



**Universidade do Minho**  
Escola de Engenharia

**Mestrado em Engenharia Informática**

**Unidade Curricular Complementar**

**Base de Dados NoSQL**

Ficha de Exercícios 01

4º Ano, 1º Semestre

Ano letivo 2020/2021

Diogo Alexandre Rodrigues Lopes

PG42823

Joel Costa Carvalho

PG42837

Ana Margarida da Rocha Ferreira

PG44412

Bruno Xavier Brás dos Santos

PG44414

## Conteúdo

<b>Tarefa 1.</b> Crie um <i>schema</i> Oracle utilizando os comandos fornecidos nos slides das aulas, recorrendo ao utilizador <i>system</i> da <i>pluggable database</i> .....	3
<b>Tarefa 2.</b> Faça a execução do modelo físico para a utilização em Oracle.....	3
<b>Tarefa 3.</b> Com os dados importados e utilizando os conhecimentos SQL adquiridos anteriormente, responda às seguintes questões, indicando o comando SQL utilizado para chegar à resposta:.....	5
<b>a)</b> Quantos títulos possui a coleção?.....	5
<b>b)</b> Quantas músicas no total possui toda a coleção? .....	5
<b>c)</b> Quantos autores existem na coleção? .....	6
<b>d)</b> Quantas editoras distintas existem na coleção?.....	6
<b>e)</b> O autor “Max Changmin” é o principal autor de quantos títulos? .....	6
<b>f)</b> No ano de 1970, quais foram os títulos comprados pelo utilizador? .....	6
<b>g)</b> Qual o autor do título que foi adquirido em “01-02-2010”, cujo preço foi de 12€? .....	6
<b>h)</b> Na alínea anterior indique qual a editora desse título?.....	6
<b>i)</b> Quais as reviews (data e classificação) existentes para o título “oh whoa oh” .....	7
<b>j)</b> Quais as reviews (data e classificação) existentes para o título “pump”, ordenadas por data da mais antiga para a mais recente? .....	7
<b>k)</b> Quais os diversos autores das músicas do título lançado a ‘04-04-1970’ com o preço de 20€?7	
<b>l)</b> Qual foi total de dinheiro investido em compras de título da editora ‘EMI’? .....	7
<b>m)</b> Qual o título mais antigo cujo preço foi de 20€? .....	7
<b>n)</b> Quantos “MP3” tem a coleção? .....	8
<b>o)</b> Destes mp3 quais são os títulos cujo género é: Pop Rock? .....	8
<b>p)</b> Qual o custo total com “Blue-Ray”?.....	8
<b>q)</b> Qual o custo total com “Blue-Ray” cuja editora é a EMI?.....	8
<b>r)</b> Qual o património total dos títulos da coleção?.....	8
<b>s)</b> Qual a editora na qual o colecionador investiu mais dinheiro?.....	9
<b>t)</b> Qual a editora que possui mais títulos de “Heavy Metal” na coleção? Quanto título possui essa editora? .....	9

**Tarefa 1.** Crie um *schema* Oracle utilizando os comandos fornecidos nos slides das aulas, recorrendo ao utilizador *system* da *pluggable database*

**Create Tablespace:**

```
CREATE TABLESPACE music_tables DATAFILE 'music_files_01.dbf' SIZE 200mb;
```

**Create User:**

```
CREATE USER music IDENTIFIED BY "music2020" DEFAULT TABLESPACE  
music_tables QUOTA UNLIMITED ON music_tables;
```

**Grant Permissions:**

```
GRANT CONNECT, RESOURCE, CREATE VIEW, CREATE SEQUENCE TO music;
```

**Tarefa 2.** Faça a execução do modelo físico para a utilização em Oracle

**CREATE TABLE AUTOR**

```
CREATE TABLE autor(  
    "id_autor" Number(3, 0) NOT NULL ENABLE,  
    "nome" VARCHAR2(200 byte) NOT NULL ENABLE,  
    CONSTRAINT "autor_PK" PRIMARY KEY ("id_autor")  
);
```

**CREATE TABLE EDITORA**

```
CREATE TABLE editora(  
    "id_editora" Number(3, 0) NOT NULL ENABLE,  
    "nome" VARCHAR2(200 byte) NOT NULL ENABLE,  
    CONSTRAINT "editora_PK" PRIMARY KEY ("id_editora")  
);
```

## CREATE TABLE GENERO

```
CREATE TABLE genero (  
    "id_genero" Number(3, 0) NOT NULL ENABLE,  
    "nome" VARCHAR2(200 byte) NOT NULL ENABLE,  
    CONSTRAINT "genero_PK" PRIMARY KEY ("id_genero")  
);
```

## CREATE TABLE SUPORTE

```
CREATE TABLE suporte (  
    "id_suporte" Number(3, 0) NOT NULL ENABLE,  
    "nome" VARCHAR2(200 byte) NOT NULL ENABLE,  
    CONSTRAINT "suporte_PK" PRIMARY KEY ("id_suporte")  
);
```

## CREATE TABLE TITULO

```
CREATE TABLE titulo (  
    "id_titulo" NUMBER(5,0) NOT NULL ENABLE,  
    "titulo" VARCHAR2(200 BYTE),  
    "preco" NUMBER,  
    "dta_compra" DATE,  
    "id_editora" NUMBER(3,0),  
    "id_suporte" NUMBER(3,0),  
    "id_genero" NUMBER(3,0),  
    "id_autor" NUMBER(3,0),  
    CONSTRAINT "TITULO_PK" PRIMARY KEY ("id_titulo"),  
    CONSTRAINT "TITULO_FK1" FOREIGN KEY ("id_editora")  
        REFERENCES "MUSIC"."EDITORA" ("id_editora") ENABLE,  
    CONSTRAINT "TITULO_FK2" FOREIGN KEY ("id_suporte")  
        REFERENCES "MUSIC"."SUPORTE" ("id_suporte") ENABLE,  
    CONSTRAINT "TITULO_FK3" FOREIGN KEY ("id_genero")  
        REFERENCES "MUSIC"."GENERO" ("id_genero") ENABLE,  
    CONSTRAINT "TITULO_FK4" FOREIGN KEY ("id_autor")  
        REFERENCES "MUSIC"."AUTOR" ("id_autor") ENABLE  
);
```

## CREATE TABLE MUSICA

```
CREATE TABLE musica (  
    "id_musica" NUMBER(5,0) NOT NULL ENABLE,  
    "nome" VARCHAR2(20 BYTE) NOT NULL ENABLE,  
    "id_autor" NUMBER(3,0) NOT NULL ENABLE,  
    "id_titulo" NUMBER(5,0) NOT NULL ENABLE,  
    CONSTRAINT "MUSICA_PK" PRIMARY KEY ("id_musica"),  
    CONSTRAINT "MUSICA_FK1" FOREIGN KEY ("id_autor")  
        REFERENCES "MUSIC"."AUTOR" ("id_autor") ENABLE,  
    CONSTRAINT "MUSICA_FK2" FOREIGN KEY ("id_titulo")  
        REFERENCES "MUSIC"."TITULO" ("id_titulo") ENABLE  
);
```

## CREATE TABLE REVIEW

```
CREATE TABLE review (  
    "id_review" NUMBER(5,0) NOT NULL ENABLE,  
    "id_titulo" NUMBER(5,0) NOT NULL ENABLE,  
    "dta_review" DATE NOT NULL ENABLE,  
    "conteudo" VARCHAR2(500 BYTE) NOT NULL ENABLE,  
    CONSTRAINT "REVIEW_PK" PRIMARY KEY ("id_review"),  
    CONSTRAINT "REVIEW_FK1" FOREIGN KEY ("id_titulo")  
        REFERENCES "MUSIC"."TITULO" ("id_titulo") ENABLE  
);
```

**Tarefa 3.** Com os dados importados e utilizando os conhecimentos SQL adquiridos anteriormente, responda às seguintes questões, indicando o comando SQL utilizado para chegar à resposta:

- a) Quantos títulos possui a coleção?

```
select count(*) from titulo;
```

- b) Quantas músicas no total possui toda a coleção?

```
select count(*) from musica;
```

- c) Quantos autores existem na coleção?

```
select count(id_autor) from autor;
```

- d) Quantas editoras distintas existem na coleção?

```
select distinct count(*) from editora;
```

- e) O autor “Max Changmin” é o principal autor de quantos títulos?

```
select count(*) from titulo
join autor
on autor.id_autor = titulo.id_autor
where autor.nome = 'Max Changmin';
```

- f) No ano de 1970, quais foram os títulos comprados pelo utilizador?

```
select * from titulo
where dta_compra >= TO_DATE('01/01/1970', 'dd/mm/yyyy') and
dta_compra <= TO_DATE('31/12/1970', 'dd/mm/yyyy');
```

- g) Qual o autor do título que foi adquirido em “01-02-2010”, cujo preço foi de 12€?

```
select autor.nome from titulo
join autor
on autor.id_autor = titulo.id_autor
where dta_compra = TO_DATE('01/02/2010', 'dd/mm/yyyy') and
preco = 12;
```

- h) Na alínea anterior indique qual a editora desse título?

```
select editora.* from titulo
join autor
on autor.id_autor = titulo.id_autor
join editora
on editora.id_editora = titulo.id_editora
where dta_compra = TO_DATE('01/02/2010', 'dd/mm/yyyy') and
preco = 12;
```

- i) Quais as reviews (data e classificação) existentes para o título “oh whoa oh”

```
select review.dta_review, review.conteudo from review
join titulo
on titulo.id_titulo = review.id_titulo
where titulo.titulo = 'oh whoa oh';
```

- j) Quais as reviews (data e classificação) existentes para o título “pump”, ordenadas por data da mais antiga para a mais recente?

```
select review.dta_review, review.conteudo from review
join titulo
on titulo.id_titulo = review.id_titulo
where titulo.titulo = 'pump'
order by review.dta_review desc;
```

- k) Quais os diversos autores das músicas do título lançado a '04-04-1970' com o preço de 20€?

```
select autor.* from autor
join titulo
on titulo.id_autor = autor.id_autor
where titulo.dta_compra = TO_DATE('04/04/1970', 'dd/mm/yyyy')
and titulo.preco = 20;
```

- l) Qual foi total de dinheiro investido em compras de título da editora 'EMI'?

```
select sum(preco) from titulo
join editora
on editora.id_editora = titulo.id_editora
where editora.nome = 'EMI';
```

- m) Qual o título mais antigo cujo preço foi de 20€?

```
select titulo.* from titulo
where preco = 20
order by dta_compra asc
fetch first 1 rows only;
```

n) Quantos “MP3” tem a coleção?

```
select count(*) from titulo
join suporte
on suporte.id_suporte = titulo.id_suporte
where suporte.nome = 'MP3';
```

o) Destes mp3 quais são os títulos cujo género é: Pop Rock?

```
select count(*) from titulo
join genero
on genero.id_genero = titulo.id_genero
where titulo.id_suporte = 5 and genero.nome = 'Pop Rock';
```

p) Qual o custo total com “Blue-Ray”?

```
select sum(preco) from titulo
join suporte
on suporte.id_suporte = titulo.id_suporte
where suporte.nome = 'Blue-Ray';
```

q) Qual o custo total com “Blue-Ray” cuja editora é a EMI?

```
select sum(preco) from titulo
join suporte
on suporte.id_suporte = titulo.id_suporte
join editora
on editora.id_editora = titulo.id_editora
where editora.nome = 'EMI' and suporte.nome = 'Blue-Ray';
```

r) Qual o património total dos títulos da coleção?

```
select sum(preco) from titulo;
```



s) Qual a editora na qual o colecionador investiu mais dinheiro?

```
select editora.nome, sum(titulo.PRECO) from editora
join titulo
on titulo.id_editora = editora.id_editora
  by editora.NOME
order by sum(titulo.PRECO) desc
fetch first
rows only;
```

t) Qual a editora que possui mais títulos de “Heavy Metal” na coleção? Quanto título possui essa editora?

```
select editora.nome, count(editora.nome) from editora
join titulo
on titulo.id_editora = editora.id_editora
join genero
on genero.id_genero = titulo.id_genero
where genero.nome = 'Heavy Metal'
group by editora.nome;
```