




PL4 - SQL DML SELECT, Queries simples

 Category	Exercícios
 Files	BDDAD_PL04.zip PL04.S1-CREATE e ALTER TABLE.sql PL04.S2-INSERT.sql PL04-SQL DML SELECT, queries simples - PT.pdf
 Status	DONE

▼ Quick Access

[A. Projeção](#)

[B. SELEÇÃO](#)

[C. Projeção e Seleção](#)

[D. Ordenação](#)

[E. Funções de agregação](#)

▼ PL04.S1-CREATE e ALTER TABLE.sql

```
-- ** eliminar tabelas se existentes **
-- CASCADE CONSTRAINTS para eliminar as restrições de inte
-- PURGE elimina a tabela da base de dados e da "reciclage

DROP TABLE cd CASCADE CONSTRAINTS PURGE ;
DROP TABLE musica CASCADE CONSTRAINTS PURGE ;

-- ** criar tabelas **

CREATE TABLE cd (
    codCd          INTEGER CONSTRAINT pkCdCod PRIMARY KEY,
    titulo         VARCHAR(40) CONSTRAINT nnCdTitulo NOT N
    dataCompra     DATE,
    valorPago      NUMERIC(5,2),
    localCompra    VARCHAR(20)
);

CREATE TABLE musica (
```

```

        nrMusica    INTEGER,
        codCd       INTEGER,
        titulo      VARCHAR(40) CONSTRAINT nnMusicaNome NOT NU
        interprete  VARCHAR(30) CONSTRAINT nnMusicaInterprete
        duracao     NUMERIC(4,2),

        CONSTRAINT pkMusicaNrMusicaCodCd PRIMARY KEY (codCd,
);

ALTER TABLE musica ADD CONSTRAINT fkMusicaCodCd FOREIGN KE

-- ** guardar em DEFINITIVO as alterações na base de dados
-- COMMIT;

```

▼ PL04.S2-INSERT.sql

```

-- ** tabela cd **

INSERT INTO cd VALUES (1, 'Punkzilla', TO_DATE('22/April/20
INSERT INTO cd VALUES (2, 'Classic Duets', TO_DATE('21/May
INSERT INTO cd VALUES (3, 'The Wall', TO_DATE('22/April/20
INSERT INTO cd VALUES (4, 'Hits 4', TO_DATE('10/September/
INSERT INTO cd VALUES (5, 'Songs of Experience', TO_DATE('
INSERT INTO cd VALUES (6, 'Giesta 2', TO_DATE('15/June/201

-- ** tabela musica **

INSERT INTO musica VALUES (1, 1, 'Dream of Waking', 'AFI',
INSERT INTO musica VALUES (2, 1, 'Still', 'Rufio', 3.02);
INSERT INTO musica VALUES (3, 1, 'Behind the Music', 'The
INSERT INTO musica VALUES (4, 1, 'Why Are You Alive', 'The
INSERT INTO musica VALUES (5, 1, 'Vandals', 'The Vandals',
INSERT INTO musica VALUES (6, 1, 'Days of the Phoenix', 'A
INSERT INTO musica VALUES (7, 1, 'Wester', 'AFI', 3.02);

INSERT INTO musica VALUES (1, 2, 'Bizet: Les pêcheurs de p
INSERT INTO musica VALUES (2, 2, 'Verdi: Otello/Act 2', 'O
INSERT INTO musica VALUES (3, 2, 'Verdi: Aida/Act 4', 'Lor

```

```

INSERT INTO musica VALUES (4, 2, 'Puccini: Turandot', 'Zub
INSERT INTO musica VALUES (1, 3, 'In the Flesh', 'Pink Flo
INSERT INTO musica VALUES (2, 3, 'The Thin Lee', 'Pink Flo
INSERT INTO musica VALUES (3, 3, 'Mother', 'Pink Floyd', 5
INSERT INTO musica VALUES (4, 3, 'Don''t Leave Me Now', 'P
INSERT INTO musica VALUES (5, 3, 'Young Lust', 'Pink Floyd

INSERT INTO musica VALUES (1, 4, 'It''s Alright(Baby''s Co
INSERT INTO musica VALUES (2, 4, 'Hounds of Love' , 'Kate
INSERT INTO musica VALUES (3, 4, 'Calling America', 'Elect
INSERT INTO musica VALUES (4, 4, 'Suspicious Minds', 'Fine
INSERT INTO musica VALUES (5, 4, 'I''m Your Man', 'Wham!',

INSERT INTO musica VALUES (1, 5, 'Love is All We Have Left
INSERT INTO musica VALUES (2, 5, 'Lights of Home' , 'U2',
INSERT INTO musica VALUES (3, 5, 'You''re the Best Thing A
INSERT INTO musica VALUES (4, 5, 'Get Out of Your Own Way'
INSERT INTO musica VALUES (5, 5, 'American Soul', 'U2', 4.
INSERT INTO musica VALUES (6, 5, 'Summer of Love', 'U2', 3
INSERT INTO musica VALUES (7, 5, 'Red Flag Day', 'U2', 3.1
INSERT INTO musica VALUES (8, 5, 'The Showman', 'U2', 3.23

INSERT INTO musica VALUES (1, 6, 'Valsa em Espiral', 'Migu
INSERT INTO musica VALUES (2, 6, '1987' , 'Miguel Araújo',
INSERT INTO musica VALUES (3, 6, 'Meio Conto', 'Miguel Ara
INSERT INTO musica VALUES (4, 6, 'Via Norte', 'Miguel Araújo
INSERT INTO musica VALUES (5, 6, 'Sangemil', 'Miguel Araújo
INSERT INTO musica VALUES (6, 6, 'Lurdes Valsa Lenta', 'Mi
INSERT INTO musica VALUES (7, 6, 'Axl Rose', 'Miguel Araújo
INSERT INTO musica VALUES (8, 6, '20% Mais', 'Miguel Araújo
INSERT INTO musica VALUES (9, 6, 'Vândalo', 'Miguel Araújo
INSERT INTO musica VALUES (10, 6, 'Aqui Dali', 'Miguel Ara

-- ** guardar em DEFINITIVO as alterações na base de dados
-- COMMIT;

```

▼ 1) Mostrar todos os dados da tabela CD

```
SELECT * FROM cd;
```

- ▼ 2) Mostrar todos os dados da tabela Musica

```
SELECT * FROM musica;
```

A. Projeção

- ▼ 1) Mostrar o título e a data de compra de todos os CD;

```
SELECT titulo, dataCompra FROM cd;
```

- ▼ 2) Mostrar o resultado da alínea anterior, mas sem registos repetidos

```
SELECT DISTINCT titulo, dataCompra FROM cd;
```

- ▼ 3) Mostrar o código dos CD e os respetivos intérpretes, sem registos repetidos

```
SELECT DISTINCT codCd, interprete FROM musica;
```

- ▼ 4) Mostrar o resultado anterior com a primeira coluna intitulada "Código do CD"

```
SELECT DISTINCT codCd AS "Código do CD", interprete FROM m
```

- ▼ 5) Mostrar o título, o valor pago e o respetivo valor do IVA de todos os CD. O valor do IVA é calculado de acordo com a seguinte fórmula: $\text{valor do IVA} = (\text{valor pago} * 0.23) / 1.23$

```
SELECT
    titulo,
    valorPago,
    (valorPago * 0.23) / 1.23 AS valorIVA
FROM cd;
```

B. SELEÇÃO

- ▼ 1) Mostrar todos os dados das músicas do CD com o código 2

```
SELECT * FROM CD WHERE codCd = 2;
```

- ▼ 2) Mostrar todos os dados das músicas que não pertencem ao CD com o código 2

```
SELECT * FROM CD WHERE codCd != 2;
```

- ▼ 3) Mostrar todos os dados das músicas do CD com o código 2 cuja duração pertença ao intervalo [4,6]

```
SELECT * FROM musica WHERE codCd = 2 AND duracao BETWEEN 4
```

▼ **AND**

- Usado para filtrar registos baseados em mais de uma condição
- Mostra o registo se todas as condições forem verdadeiras

▼ **BETWEEN**

- Pode ser utilizada em campos que armazenam valores do tipo data e hora
- Permite a seleção de um intervalo de valores

- ▼ 4) Mostrar todos os dados das músicas do CD com o código 2 cuja duração seja inferior a 4 ou superior a 6;

```
SELECT *  
FROM musica  
WHERE codCd = 2 AND (duracao < 4 OR duracao > 6);
```

▼ **OR**

- Usado para filtrar registos baseados em mais de uma condição
- Mostra um registo se pelo menos uma das condições for verdadeira

▼ **NOT**

- Inverte o estado lógico

▼ 5) Mostrar todos os dados das músicas com o número: 1, 3, 5 ou 6

```
SELECT *  
FROM musica  
WHERE nrMusica IN (1, 3, 5, 6);
```

▼ **IN**

- É usada para especificar múltiplos valores em uma condição de pesquisa.
- Permite selecionar registros que correspondam a qualquer um dos valores especificados na lista.
- Útil para filtrar registros com base em uma lista de valores possíveis. Simplifica a escrita da consulta

▼ 6) Mostrar todos os dados das músicas com o número diferente de 1, 3, 5 e 6

```
SELECT *  
FROM musica  
WHERE nrMusica NOT IN (1, 3, 5, 6);
```

▼ **NOT IN**

- Usado para verificar se um valor não está presente em um conjunto de valores especificados
- Retorna verdadeiro se o valor não estiver presente e falso se estiver

▼ 7) Mostrar todos os dados das músicas cujo intérprete é uma orquestra

```
SELECT *  
FROM musica  
WHERE interprete LIKE '%Orquestra%';
```

▼ **LIKE**

- Utilizada para realizar buscas de padrões em uma determinada coluna de uma tabela
- Utilizada em conjunto com os caracteres curinga **%** e **_**
- **%** representa qualquer sequência de caracteres em um padrão

- `%` representa qualquer caractere único em um padrão
- É possível buscar registros que correspondam a um padrão específico

▼ 8) Mostrar todos os dados das músicas cujo nome do intérprete tem a letra Y

```
SELECT *
FROM musica
WHERE interprete LIKE '%Y%';
```

▼ 9) Mostrar todos os dados das músicas cujo nome termina com DAL?, sendo ? qualquer carácter

```
SELECT *
FROM musica
WHERE interprete LIKE '%DAL?';
```

▼ 10) Mostrar todos os dados das músicas cujo título tem o carácter %;

```
SELECT *
FROM musica
WHERE titulo LIKE '%\%%' ESCAPE '\';
```

▼ ESCAPE

The **ESCAPE** clause is used to specify an escape character, and in this case, `\` is used as the escape character. This query will select all columns from the `musica` table where the title contains the character `%`.

▼ 11) Mostrar todos os dados das músicas cujo título é iniciado pela letra B, D ou H

```
SELECT *
FROM musica
WHERE titulo LIKE 'B%' OR titulo LIKE 'D%' OR titulo LIKE
```

▼ 12) Mostrar todos os dados dos CD sem o local de compra registado

```
SELECT *  
FROM cd  
WHERE localCompra IS NULL;
```

- ▼ 13) Mostrar todos os dados dos CD com o local de compra registado

```
SELECT *  
FROM cd  
WHERE localCompra IS NOT NULL;
```

CODCD	TITULO	DATACOMPRA	VALORPAGO	LOCALCOMPRA
1	Punkzilla	22-APR-13	55	FNAC
2	Classic Duets	21-MAY-19	30.5	Worten
3	The Wall	22-APR-13	25.8	FNAC
4	Hits 4	10-SEP-19	42.3	Worten

C. Projeção e Seleção

- ▼ 1) Mostrar os títulos dos CD comprados na FNAC

```
SELECT *  
FROM cd  
WHERE localCompra = "FNAC";
```

CODCD	TITULO	DATACOMPRA	VALORPAGO	LOCALCOMPRA
1	Punkzilla	22-APR-13	55	FNAC
3	The Wall	22-APR-13	25.8	FNAC

- ▼ 2) Mostrar os títulos dos CD que não foram comprados na FNAC


```
SELECT *
FROM cd
WHERE localCompra != 'FNAC';
```

CODCD	TITULO	DATACOMPRA	VALORPAGO	LOCALCOMPRA
2	Classic Duets	21-MAY-19	30.5	Worten
4	Hits 4	10-SEP-19	42.3	Worten

D. Ordenação

- ▼ 1) Mostrar os títulos dos CD que não foram comprados na FNAC, por ordem alfabética inversa

```
SELECT *
FROM cd
WHERE localCompra != 'FNAC'
ORDER BY titulo DESC;
```

CODCD	TITULO	DATACOMPRA	VALORPAGO	LOCALCOMPRA
4	Hits 4	10-SEP-19	42.3	Worten
2	Classic Duets	21-MAY-19	30.5	Worten

- ▼ 2) Mostrar o título e a data de compra dos CD, por ordem descendente da data de compra do CD;

```
SELECT titulo, dataCompra
FROM cd
ORDER BY dataCompra DESC;
```

TITULO	DATACOMPRA
Songs of Experience	01-JAN-20
Hits 4	10-SEP-19
Giesta 2	15-JUN-19
Classic Duets	21-MAY-19
Punkzilla	22-APR-13
The Wall	22-APR-13

▼ 3) Mostrar o título e o local de compra dos CD, por ordem ascendente do local de compra do CD

```
SELECT titulo, localCompra
FROM cd
ORDER BY localCompra ASC;
```

TITULO	LOCALCOMPRA
Punkzilla	FNAC
The Wall	FNAC
Hits 4	Worten
Classic Duets	Worten
Giesta 2	-
Songs of Experience	-

▼ 4) Mostrar o título, o valor pago e o respetivo valor do IVA dos CD, por ordem decrescente do IVA;

```
SELECT titulo, valorPago, (valorPago * 0.23) / 1.23 AS IVA
FROM cd
```

```
ORDER BY IVA DESC;
```

TITULO	VALORPAGO	IVA
Punkzilla	55	10.284552845528455284552845528455285
Hits 4	42.3	7.90975609756097560975609756097561
Classic Duets	30.5	5.703252032520325203252032520325203
The Wall	25.8	4.824390243902439024390243902439024
Songs of Experience	10.99	2.055040650406504065040650406504065
Giesta 2	9.1	1.701626016260162601626016260162602

▼ 5) Mostrar o título do CD por ordem descendente da data de compra e, no caso da igualdade de datas, por ordem alfabética do título.

```
SELECT titulo  
FROM cd  
ORDER BY dataCompra DESC, titulo ASC;
```

TITULO
Songs of Experience
Hits 4
Giesta 2
Classic Duets
Punkzilla
The Wall

E. Funções de agregação

▼ 1) Mostrar a quantidade de locais de compra distintos

```
SELECT DISTINCT localCompra  
FROM cd;
```

LOCALCOMPRA
-
FNAC
Worten

- ▼ 2) Mostrar o total gasto com a compra de todos os CD, o maior e o menor valor pago por um CD

```
SELECT SUM(valorPago) AS Soma_Total, MAX(valorPago) AS Maior_valor_pago, MIN(valorPago) AS Menor_valor_pago
FROM cd;
```

SOMA_TOTAL	MAIOR_VALOR_PAGO	MENOR_VALOR_PAGO
173.69	55	9.1

- ▼ 3) Mostrar a média da duração de todas as músicas

```
SELECT AVG(duracao) AS Duração_Media
FROM musica;
```

DURAÇÃO_MEDIA
3.53128205128205128205128205128205128205

- ▼ 4) Mostrar o total do valor pago na FNAC

```
SELECT SUM(valorPago) AS Total_Pago_FNAC
FROM cd
WHERE localCompra = 'FNAC';
```

TOTAL_PAGO_FNAC
80.8

-
- ▼ 5) Mostrar a diferença entre o maior e o menor valor pago na FNAC

```
SELECT MAX(valorPago) - MIN(valorPago) AS Diferença  
FROM cd  
WHERE localCompra = 'FNAC';
```

DIFERENÇA
29.2