

Responda a todas as perguntas nesta folhas, no espaço reservado para o efeito. Por favor, não ultrapasse esse espaço. A cotação de 0 a 20 para cada pergunta está indicada na margem esquerda dentro de um quadrado. Duração: 2 horas.

Nº: _____ Nome: _____

Exercício 1

1. Considere as seguintes relações:

```
estudantes(enum: integer, enome: string, datanascimento date, anocurso integer);
turmas(tnome: string, sala: string, fid: integer);
matriculas(enum: integer, tnome: string);
faculdades(fid: integer, fnome: string, deptid: integer);
```

3

- (a) Escreva em Álgebra relacional as expressões para implementar as questões:

- i. Qual é o nome (*enome*) da faculdade em que o aluno *tnome* = 'JOAQUIM SILVA' está matriculado em pelo menos uma turma?

.....
.....

- ii. Qual é o nome das faculdades que não têm ainda qualquer matrícula registada?

.....
.....
.....

- iii. Qual é o nome dos estudantes matriculados ao mesmo tempo nas turmas *tnome* = 'SD01' e *tnome* = 'SD02'?

.....
.....
.....

1

- (b) Escreva em PL/SQL uma função que dado o código de uma faculdade *fid* calcule a cardinalidade do conjunto de turmas diferentes que oferece.

.....
.....
.....
.....
.....
.....
.....

3¹/₂

(c) Escreva em SQL as respostas às questões:

i. Qual é o nome (*enome*) da faculdade em que o aluno *tnome* = 'JOAQUIM SILVA' está matriculado em pelo menos uma turma?

.....
.....

ii. Qual é o nome das faculdades que não têm ainda qualquer matrícula registada?

.....
.....
.....

iii. Qual é o nome dos estudantes matriculados ao mesmo tempo nas turmas *tnome* = 'SD01' e *tnome* = 'SD02'?

.....
.....
.....

iv. Qual é o nome dos estudantes matriculados em todas as turmas que usam a sala 'S221'?

.....
.....
.....

v. Quantas faculdades têm mais de 20 turmas?

.....
.....
.....

vi. Para cada faculdade, qual é o número de estudantes matriculados?

.....
.....
.....

vii. Qual é o nome de cada estudante matriculado em turmas de mais de 1 faculdade?

.....
.....
.....

1/2

(d) Traduza para Álgebra Relacional a produção:

```

SELECT * FROM ESTUDANTES E
WHERE NOT EXISTS
(SELECT * FROM TURMA T WHERE T.SALA = 'S200' AND T.TNOME NOT IN
(SELECT M.TNOME FROM MATRICULAS M WHERE M.ENUM = E.ENUM))

```

.....

.....

.....

Exercício 2

2. É dado o seguinte programa em PLSQL:

```

drop table user1.tab1;
create table user1.tab1 as
select f.d1 as desp,f.c1 as cesp,f.d2 as dunid,
nvl(f.c2,0) as cunid,
to_char(f.dataq,'yyyymm') as mesq,
user1.x1(nvl(f.c2,0),f.c1,to_char(f.dataq,'yyyymm')) as tot,
user1.x2(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm')) as totr,
user1.x3(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),1) as q1,
user1.x3(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),2) as q2,
user1.x3(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),3) as q3,
user1.x3(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),4) as q4,
user1.x3(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),5) as q5,
user2.x4(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),1) as tq1,
user2.x4(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),2) as tq2,
user2.x4(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),3) as tq3,
user2.x4(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),4) as tq4,
user3.x4(nvl(f.c2,0), f.c1,to_char(f.dataq,'yyyymm'),5) as tq5
from user2.tab2 f
group by f.d1,f.c1,f.d2,
nvl(f.c2,0),
to_char(f.dataq,'yyyymm');

```

1

(a) Indique se as funções user2.x4 e user3.x4 são o mesmo objecto.

i. Resposta:

i. _____

ii. Porque:

ii. _____

i. _____

.....

.....

1

i. _____

.....

.....

2

.....

.....

.....

Exercício 3

3. É dado o seguinte programa em PLSQL:

```
create table encomenda
(encomendaaid int primary key,
fornecedor name varchar2(50) not null,
valortotal number(10 ,2) not null ,
constraint qmzero check(valortotal > 0)
);

create table encomendalinha
(encomendaaid int not null,
numlinha int not null check(lineno > 0),
dsc varchar2(50) not null,
quantidade int not null,
precunitario number(6 ,2) check ( precunitario > 0.0),
constraint ol pk primary key (encomendaaid,numlinha),
constraint olofk foreign key(encomendaaid) references encomenda(encomendaaid)
);

create or replace trigger settotal
after insert or update or delete on encomendalinha
for each row
declare
val number(10 ,2) := 0;
eid int;
begin
if inserting then
    val := :new.quantidade * :new.precunitario;
    eid := :new. encomendaaid ;
elsif updating then
    val := :new.quantidade * :new. precunitario -
           :old.quantidade * :old.precunitario;
    eid := :new.encomendaaid;
elsif deleting then
    val := 0 - :old.quantidade * :old.precunitario;
    eid := :old.encomendaaid;
end if;
execute updateencomenda(eid,val);
end;

create or replace procedure updateencomenda(eid int, val number(10,2)) is
begin
```

...
end;

Insert into encomenda ...
Insert into encomendalinha ...
...

2

- (a) Complete o procedimento *updateencomenda* de forma a actualizar o atributo *valortotal*.

.....
.....
.....

2½

- (b) Admita que num determinado momento a encomenda 335 tem 2 linhas e o seu *valortotal* é 100. São executados os comandos:

INSERT INTO encomendalinha
VALUES(335, 3, 'TESTE DEVOLUÇÃO DE LINHA',-5,20);
COMMIT;

O resultado dos comandos é um ERRO.

- i. Que tipo de erro é devolvido?

.....

- ii. O erro acontece no procedimento ou no trigger?

ii. _____

- iii. Dos dois comandos executados, qual é aquele que provoca o erro?

iii. _____

Exercício 4

 $2^{1/2}$

4. Mostre os benefícios da base de dados referida no exercício 1 ser uma base de dados distribuída. Mostre como os dados poderiam ser replicados e fragmentados.

[illegible]

Pergunta	1	2	3	4	Total
Pontos	8	5	4½	2½	20
Pontos Obtidos					