



Universidade do Minho
Escola de Engenharia

Bases de Dados NoSQL

Mestrado em Bioinformática

Resolução da Ficha de Trabalho 1

Docente: António Abelha / Cristiana Neto

Aluno

Duarte Alves Velho A96815

2023/2024

➔ FE01 – “Introdução ao RDBM Oracle”

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

Poderá por exemplo criar um user “music” com as credenciais “music2020”.

O tablespace chamar-se-á music_tables e o data file music_files_01.dbf com 200Mb de espaço

Considerando:

```
AUTOR {ID_AUTOR, NOME}
EDITORA {ID_EDITORA, NOME}
GENERO {ID_GENERO, NOME}
SUPORTE {ID_SUPORTE, NOME}
TITULO (ID_TITULO, TITULO, PRECO, DTA_COMPRA, ID_EDITORA, ID_SUPORTE, ID_GENERO, ID_AUTOR)
MUSICA {ID_MUSICA, NOME, ID_AUTOR, ID_TITULO}
REVIEW { ID_REVIEW, ID_TITULO, DTA_REVIEW, CONTEUDO}
```

Faça a execução do modelo físico para a utilização em Oracle.

- CREATE TABLESPACE music_tables DATAFILE 'music_files_01.dbf' SIZE 200m;
- ALTER SESSION
SET "_ORACLE_SCRIPT"=TRUE;
- CREATE USER music IDENTIFIED BY music2020 DEFAULT TABLESPACE music_tables QUOTA
UNLIMITED ON music_tables;
- GRANT CONNECT, RESOURCE, CREATE VIEW, CREATE SEQUENCE TO music;

[2] Utilizando o sqldeveloper, efetue a ligação ao Schema oracle criado corra os scripts obtidos na questão [1] e crie toda a infraestrutura da base de dados.

Uma vez que na minha máquina tive algumas dificuldades em correr o sql Developer, recorri ao “liveoracle” para responder às questões da ficha.

Fiz upload de todos os scripts presentes na BlackBoard para criar a base de dados. Para conseguir criar todas as tabelas, foi necessário criar um código respetivo à tabela “Título” que estava em falta nesses mesmos scripts:

```
CREATE TABLE TITULO (
  "ID_TITULO" NUMBER(4,0) NOT NULL ENABLE,
  "TITULO" VARCHAR(200) NOT NULL ENABLE,
  "PRECO" NUMBER(6,2) NOT NULL ENABLE,
  "DTA_COMPRA" DATE NOT NULL ENABLE,
  "ID_EDITORA" NUMBER(4,0) NOT NULL ENABLE,
  "ID_SUPORTE" NUMBER(4,0) NOT NULL ENABLE,
  "ID_GENERO" NUMBER(4,0) NOT NULL ENABLE,
  "ID_AUTOR" NUMBER(4,0) NOT NULL ENABLE,
  CONSTRAINT "TITULO_PK" PRIMARY KEY ("ID_TITULO"),
  CONSTRAINT "TITULO_FK1" FOREIGN KEY ("ID_EDITORA")
  REFERENCES EDITORA("ID_EDITORA") ENABLE,
  CONSTRAINT "TITULO_FK2" FOREIGN KEY ("ID_SUPORTE")
  REFERENCES SUPORTE("ID_SUPORTE") ENABLE,
```

```
CONSTRAINT "TITULO_FK3" FOREIGN KEY ("ID_GENERO")
REFERENCES GENERO ("ID_GENERO"),
CONSTRAINT "TITULO_FK4" FOREIGN KEY ("ID_AUTOR")
REFERENCES AUTOR ("ID_AUTOR") ENABLE)
```

[3] Com a estrutura da Base de Dados criada importe o ficheiro **scripts.sql** que poderá encontrar na página da UC. Nomeadamente:
editora.sql | título.sql | review.sql | autor.sql | genero.sql | suporte.sql | musica.sql

No processo de desenvolvimento do código, corri os scripts todos na seguinte ordem: criei as tabelas de “Autor”, “Editora”, “Género” e “Suporte”, posteriormente, povoei as mesmas, criei a tabela “Título”, e povoei a mesma, por último criei as tabelas “Música” e “Review” e povoei as mesmas ficando assim com a base de dados completa.

Desenvolvi a correção dos exercícios no Live SQL ao invés de utilizar o Oracle SQL Developer, e importei o script desenvolvido. Este foi enviado em anexo com o nome “music.sql”.

[4] 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;
```

R: 49

b. Quantas músicas no total possui toda a coleção?

```
SELECT COUNT(*) FROM MUSICA;
```

R: 499

c. Sem utilizar tabela autores, quantos autores existem na coleção?

```
SELECT COUNT(DISTINCT ID_AUTOR) FROM TITULO;
```

R: 36

d. Quantas editoras distintas existem na coleção?

```
SELECT COUNT(DISTINCT ID_EDITORA) FROM TITULO;
```

R: 5

e. O autor “Max Changmin” é o principal autor de quantos título?

```
SELECT COUNT(*) FROM TITULO WHERE ID_AUTOR = (SELECT ID_AUTOR FROM
AUTOR WHERE NOME = 'Max Changmin');
```

R: 2

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

```
SELECT TITULO FROM TITULO WHERE EXTRACT(YEAR FROM DTA_COMPRA) =
1970;
```

R:

```
TITULO
boogie
love
woman
doo
ron
get get get get
float
band
```

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

```
SELECT A.NOME FROM AUTOR A JOIN TITULO T ON A.ID_AUTOR = T.ID_AUTOR
WHERE T.DTA_COMPRA = '01-FEB-2010' AND T.PRECO = 12;
```

R: Yumi Yoshimura

h. Na alínea anterior indique nome da editora desse título?

```
SELECT E.NOME FROM EDITORA E JOIN TITULO T ON E.ID_EDITORA =
T.ID_EDITORA WHERE T.DTA_COMPRA = '01-FEB-2010' AND T.PRECO = 12;
```

R: EMI

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

```
SELECT DTA_REVIEW, CONTEUDO FROM REVIEW WHERE ID_TITULO = (SELECT
ID_TITULO FROM TITULO WHERE TITULO = 'oh whoa oh');
```

R:

```
DTA_REVIEW CONTEUDO
07-MAY-17 EXCELENTE
24-MAY-16 EXCELENTE
24-JUL-17 EXCELENTE
```

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

```
SELECT DTA_REVIEW, CONTEUDO FROM REVIEW WHERE ID_TITULO = (SELECT
ID_TITULO FROM TITULO WHERE TITULO = 'pump') ORDER BY DTA_REVIEW;
```

R:

```
DTA_REVIEW CONTEUDO
12-JUL-14 EXCELENTE
04-JAN-15 OBRA PRIMA
01-FEB-16 PÉSSIMO
24-SEP-18 PÉSSIMO
```

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

```
SELECT DISTINCT AUTOR.NOME  
FROM AUTOR  
JOIN MUSICA ON AUTOR.ID_AUTOR = MUSICA.ID_AUTOR  
JOIN TITULO ON MUSICA.ID_TITULO = TITULO.ID_TITULO  
WHERE TITULO.DTA_COMPRA = TO_DATE('04-04-1970', 'DD-MM-YYYY')  
AND TITULO.PRECO = 20;
```

R:

AUTOR
Marisa Liz
Matt Cardle
Ryan Tedder
Han Hye-ri
Bruno Branco
Fiuk
Robbie Williams
Katy Perry
Piret Järvis
Haeryung

l. Qual foi o total de dinheiro investido em compras de título da editora ‘EMI’?

```
SELECT SUM(TITULO.PRECO) AS TOTAL_INVESTIDO  
FROM TITULO  
JOIN EDITORA ON TITULO.ID_EDITORA = EDITORA.ID_EDITORA  
WHERE EDITORA.NOME = 'EMI';
```

R: 104

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;
```

R: woman

n. Quantos “MP3” tem a coleção?

```
SELECT COUNT(*)  
  
FROM TITULO  
  
JOIN SUPORTE ON TITULO.ID_SUPORTE = SUPORTE.ID_SUPORTE  
  
WHERE SUPORTE.NOME = 'MP3';
```

R: 7

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

```
SELECT TITULO.TITULO  
  
FROM TITULO  
  
JOIN SUPORTE ON TITULO.ID_SUPORTE = SUPORTE.ID_SUPORTE  
  
JOIN GENERO ON TITULO.ID_GENERO = GENERO.ID_GENERO  
  
WHERE SUPORTE.NOME = 'MP3'  
  
AND GENERO.NOME = 'Pop Rock';
```

R:

```
      TITULO  
      imma  
      pop
```

p. Qual o custo total com “Blue-Ray”?

```
SELECT SUM(PRECO) AS TOTAL_CUSTO  
  
FROM TITULO  
  
JOIN SUPORTE ON TITULO.ID_SUPORTE = SUPORTE.ID_SUPORTE  
  
WHERE SUPORTE.NOME = 'Blue-Ray';
```

R: 72

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

```
SELECT SUM(TITULO.PRECO) AS TOTAL_CUSTO  
  
FROM TITULO  
  
JOIN SUPORTE ON TITULO.ID_SUPORTE = SUPORTE.ID_SUPORTE  
  
JOIN EDITORA ON TITULO.ID_EDITORA = EDITORA.ID_EDITORA  
  
WHERE SUPORTE.NOME = 'Blue-Ray'  
  
AND EDITORA.NOME = 'EMI';
```

R: 48

r. Qual(ais) a(s) editora(s) na qual o colecionador investiu mais dinheiro e qual o valor do investimento?

```

SELECT EDITORA.NOME, SUM(TITULO.PRECO) AS TOTAL_INVESTIDO
FROM TITULO
JOIN EDITORA ON TITULO.ID_EDITORA = EDITORA.ID_EDITORA
GROUP BY EDITORA.NOME
ORDER BY SUM(TITULO.PRECO) DESC
FETCH FIRST 1 ROWS ONLY;

```

R:

EDITORA	TOTAL_INVESTIDO
Apple	113

- t. Qual a editora que possui mais títulos de “Heavy Metal” na coleção? Quantos título possui essa editora?

```

SELECT EDITORA.NOME, COUNT(*) AS TOTAL_TITULOS
FROM TITULO
JOIN EDITORA ON TITULO.ID_EDITORA = EDITORA.ID_EDITORA
JOIN GENERO ON TITULO.ID_GENERO = GENERO.ID_GENERO
WHERE GENERO.NOME = 'Heavy Metal'
GROUP BY EDITORA.NOME
ORDER BY COUNT(*) DESC
FETCH FIRST 1 ROWS ONLY;

```

R:

EDITORA	TOTAL_TITULOS
Valentim de Carvalho	3

[5] Insira uma coluna na tabela review denominada nota. Preencha esta coluna tendo em conta a coluna conteúdo: obra prima – 5, excelente – 4, bom – 3, mau – 2, péssimo – 1. Mostre os comandos SQL que utilizou.

```

UPDATE REVIEW
SET NOTA= CASE CONTEUDO
    WHEN 'OBRA PRIMA' THEN 5
    WHEN 'EXCELENTE' THEN 4
    WHEN 'BOM' THEN 3
    WHEN 'MAU' THEN 2
    WHEN 'PÉSSIMO' THEN 1
    ELSE NULL

```

END;