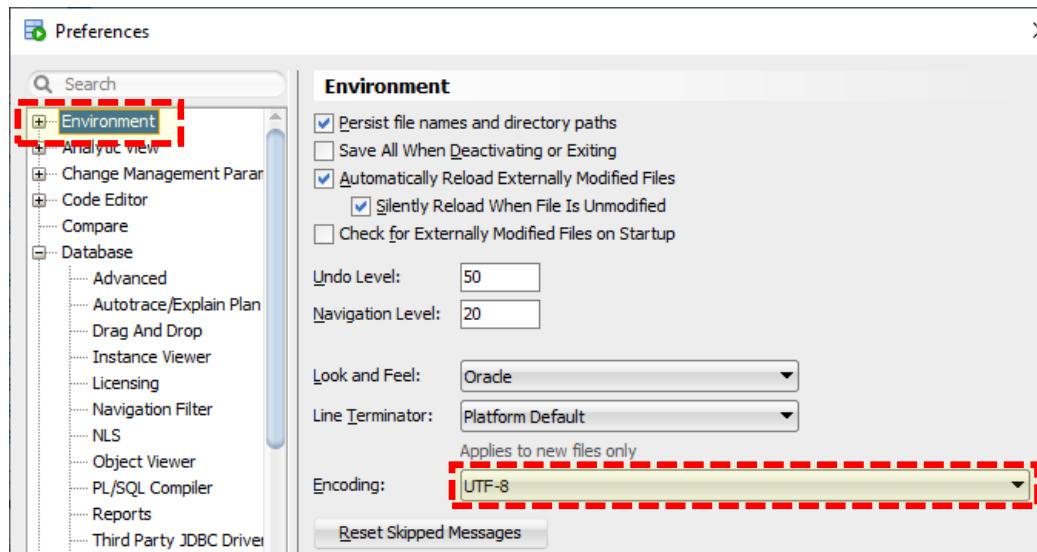


Figura 10 – *Encoding* para ler/gravar ficheiros locais.

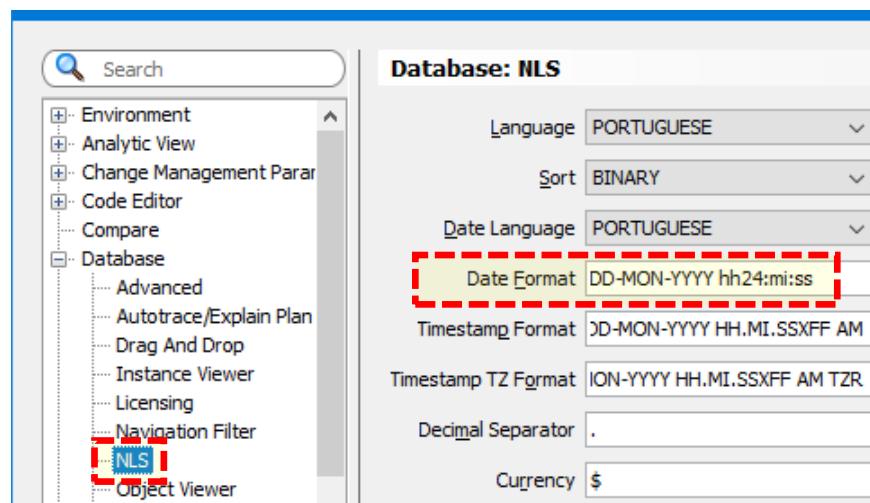


Figura 11 – *Formato por omissão* apresentado nas datas.

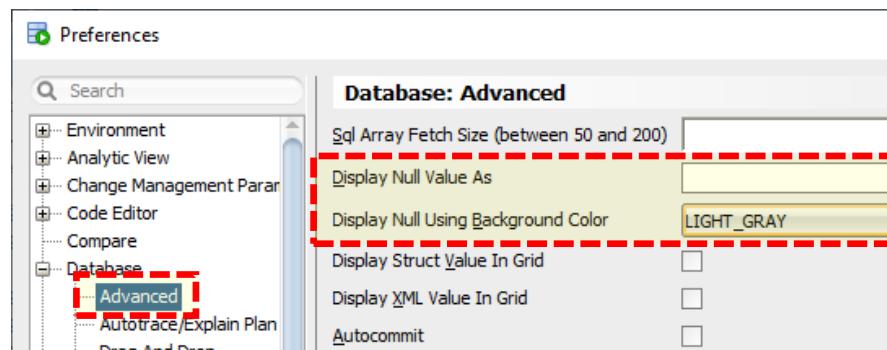
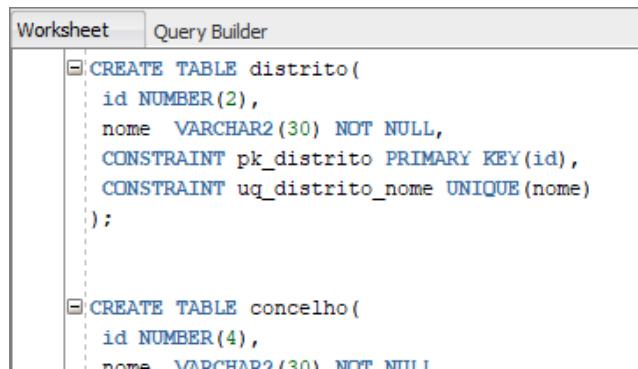


Figura 12 – Forma como os dados em falta (*NULL*) serão apresentados.

Preparação do cenário: carregar dados

1. Usando a funcionalidade *drag-drop* do sistema operativo, arrastar para a *worksheet* o ficheiro **fichas_SQL_tabelas.sql** anexo a esta ficha.
2. Analisar e executar os comandos contidos no ficheiro (Figura 13).



```
CREATE TABLE distrito(
    id NUMBER(2),
    nome VARCHAR2(30) NOT NULL,
    CONSTRAINT pk_distrito PRIMARY KEY(id),
    CONSTRAINT uq_distrito_nome UNIQUE(nome)
);

CREATE TABLE concelho(
    id NUMBER(4),
    nome VARCHAR2(30) NOT NULL
```

Figura 13 – Vista parcial do ficheiro **fichas_SQL_tabelas.sql**.



Caso a funcionalidade *drag-drop* não funcione para abertura ficheiros, garantir que:

- O SQL Developer não foi aberto em modo de administração: este modo impede o uso da funcionalidade *drag-drop*.
- Há pelo menos uma *worksheet* aberta.

Se ainda assim a funcionalidade *drag-drop* não funcionar, os ficheiros podem ser abertos usando os seguintes passos:

- Ativar o separador **Files** (menu **View->Files**).
- Procurar o ficheiro na árvore de pastas e abri-lo (Figura 14).

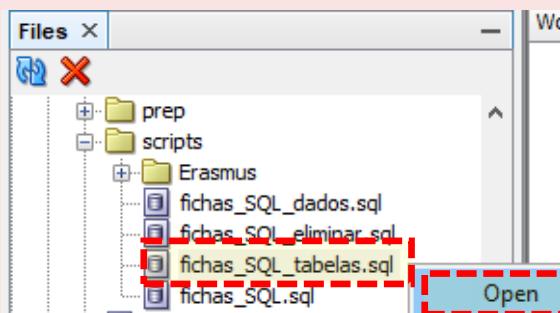
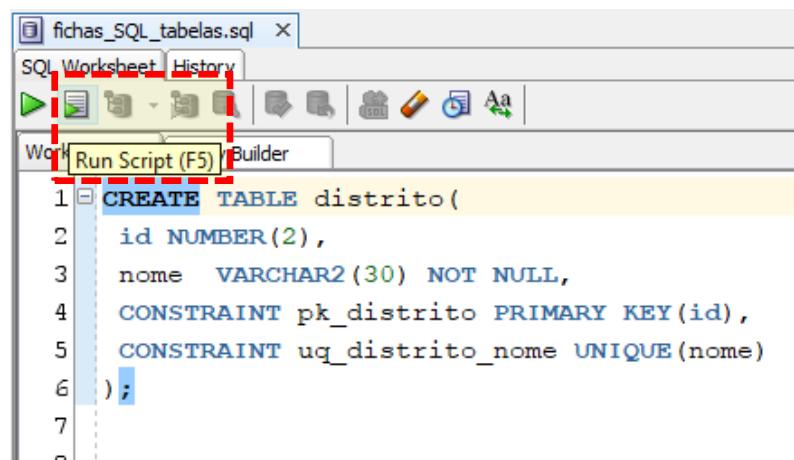


Figura 14 – Abrir um ficheiro a partir do separador *Files*.

3. Executar os comandos contidos no ficheiro (Figura 15).



```
1 CREATE TABLE distrito(
2     id NUMBER(2),
3     nome VARCHAR2(30) NOT NULL,
4     CONSTRAINT pk_distrito PRIMARY KEY(id),
5     CONSTRAINT uq_distrito_nome UNIQUE(nome)
6 );
7
```

Figura 15 – Executar bloco de comandos SQL.

4. Confirmar que as tabelas foram criadas na conta: fazer *refresh* do lado da aplicação cliente *SQL Developer* (Figura 16 e Figura 17).

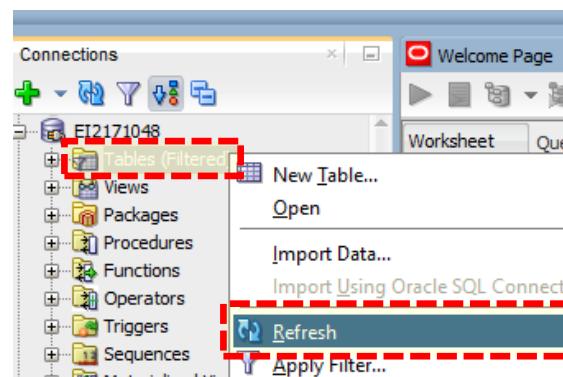


Figura 16 – Refresh na árvore de objetos após criação de novos objetos: usar o botão direito do rato sobre o nó *Tables (Filtered)* para aceder à opção *Refresh*.

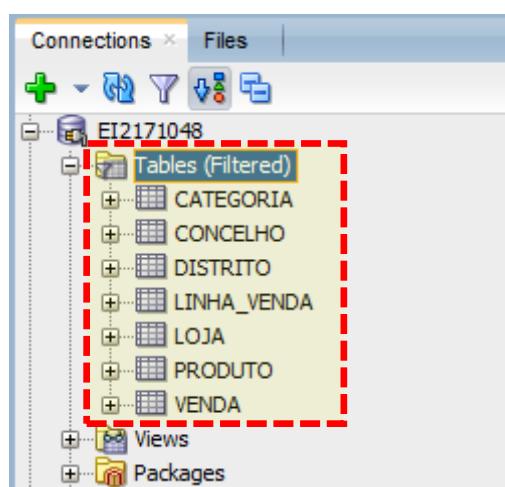


Figura 17 – Confirmação de que os objetos foram criados, após o *refresh*.

- Repetir o processo das perguntas 1 a 3, desta vez para o ficheiro **fichas_SQL_dados.sql**.
- Após a execução dos comandos, analisar os dados armazenados nas tabelas (Figura 18).

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar displays a connection named 'EI2171048' with several tables listed under 'Tables (Filtered)': 'CATEGORIA', 'CONCELHO', 'DISTRITO', and 'LINHA_VENDA'. The 'CONCELHO' table is selected and highlighted with a red dashed box. On the right, the main workspace shows the 'Data' tab for the 'CONCELHO' table, also with a red dashed box around it. The table has three columns: 'ID', 'NOME', and 'DISTRITO_ID'. The data is as follows:

ID	NOME	DISTRITO_ID
1	1312 PORTO	13
2	1004 BATALHA	10
3	1009 LEIRIA	10
4	1010 MARINHA GRANDE	10
5	0603 Coimbra	06
6	0604 CONDEIXA-A-NOVA	06

Figura 18 – Consulta dos dados nas tabelas.



É importante recolher notas dos exercícios realizados na aula, guardando as resoluções e apontamentos em ficheiro (botão Save ou, em alternativa, menu File->Save), tal como apresentado na Figura 19. Assim, o estudo autónomo será muito facilitado.

The screenshot shows the Oracle SQL Developer interface with the 'Worksheet' tab active. The 'Connections' sidebar shows a connection named 'EI2171048' with various tables listed. The worksheet contains handwritten notes in red and blue ink over a SQL script. Red highlights are present on the 'Save (Ctrl-S)' button in the toolbar, the 'Save' button in the connections pane, and several lines of the code. The code itself is a mix of SQL and personal notes:

```

1 -- ficha SQL 1
2 -- Exercício 1
3 -----
4
5 SELECT nome, data
6 FROM ....;
7
8 -- SELECT: cláusula que serve para definir as colunas que desejo obter
9 -- FROM: não apanhei... nunca mais chegarei atrasado à aula prática de BD
10
11
12 -- Exercício 2
13 -----
14 SELECT ...
15 FROM ...
16 WHERE ...;
17
18 -- já suei um pouco...
19

```

Figura 19 – Gravação, em ficheiro de texto, dos comandos realizados durante uma aula.

Preparação do cenário: configurar aplicação cliente

1. Descomprimir o ficheiro *supergest.zip*, para uma pasta do sistema operativo. A pasta conterá o código fonte de uma *app java* que será usada durante todo o semestre.
2. Executando o ficheiro *supergest.bat* (sistema Windows) ou *supergest.sh* (sistemas LINUX e MAC OS), deverá surgir o menu da Figura 20.

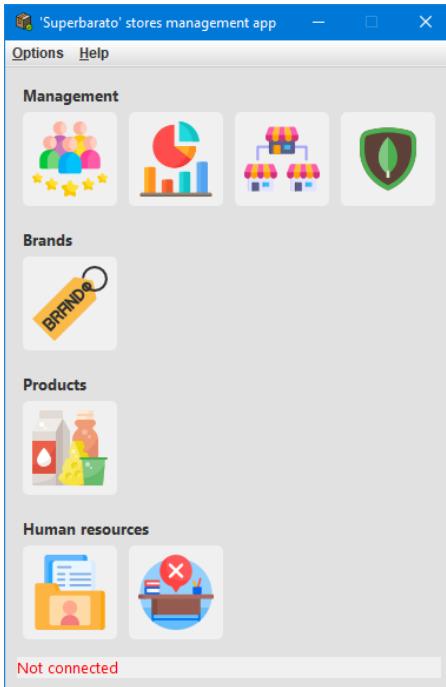


Figura 20 – Menu principal da *app* (GUI) de gestão.

3. Pelo menu **Options-> Login** (ou por combinação das teclas **Ctrl+L**), aceder ao formulário de autenticação (*logon*) introduzir as credenciais de acesso ao servidor de bases de dados.

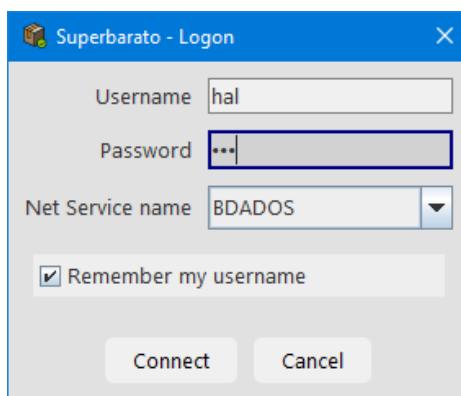


Figura 21 – Formulário de autenticação da *app* de gestão.



Neste momento, as funcionalidades da *app* não estão implementadas.
Cada estudante irá desempenhar o papel de *GUI developer* no início do capítulo IV.

Exercícios de aula

Nas aulas práticas desta unidade curricular, cada estudante irá assumir o papel de **database developer** (= *programador de base de dados*), responsável por criar e manter os objetos da base de dados que serão invocados pela GUI.

Nesta ficha, é necessário começar a compreender como utilizar uma base de dados **usando os 4 comandos CRUD disponíveis da linguagem SQL**, comandos esses que **qualquer programador** (seja da GUI ou da base de dados) deve saber usar.

Numa base de dados relacional, as operações CRUD [5] são implementadas com recurso a quatro comandos da linguagem SQL (Tabela 1).

Operação CRUD	Comando SQL	Objetivo
Create	INSERT	Criar linha(s) numa tabela
Read	SELECT	Gerar informação a partir de dados
Update	UPDATE	Atualizar célula(s) de uma tabela
Delete	DELETE	Remover linha(s) de uma tabela

Tabela 1 – Comandos da linguagem SQL para realizar as 4 operações CRUD.

Operação CRUD Read

1. No *SQL Developer*, criar uma *query* para apresentar o nome e a marca dos produtos.

output	
◊ NOME	◊ MARCA
LIXO	
LIXO	
PEN 64 GB	QINGSTON
PEN 32 GB	QINGSTON
Maçã Fuji	BemBoa
Maçã Fuji	Envy
Caju a GRANEL	
Tapete para cozinha 1mx2m	
Monitor 24" Philips 243V	Philips
Pilhas AAA	Duracell
Monitor 27" Gaming E-Line 272E1GAJ	PHILIPS
Tarte de maçã	

2. Reutilizar a *query* anterior para que os produtos sejam apresentados por ordem alfabética do nome.

output	
lista ordenada por nome, de A a Z	◊ NOME
	Caju a GRANEL
	LIXO
	LIXO
	Maçã Fuji
	Envy
	...

Operação CRUD Update

3. Existem dois produtos da marca 'QINGSTON'. Essa marca não existe, é necessário corrigi-la.
 - a) No *SQL Developer*, encontrar os produtos da marca 'QINGSTON', com uma *query*.
 - b) No *SQL Developer*, usando um comando SQL, corrigir a marca 'QINGSTON' para 'KINGSTON'.
 - c) Confirmar a operação de atualização, caso contrário, esta poderá ser anulada.

Nota: esta necessidade de confirmação irá ser estudada mais à frente na UC.

Operação CRUD Create

4. Criar um produto na categoria PERIFÉRICOS. Os dados do novo produto deverão ser os seguintes:

identificador	nome do produto	marca	preço atual	iva cobrado
2005	Rato SCULPT MOBILE FLAME RED	Microsoft	26.56	23

Operação CRUD Delete

5. Escrever e executar o comando para eliminar os produtos com nome *LIXO* (não confirmar no final).
6. Anular a operação de eliminação e verificar que foram repostos os dados anteriormente apagados.
7. Agora sim, remover **e confirmar** a remoção dos produtos com nome *LIXO*.

Referências principais

- [1] Elmasri, R. & Navathe, S. (2016). **Fundamentals of Database Systems** (7ª edição, página 4). Pearson
- [2] Elmasri, R. & Navathe, S. (2016). **Fundamentals of Database Systems** (7ª edição, capítulo 1.7). Pearson
- [3] Elmasri, R. & Navathe, S. (2016). **Fundamentals of Database Systems** (7ª edição, páginas 177-178). Pearson
- [4] Coronel, C., Morris, S. & and Rob, P. (2011). **Database Systems – Design, Implementation and Management** (9ª edição, capítulo 7). Course Technology – Cengage Learning
- [5] Elmasri, R. & Navathe, S. (2016). **Fundamentals of Database Systems** (7ª edição, capítulo 24). Pearson