



Universidade de Coimbra
Departamento de Engenharia Informática

Bases de Dados

2023/2024

Exercícios Práticos para a Disciplina de
Bases de Dados

Versão 2.1

Marco Vieira

Índice

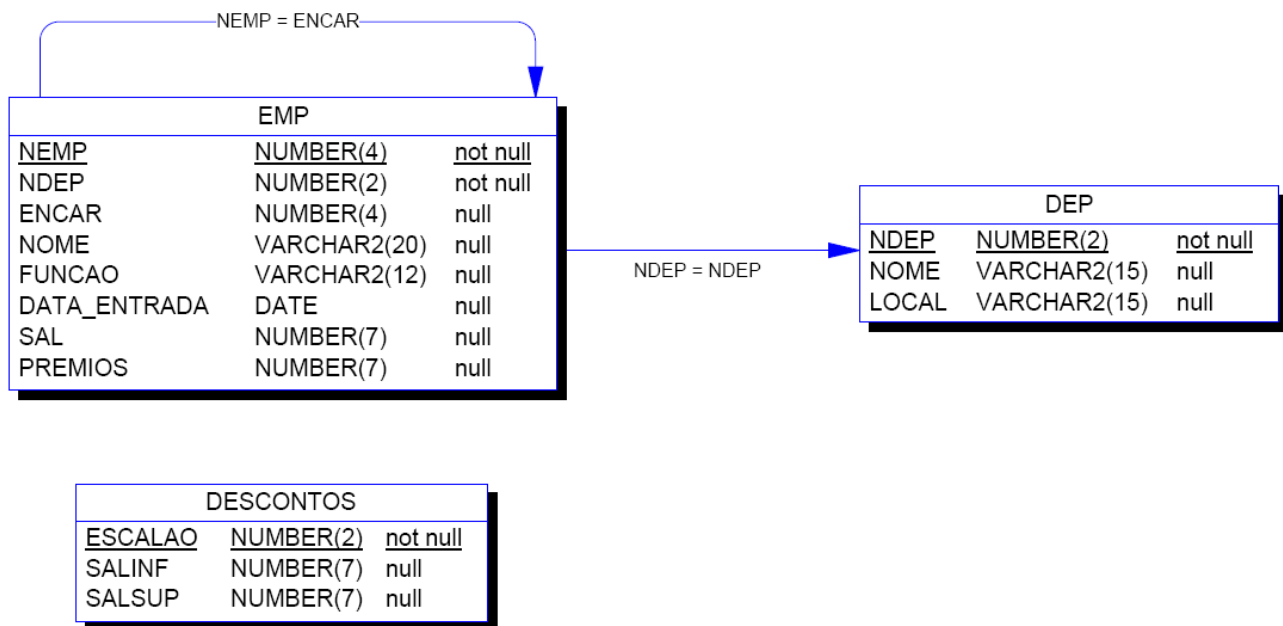
1. Linguagem SQL.....	3
2. Diagramas ER	22
3. Linguagem PL/SQL	30
4. Transacções e controlo de concorrência	37
5. Exercícios extra - SQL	41
6. Exercícios extra - Diagramas ER	51
7. Exercícios extra - PL/SQL	60

Este documento é um compêndio de exercícios práticos propostos para a disciplina de Bases de Dados, sendo incluídos quatro tópicos principais:

- 1. Linguagem SQL*
- 2. Diagramas Entidade-Relacionamento (ER)*
- 3. Linguagem PL/SQL*
- 4. Transacções*

1. Linguagem SQL

Assuma as seguintes tabelas, que pretendem modelar uma empresa com uma base de dados muito simples. Existe apenas informação sobre empregados, departamentos e escalões. Todos os empregados pertencem a um departamento. Essa relação é indicada através da coluna de chave forasteira, *ndep*, na tabela *emp*, que indica um valor existente na coluna de chave primária *ndep* da tabela *dep*. Cada empregado pertence também a um escalão de descontos de IRS. A relação entre empregados e descontos é calculada através do valor do salário do empregado. Assim, diz-se que um empregado, de número *nemp*, pertence a um escalão de número *escalao* se o seu valor de salário se situar entre os valores mínimo e máximo desse escalão; ou seja, apenas se e só se $descontos.salinf \leq emp.sal \leq descontos.salsup$.



Os comandos necessários para criar as tabelas e inserir dados de teste encontram-se na página da disciplina.

Baseando-se nas tabelas apresentadas acima, resolva os problemas:

1. Selecciona toda a informação da tabela DEP. O resultado deve ser similar ao que se segue:

NDEP	NOME	LOCAL
10	Contabilidade	Condeixa
20	Investigação	Mealhada
30	Vendas	Coimbra
40	Planeamento	Montemor

2. Mostre a lista de todos os empregados contendo o nome de cada empregado, a sua função, o salário e o número do departamento a que pertence.

NOME	FUNCAO	SAL	NDEP
Jorge Sampaio	Presidente	890000	10
Augusto Reis	Encarregado	450975	20
Duarte Guedes	Encarregado	380850	30
Silvia Teles	Encarregado	279450	10
Maria Dias	Analista	565000	20
Catarina Silva	Analista	435000	20
Joana Mendes	Vendedor	145600	30
Nelson Neves	Vendedor	212250	30
Ana Rodrigues	Vendedor	221250	30
Manuel Madeira	Vendedor	157800	30
Tome Ribeiro	Continuo	56950	30
Rita Pereira	Continuo	65100	20
Olga Costa	Continuo	68300	10
Antonio Silva	Continuo	70800	20

3. Apresente a lista de todos os departamentos ordenados decrescentemente por número de departamento.

NDEP	NOME	LOCAL
40	Planeamento	Montemor
30	Vendas	Coimbra
20	Investigação	Mealhada
10	Contabilidade	Condeixa

4. Mostre a lista de todas as funções existentes na empresa. Devem ser excluídas as repetições.

```

FUNCAO
-----
Analista
Continuo
Encarregado
Presidente
Vendedor

```

5. Apresente a lista de todos os empregados que recebem prémios, devendo a lista conter o nome do empregado, a sua função e o montante recebido em prémios.

NOME	FUNCAO	PREMIOS
Joana Mendes	Vendedor	56300
Nelson Neves	Vendedor	98500
Ana Rodrigues	Vendedor	81400
Manuel Madeira	Vendedor	0

6. Mostre a informação detalhada (toda a informação disponível) dos empregados dos que pertencem ao departamento 10 ou 30.

NEMP	NOME	FUNCAO	ENCAR	DATA_ENTR	SAL	PREMIOS	NDEP
1839	Jorge Sampaio	Presidente		11-FEB-84	890000		10
1698	Duarte Guedes	Encarregado	1839	25-NOV-91	380850		30
1782	Silvia Teles	Encarregado	1839	03-NOV-86	279450		10
1499	Joana Mendes	Vendedor	1698	04-OCT-84	145600	56300	30
1521	Nelson Neves	Vendedor	1698	27-FEB-83	212250	98500	30
1654	Ana Rodrigues	Vendedor	1698	17-DEC-90	221250	81400	30
1844	Manuel Madeira	Vendedor	1698	21-APR-85	157800	0	30
1900	Tome Ribeiro	Continuo	1698	05-MAR-94	56950		30
1934	Olga Costa	Continuo	1782	22-JUN-86	68300		10

7. Mostre os nomes de todos os analistas que trabalham no departamento 20.

NOME

 Maria Dias
 Catarina Silva

8. Mostre agora a lista de funcionários (nome e função) cujos nomes começam pela letra 'A' e contêm a sequência de letras 'us' e também os que começando por 'R' contêm a sequência de letras 'ei'. Tenha em atenção as diferenças de maiúsculas e minúsculas.

NOME	FUNCAO
Augusto Reis	Encarregado
Rita Pereira	Continuo

9. Mostre a lista da remuneração anual de todos os empregados contendo o nome do empregado, a função. A remuneração anual consiste em 14 vezes o salário mais o valor dos prémios, se existir.

Nome	Função	Remuneração Anual
Jorge Sampaio	Presidente	12460000
Augusto Reis	Encarregado	6313650
Duarte Guedes	Encarregado	5331900
Silvia Teles	Encarregado	3912300
Maria Dias	Analista	7910000
Catarina Silva	Analista	6090000
Joana Mendes	Vendedor	2094700
Nelson Neves	Vendedor	3070000
Ana Rodrigues	Vendedor	3178900
Manuel Madeira	Vendedor	2209200
Tome Ribeiro	Continuo	797300
Rita Pereira	Continuo	911400
Olga Costa	Continuo	956200
Antonio Silva	Continuo	991200

10. Apresente o nome de empregado, o salário, assim como o número e o nome do departamento de todos os empregados cujo nome começa por 'A' e o apelido por 'R'. Assuma que os nomes dos empregados são todos constituídos por apenas um nome próprio e um apelido, i.e., não têm nomes do meio.

NOME	SAL	NDEP	NOME_DEP
Augusto Reis	450975	20	Investigação
Ana Rodrigues	221250	30	Vendas

11. Apresente o nome, salário, nome do departamento e respectiva localização para todos os empregados cujo salário é inferior a 150000.

NOME	SAL	NOME	LOCAL
Joana Mendes	145600	Vendas	Coimbra
Tome Ribeiro	56950	Vendas	Coimbra
Rita Pereira	65100	Investigação	Mealhada
Olga Costa	68300	Contabilidade	Condeixa
Antonio Silva	70800	Investigação	Mealhada

12. Mostre o escalão de descontos (ou escalões) de cada função. O resultado deve ficar ordenado por escalão e dentro de cada escalão por função. Como pode existir mais do que uma pessoa por função dentro do mesmo escalão poderiam aparecer linhas repetidas. Garanta que NÃO aparecem linhas repetidas.

FUNCAO	ESCALAO
Continuo	1
Vendedor	2
Encarregado	3
Vendedor	3
Analista	4
Encarregado	4
Analista	5
Presidente	5

13. Mostre o nome, função e salário de todos os empregados de escalão salarial igual a 4, sendo o resultado ordenado por nome de empregado.

NOME	FUNCAO	SAL
Augusto Reis	Encarregado	450975
Catarina Silva	Analista	435000
Duarte Guedes	Encarregado	380850

14. A mesma informação que na questão anterior mas agora mostre também o nome do departamento de cada empregado.

NOME	FUNCAO	DEPARTAMENTO	SAL
Augusto Reis	Encarregado	Investigação	450975
Catarina Silva	Analista	Investigação	435000
Duarte Guedes	Encarregado	Vendas	380850

15. Mostre o nome, função, salário e local de trabalho de todos os empregados de 'Coimbra' e cujo salário é superior a 150000.

NOME	FUNCAO	SAL	LOCAL
-----	-----	-----	-----
Duarte Guedes	Encarregado	380850	Coimbra
Nelson Neves	Vendedor	212250	Coimbra
Ana Rodrigues	Vendedor	221250	Coimbra
Manuel Madeira	Vendedor	157800	Coimbra

16. Apresente o nome, função, escalão salarial e nome de departamento para todos os empregados com excepção dos empregados cuja função é 'Continuo'. O resultado deve ficar ordenado por ordem decrescente de escalão salarial e dentro de cada escalão pelo nome do empregado.

NOME	FUNCAO	ESCALAO	DEPARTAMENTO
-----	-----	-----	-----
Jorge Sampaio	Presidente	5	Contabilidade
Maria Dias	Analista	5	Investigação
Augusto Reis	Encarregado	4	Investigação
Catarina Silva	Analista	4	Investigação
Duarte Guedes	Encarregado	4	Vendas
Ana Rodrigues	Vendedor	3	Vendas
Nelson Neves	Vendedor	3	Vendas
Silvia Teles	Encarregado	3	Contabilidade
Joana Mendes	Vendedor	2	Vendas
Manuel Madeira	Vendedor	2	Vendas

17. Produza o seguinte resultado que consiste em todos os nomes e números dos departamentos e em todos os nomes e números dos empregados. Note que foram acrescentadas linhas antes da sequência de departamentos e antes da sequência de empregados para aumentar a visibilidade. Note ainda que nas linhas acrescentadas foi inserido um número na segunda coluna para numa ordenação sobre essa coluna o resultado aparecer como desejado.

Sugestão: Faça uso do operador UNION. Utilize constantes para produzir as quatro linhas acrescentadas.

NOMES	NUMEROS
-----	-----
	0
DEPARTAMENTOS:	1
Contabilidade	10
Investigação	20
Vendas	30
Planeamento	40
	999
EMPREGADOS:	1000
Antonio Silva	1369
Joana Mendes	1499
Nelson Neves	1521
Augusto Reis	1566
Ana Rodrigues	1654
Duarte Guedes	1698
Silvia Teles	1782
Maria Dias	1788
Jorge Sampaio	1839
Manuel Madeira	1844

Rita Pereira	1876
Tome Ribeiro	1900
Catarina Silva	1902
Olga Costa	1934

18. Mostre o departamento que não tem empregados usando o operador MINUS.

NDEP	NOME	LOCAL
40	Planeamento	Montemor

19. Mostre o nome dos empregados Vendedores que trabalham em Coimbra e têm escalão salarial maior ou igual a 2.

20. Escreva um comando que devolva o número de empregados. O resultado deve ser semelhante ao que se segue.

```
Total de empregados
-----
14
```

21. Escreva um comando que conte o número de empregados existentes, calcule o salário médio mensal e o total de remuneração anual auferido pelo conjunto de todos os empregados. A remuneração de cada empregado deve ser calculada da seguinte forma: *salário * 14 + prémios*.

Total de Empregados	Salário médio mensal	Remuneração total anual
14	285666.071	56226750

22. Mostre a lista dos Contínuos e o respectivo salário com um aumento de 13,55% para estes empregados. O salário depois do aumento deverá ser arredondado na primeira casa decimal.

NOME	FUNCAO	SAL	SALARIO_COM_AUMENTO
Tome Ribeiro	Contínuo	56950	64666.7
Rita Pereira	Contínuo	65100	73921.1
Olga Costa	Contínuo	68300	77554.7
Antonio Silva	Contínuo	70800	80393.4

23. Escreva um comando que produza o seguinte resultado. Note que antes de cada nome de empregado existem 4 sinais de '>' (maior que) e um espaço, e que depois de cada nome existe um espaço e caracteres '<' (menor que) suficientes até se atingir um comprimento total de 25 caracteres.

```
Mariquices com strings
-----
>>>> Ana Rodrigues <<<<<<
>>>> Antonio Silva <<<<<<
>>>> Augusto Reis <<<<<<
>>>> Catarina Silva <<<<<
>>>> Duarte Guedes <<<<<<
>>>> Joana Mendes <<<<<<<
>>>> Jorge Sampaio <<<<<<
>>>> Manuel Madeira <<<<<
>>>> Maria Dias <<<<<<<<<
>>>> Nelson Neves <<<<<<<
>>>> Olga Costa <<<<<<<<<
>>>> Rita Pereira <<<<<<<<<
>>>> Silvia Teles <<<<<<<<
>>>> Tome Ribeiro <<<<<<<
```

24. Encontre o salário mais baixo, mais alto e o salário médio de todos os funcionários:

SALARIO_MAIS_BAIKO	SALARIO_MAIS_ALTO	SALARIO_MEDIO
56950	890000	285666.071

25. Mostre quantos empregados existem para cada função. Considere apenas as funções que terminam com a letra 'o' ou para os 'Analistas'.

FUNCAO	QUANTIDADE
Analista	2
Continuo	4
Encarregado	3

26. Mostre o salário médio para cada tipo de função, ordenando os resultados por ordem crescente dos salários médios. O salário médio deverá ser arredondado para o menor inteiro possível que seja superior ou igual ao seu valor. Apenas deverão ser mostradas funções onde exista mais de 1 trabalhador.

Funcao	Salário Médio
Continuo	65288
Vendedor	184225
Encarregado	370425
Analista	500000

27. Mostre o número de empregados para cada escalão de descontos. Só devem ser considerados os empregados que trabalhem em departamento localizados em Coimbra.

ESCALAO	NUM_EMPREGADOS
1	1
2	2
3	2
4	1

28. Mostre o número de empregados, o salário máximo e o salário mínimo dos empregados que trabalham em cada departamento. Só devem ser considerados os departamentos cuja média dos salários dos empregados seja superior a 200000.

DEPARTAMENTO	NUM_EMPREGADOS	SAL_MAX	SAL_MIN
Contabilidade	3	890000	68300
Investigação	5	565000	65100

29. Mostre quantos departamentos têm empregados nos escalões de descontos 1, 2 e 3.

ESCALAO	NUM_DEPART
1	3
2	1
3	2

30. Apresente a lista de todos os empregados (nome, número de departamento e salário) cujo salário está entre 150000 e 300000.

NOME	NDEP	SAL
-----	-----	-----
Silvia Teles	10	279450
Nelson Neves	30	212250
Ana Rodrigues	30	221250
Manuel Madeira	30	157800

31. Apresente a lista de funcionários (nome e função) cujos nomes aparecem as letras 'v' ou 'u'.

NOME	FUNCAO
-----	-----
Augusto Reis	Encarregado
Duarte Guedes	Encarregado
Silvia Teles	Encarregado
Catarina Silva	Analista
Nelson Neves	Vendedor
Ana Rodrigues	Vendedor
Manuel Madeira	Vendedor
Antonio Silva	Continuo

32. Mostre a lista de vendedores cujos prémios foram menores do que 10% da remuneração anual ($sal * 14 + prémios$). O resultado deve incluir o nome do vendedor, 10% da sua remuneração anual e ainda os prémios, e deve ser ordenado crescentemente pelos 10% de remuneração anual. No caso de haver vários vendedores com a mesma remuneração anual, estes devem surgir ordenados pelo nome do vendedor.

NOME	10% Sal. Anual	PREMIOS
-----	-----	-----
Joana Mendes	209470	56300
Manuel Madeira	220920	0
Nelson Neves	307000	98500
Ana Rodrigues	317890	81400

33. Mostre os nomes dos empregados, a sua função e o nome do departamento em que cada empregado trabalha. O resultado deve estar ordenado pelo nome de departamento e dentro de cada departamento pelo nome do empregado. O resultado deve ser semelhante ao que se segue:

NOME	FUNCAO	NOME_DEP
-----	-----	-----
Jorge Sampaio	Presidente	Contabilidade
Olga Costa	Continuo	Contabilidade
Silvia Teles	Encarregado	Contabilidade
Antonio Silva	Continuo	Investigação
Augusto Reis	Encarregado	Investigação
Catarina Silva	Analista	Investigação
Maria Dias	Analista	Investigação
Rita Pereira	Continuo	Investigação
Ana Rodrigues	Vendedor	Vendas
Duarte Guedes	Encarregado	Vendas
Joana Mendes	Vendedor	Vendas
Manuel Madeira	Vendedor	Vendas
Nelson Neves	Vendedor	Vendas
Tome Ribeiro	Continuo	Vendas

34. Faça novamente o exercício anterior mas apresente também o nome dos departamentos onde não exista nenhum empregado a trabalhar.

EMPREGADO	FUNCAO	DEPARTAMENTO
-----	-----	-----
Jorge Sampaio	Presidente	Contabilidade
Olga Costa	Continuo	Contabilidade
Silvia Teles	Encarregado	Contabilidade
Antonio Silva	Continuo	Investigação
Augusto Reis	Encarregado	Investigação
Catarina Silva	Analista	Investigação
Maria Dias	Analista	Investigação
Rita Pereira	Continuo	Investigação
		Planeamento
Ana Rodrigues	Vendedor	Vendas
Duarte Guedes	Encarregado	Vendas
Joana Mendes	Vendedor	Vendas
Manuel Madeira	Vendedor	Vendas
Nelson Neves	Vendedor	Vendas
Tome Ribeiro	Continuo	Vendas

35. Mostre o nome de cada empregado e do encarregado correspondente, o escalão de salário e a localidade do departamento em que trabalha o empregado. Deve mostrar apenas aqueles que têm um salário superior a 150000.

NOME	ENCARREGADO	ESCALAO	LOCALIDADE
-----	-----	-----	-----
Manuel Madeira	Duarte Guedes	2	Coimbra
Nelson Neves	Duarte Guedes	3	Coimbra
Ana Rodrigues	Duarte Guedes	3	Coimbra
Silvia Teles	Jorge Sampaio	3	Condeixa
Duarte Guedes	Jorge Sampaio	4	Coimbra
Catarina Silva	Augusto Reis	4	Mealhada
Augusto Reis	Jorge Sampaio	4	Mealhada
Maria Dias	Augusto Reis	5	Mealhada

36. Mostre o nome, função e departamento dos vendedores e apenas deles. O nome deverá aparecer todo em maiúsculas, a função em minúsculas e o departamento com a primeira letra em maiúscula e o resto em minúsculas. Ordene o resultado por nome do funcionário.

NOME	FUNCAO	DEPARTAMENTO
-----	-----	-----
ANA RODRIGUES	vendedor	Vendas
JOANA MENDES	vendedor	Vendas
MANUEL MADEIRA	vendedor	Vendas
NELSON NEVES	vendedor	Vendas

37. Mostre uma lista dos encarregados e seus subordinados. Ordene os resultados por nome de encarregado e depois por nome de empregado.

ENCARREGADO	EMPREGADO
Augusto Reis	Catarina Silva
Augusto Reis	Maria Dias
Catarina Silva	Antonio Silva
Duarte Guedes	Ana Rodrigues
Duarte Guedes	Joana Mendes
Duarte Guedes	Manuel Madeira
Duarte Guedes	Nelson Neves
Duarte Guedes	Tome Ribeiro
Jorge Sampaio	Augusto Reis
Jorge Sampaio	Duarte Guedes
Jorge Sampaio	Silvia Teles
Maria Dias	Rita Pereira
Silvia Teles	Olga Costa

38. Escreva um comando que determine quantos empregados **não** ganham prémios. O resultado deve ser semelhante ao que se segue.

```
Empregados sem premio
-----
10
```

39. Encontre a diferença entre o salário mais alto e o mais baixo para cada departamento.

DEPARTAMENTO	DIFERENCA
10	821700
20	499900
30	323900

40. Mostre o salário mais baixo dos empregados que trabalham para cada Encarregado. Exclua os grupos em que o salário mínimo seja inferior a 200000 e ordene o resultado por salário.

Encarregado	Salário Mínimo
1839	279450
1566	435000
	890000

41. Indique o salário máximo, mínimo e médio e a quantidade de empregados para cada função e de cada departamento (grupos e subgrupos). Exclua registos individuais de nome 'Jorge Sampaio' e resultados colectivos que apresentem um salário máximo inferior ou igual a 200000. Ordene por departamento e depois por salário máximo.

NDEP	FUNCAO	SAL_MAX	SAL_MIN	SAL_MED	QUANTOS
10	Encarregado	279450	279450	279450	1
20	Encarregado	450975	450975	450975	1
20	Analista	565000	435000	500000	2
30	Vendedor	221250	145600	184225	4
30	Encarregado	380850	380850	380850	1

42. Mostre o número de empregados e a soma dos salários dos empregados que trabalham para cada Encarregado. Só devem ser considerados os empregados que trabalham em Coimbra ou Condeixa. Exclua os encarregados com um ou nenhum empregado.

NOME	LOCAL	NUM_EMP	SOMA_SAL
Duarte Guedes	Coimbra	5	793850

43. Encontre o salário mais baixo de todos.

SALARIO_MINIMO
56950

44. Encontre todos os empregados que têm a mesma função da 'Olga Costa' e mostre também os seus ordenados.

NOME	FUNCAO	SAL
Tome Ribeiro	Continuo	56950
Rita Pereira	Continuo	65100
Olga Costa	Continuo	68300
Antonio Silva	Continuo	70800

45. Encontre os empregados que ganham o maior salário em cada departamento (utilize o operador IN).

NOME	SAL	NDEP
Jorge Sampaio	890000	10
Maria Dias	565000	20
Duarte Guedes	380850	30

46. Encontre os empregados que ganham mais do que o salário mais baixo do departamento 30. Não use a função MIN. Ordene os salários mostrados por ordem decrescente (utilize o operador ANY).

NOME	SAL	FUNCAO	NDEP
Jorge Sampaio	890000	Presidente	10
Maria Dias	565000	Analista	20
Augusto Reis	450975	Encarregado	20
Catarina Silva	435000	Analista	20
Duarte Guedes	380850	Encarregado	30
Silvia Teles	279450	Encarregado	10
Ana Rodrigues	221250	Vendedor	30
Nelson Neves	212250	Vendedor	30
Manuel Madeira	157800	Vendedor	30
Joana Mendes	145600	Vendedor	30
Antonio Silva	70800	Continuo	20
Olga Costa	68300	Continuo	10
Rita Pereira	65100	Continuo	20

47. Encontre os empregados que ganham mais do que o salário máximo do departamento cujo nome é 'Vendas'. Não use a função MAX. Ordene os salários mostrados por ordem decrescente (utilize o operador ALL). Ordene o resultado pelo salário.

NOME	SAL	FUNCAO	NDEP
Catarina Silva	435000	Analista	20
Augusto Reis	450975	Encarregado	20
Maria Dias	565000	Analista	20
Jorge Sampaio	890000	Presidente	10

48. Seleccione os empregados cujo salário seja superior ao salário médio dos empregados que trabalham no departamento de vendas. Ordene o resultado descendentemente pelo nome do empregado.

NOME_EMPREGADO	SAL	FUNCAO	DEPARTAMENTO
Silvia Teles	279450	Encarregado	Contabilidade
Nelson Neves	212250	Vendedor	Vendas
Maria Dias	565000	Analista	Investigação
Jorge Sampaio	890000	Presidente	Contabilidade
Duarte Guedes	380850	Encarregado	Vendas
Catarina Silva	435000	Analista	Investigação
Augusto Reis	450975	Encarregado	Investigação
Ana Rodrigues	221250	Vendedor	Vendas

49. Obtenha o dia da semana correspondente a 1 de Janeiro de 2005.

Dia da Semana

SATURDAY

50. Seleccione os empregados contratados depois de 1 de Janeiro de 1993.

NOME	DATA_ENTRADA
Catarina Silva	13-04-1993
Tome Ribeiro	05-03-1994
Rita Pereira	07-02-1996
Antonio Silva	22-12-1996

51. Execute o seguinte comando:

```
CREATE TABLE emp2
(nemp NUMBER(4) CONSTRAINT pk_emp PRIMARY KEY,
 nome VARCHAR2(20) CONSTRAINT nn_nome NOT NULL
    CONSTRAINT upper_nome CHECK (nome = UPPER(nome)),
 funcao VARCHAR2(12),
 encar NUMBER CONSTRAINT fk_encar REFERENCES emp2(nemp),
 data_entrada DATE DEFAULT SYSDATE,
 sal NUMBER(7) CONSTRAINT ck_sal CHECK (sal > 70000),
 premios NUMBER(7) DEFAULT NULL,
 ndep NUMBER(2) CONSTRAINT nn_ndep NOT NULL
    CONSTRAINT fk_ndep REFERENCES dep(ndep)
);
```

O comando cria uma nova tabela, EMP2 muito semelhante a EMP. Se a tabela já existir apague-a com 'drop table emp2' e execute novamente o comando de criação da tabela. Repare que a tabela

EMP2 é quase uma cópia da EMP. As principais diferenças são algumas restrições nas colunas NOME e SAL. Insira em EMP2 o empregado com as seguintes características:

```
nemp = 5555,
nome = 'CHICO FININHO',
funcao = 'Cantor',
encar = null,
data_entrada = hoje,
sal = 100000,
premios = null
ndep = 40
```

52. Insira em EMP2 todos os empregados de EMP excepto os de números 1902 e 1369. Note que ao inserir os dados terá que ter algum cuidado a formatar as colunas de NOME e SAL. Deverá converter o nome para maiúsculas e o sal para pelo menos 70001. Depois de inseridos os dados, a instrução *'select nemp, nome, encar, sal, premios, ndep from emp2 order by ndep'* devolve o seguinte resultado:

NEMP	NOME	ENCAR	SAL	PREMIOS	NDEP
1839	JORGE SAMPAIO		890000		10
1782	SILVIA TELES	1839	279450		10
1934	OLGA COSTA	1782	70001		10
1566	AUGUSTO REIS	1839	450975		20
1876	RITA PEREIRA	1788	70001		20
1788	MARIA DIAS	1566	565000		20
1698	DUARTE GUEDES	1839	380850		30
1844	MANUEL MADEIRA	1698	157800	0	30
1900	TOME RIBEIRO	1698	70001		30
1654	ANA RODRIGUES	1698	221250	81400	30
1499	JOANA MENDES	1698	145600	56300	30
1521	NELSON NEVES	1698	212250	98500	30
5555	CHICO FININHO		100000		40

Qual deverá ser o comando INSERT? (Note que o empregado 'CHICO FININHO' resulta do comando anterior e não deste.)

53. Apague todos os empregados do departamento 20. Após o comando, a tabela EMP2 deverá conter os valores (nem todas as colunas são mostradas):

NEMP	NOME	ENCAR	SAL	PREMIOS	NDEP
1839	JORGE SAMPAIO		890000		10
1782	SILVIA TELES	1839	279450		10
1934	OLGA COSTA	1782	70001		10
1698	DUARTE GUEDES	1839	380850		30
1521	NELSON NEVES	1698	212250	98500	30
1844	MANUEL MADEIRA	1698	157800	0	30
1900	TOME RIBEIRO	1698	70001		30
1654	ANA RODRIGUES	1698	221250	81400	30
1499	JOANA MENDES	1698	145600	56300	30
5555	CHICO FININHO		100000		40

54. Apague de EMP2 o empregado de número 1654.

55. Apague todos os elementos de EMP2. Insira cópias dos empregados de EMP em EMP2 com o cuidado de alterar o nome para maiúsculas e garantir que o salário cumpre as restrições. Altere os elementos dos empregados da seguinte maneira: mude os empregados do departamento 10 para o 20, os do 20 para o 30 e os do 30 para o 40. Altere ainda os seus salários para 110% da média de empregados do seu departamento e altere os prémios para 150% da média dos prémios do seu departamento (subconsulta correlacionada). Altere apenas os empregados que não sejam os que ganhem mais de cada departamento. Execute todas as alterações num único comando UPDATE.

Após as alterações, a tabela emp2 deverá conter os dados:

NEMP	NOME	ENCAR	SAL	PREMIOS	NDEP
1839	JORGE SAMPAIO		890000		10
1782	SILVIA TELES	1839	454465		20
1934	OLGA COSTA	1782	454465		20
1788	MARIA DIAS	1566	565000		20
1566	AUGUSTO REIS	1839	350191		30
1698	DUARTE GUEDES	1839	380850		30
1902	CATARINA SILVA	1566	350191		30
1369	ANTONIO SILVA	1902	350191		30
1876	RITA PEREIRA	1788	350191		30
1499	JOANA MENDES	1698	217754	88575	40
1521	NELSON NEVES	1698	217754	88575	40
1654	ANA RODRIGUES	1698	217754	88575	40
1844	MANUEL MADEIRA	1698	217754	88575	40
1900	TOME RIBEIRO	1698	217754	88575	40

56. Por fim apague a tabela EMP2 com comando *'drop table emp2'*.

57. Cria uma tabela, projectos, com as colunas, NPROJ (number(7)), NOME (varchar2(20)), e DESCRICAO (varchar2(60)). A chave primária deve ser NPROJ. A coluna NOME deve ser NOT NULL e UNIQUE. A coluna DESCRICAO deve ser NOT NULL.

58. Cria uma sequência, SEQ_NPROJ para usar nos valores de NPROJ. A sequência deve começar em 500 e avançar de 10 em 10. Não deverá voltar ao princípio quando terminar e não deve estar limitada a nenhum valor máximo.

59. Insira vários registos na tabela projectos (usando a sequência SEQ_NPROJ). Insira por exemplo os projectos:

Nome	Descrição
'Sificap'	'BD para controlo das actividades das pescas'
'IPQ'	'Instituto Português da Qualidade'
'Apolo11'	'This is a small step for man, but a giant leap for Mankind'

60. Faça uma cópia da tabela EMP para uma nova tabela EMP3. Use o comando CREATE TABLE com subconsulta. Na cópia de valores transforme os nomes dos empregados para maiúsculas e garanta que todos os empregados ganhem mais do que 70000. Relembre o uso das funções UPPER e GREATEST (se não se lembrar veja no manual). Note que como está a produzir alterações sobre algumas colunas irá ter que usar pseudónimos para a tabela EMP3 possuir colunas com nomes válidos.

61. Como ao criar a nova tabela EMP3 as restrições não são copiadas altere a tabela de modo a incluir as restrições que existiam na tabela EMP. Use o comando ALTER TABLE.

62. Cria uma sequência EMP3_NUMBER que comece em 1 e aumente de 10 em 10 e que depois de chegar ao fim não recomece do princípio.

63. Insira em EMP3 os empregados com as seguintes características:

```
nome = 'CHICO FININHO',  
funcao = 'Cantor',  
encar = null,  
data_entrada = hoje, (use sysdate)  
sal = 100000,  
premios = null  
ndep = 40.
```

```
nome = 'CHICO FINÃO',  
funcao = 'Cantor Pop',  
encar = null,  
data_entrada = hoje, (use sysdate)  
sal = 200000,  
premios = 100  
ndep = 30
```

```
nome = 'CHICO FINISSIMO',  
funcao = 'Cantor Rock',  
encar = null,  
data_entrada = hoje, (use sysdate)  
sal = 400000,  
premios = 200  
ndep = 10.
```

Repare como os números de empregado foram introduzidos.

64. Acrescente uma restrição a EMP3 que garanta que o valor do prémio é sempre inferior ao valor do sal para cada empregado. Teste se a restrição está a funcionar.

65. Acrescente uma coluna, NPROJ, do tipo NUMBER(7) a EMP3.

66. Garanta que NPROJ é NOT NULL. Acha que é possível fazer isso? Porquê? Ou como?

67. Garanta então que NPROJ (em EMP3) é NOT NULL mas mantenha a restrição no modo DISABLE. Atribua o nome NN_NPROJ_EMP3 à restrição.

68. Modifique a coluna NPROJ da tabela EMP3 de modo a que passa a ser uma chave forasteira que se refira à coluna do mesmo nome na tabela projectos. Repare que pode declarar a coluna NPROJ como sendo uma chave forasteira mesmo com esta a conter valores nulos.

69. Altere os valores de NPROJ na tabela EMP3 de modo a colocar os empregados a trabalhar em projectos.

70. Active (ENABLE) a restrição NN_NPROJ_EMP3 que tinha sido criada no exercício 25 mas que estava desactivada (DISABLE).

71. Crie um comando de criação de uma tabela EMP4 para que esta tenha o conteúdo de EMP mais uma coluna com o número de empregados no departamento correspondente ao empregado. Use neste comando uma sub-consulta da cláusula FROM.

72. Encontre o empregado que ganha o salário mais baixo de todos (utilize subconsulta).

NOME	FUNCAO	SAL
Tome Ribeiro	Continuo	56950

73. Mostre os departamentos que têm um salário médio superior ao salário médio do departamento 30.

NOME	SAL_MED
Contabilidade	412583.333
Investigação	317375

74. Mostre o departamento que não tem empregados (não use o operador MINUS).

NDEP	NOME	LOCAL
40	Planeamento	Montemor

75. Obtenha o seguinte *output*:

DEPARTAMENTO	NUM_EMPREGADOS
Contabilidade	3
Investigação	5
Planeamento	Nenhum
Vendas	6

76. Selecciona o número e o nome dos empregados que são encarregados de mais de dois outros empregados.

NEMP	NOME
1839	Jorge Sampaio
1698	Duarte Guedes

77. Fazer o mesmo mas a mostrar quantos empregados são geridos por cada encarregado. (subconsulta no FROM).

NEMP	NOME	QUANTOS
1839	Jorge Sampaio	3
1698	Duarte Guedes	5

78. Execute o seguinte comando:

```
CREATE TABLE emp2
(nemp NUMBER(4) CONSTRAINT pk_emp PRIMARY KEY,
 nome VARCHAR2(20) CONSTRAINT nn_nome NOT NULL
   CONSTRAINT upper_nome CHECK (nome = UPPER(nome)),
 funcao VARCHAR2(12),
 encar NUMBER CONSTRAINT fk_encar REFERENCES emp2(nemp),
 data_entrada DATE DEFAULT SYSDATE,
 sal NUMBER(7) CONSTRAINT ck_sal CHECK (sal > 70000),
 premios NUMBER(7) DEFAULT NULL,
 ndep NUMBER(2) CONSTRAINT nn_ndep NOT NULL
   CONSTRAINT fk_ndep REFERENCES dep(ndep)
);
```

O comando cria uma nova tabela, EMP2 muito semelhante a EMP. Se a tabela já existir apague-a com *'drop table emp2'* e execute novamente o comando de criação da tabela. Repare que a tabela EMP2 é quase uma cópia da EMP. As principais diferenças são algumas restrições nas colunas nome e sal. Execute o seguinte comando para inserir registos em EMP2:

```
insert into emp2
select nemp, upper(nome) nome, funcao, encar, data_entrada, GREATEST(sal, 70001)
sal, premios, ndep
from emp;
```

Insira em EMP2 o empregado com as seguintes características:

```
nemp = 5555,
nome = 'CHICO FININHO',
funcao = 'Cantor',
encar = null,
data_entrada = hoje,
sal = 100000,
premios = null
ndep = 40
```

- 79.** Apague todos os empregados do departamento 10. Tente perceber porque é que obtém um código de erro.
- 80.** Apague de EMP2 os empregados que recebam menos que a média de salários dos empregados do seu departamento (use subconsulta correlacionada).
- 81.** Apague a tabela EMP2 com comando *'drop table emp2'*.
- 82.** Faça uma cópia da tabela EMP para uma nova tabela EMP3.
- 83.** Crie um atributo novo em EMP3 de nome ULTIMO_NOME_REPETIDO tipo CHAR(1).
- 84.** Use o comando UPDATE para que todos os empregados com repetição do último nome tenham o valor deste atributo a 'S'.

2. Diagramas ER

1. Desenhe o diagrama ER correcto e o mais completo possível para cada uma das seguintes situações muito simples (inclua todos os atributos que consiga imaginar para cada caso, assim como a correcta definição de graus e de participações):
 - a) Numa empresa de transportes cada motorista conduz um veículo. O mesmo veículo pode ser conduzido por vários motoristas (por exemplo, em regime de turnos).
 - b) Num hospital todos os doentes têm de ter médico assistente. Nos casos mais graves podem mesmo ter mais do que um médico. Alguns médicos estão dispensados de exercer medicina por participarem na gestão do hospital.
 - c) Numa empresa de venda por catálogo cada armazém tem um gestor e cada gestor só gere um armazém. Cada armazém contém muitas peças, mas a mesma peça pode existir em vários armazéns.
 - d) Uma cadeia de lojas pretende armazenar informação sobre cada loja individual e sobre os fornecedores a que cada loja compra produtos. Cada loja pode comprar produtos a vários fornecedores e cada fornecedor pode fornecer diversas lojas.
 - e) Uma base de dados de uma oficina contém informação sobre mecânicos e sobre automóveis. Cada mecânico pode trabalhar em vários carros mas cada carro é atribuído apenas a um mecânico.
 - f) Uma empresa de pintura de construção civil pretende ter uma base de dados com os pintores existentes no mercado filiados no Sindicato dos Pintores de Construção Civil da Região Centro e sobre as casas que estão actualmente a ser pintadas pela empresa. Cada pintor pinta apenas uma casa de cada vez mas vários pintores podem ser designados para pintar a mesma casa. Interessa considerar as casas que estão a ser pintadas e também as que já foram pintadas no passado e quando foram as pinturas.
 - g) O serviço de prospecção de novos assinantes de uma empresa de telecomunicações pretende criar uma base de dados com endereços do concelho de Coimbra e os respectivos números de telefone (pode ser mais do que um por endereço), de modo a poder determinar as zonas de maior ou menor cobertura da rede telefónica.
 - h) Um empregado pode efectuar limpeza a mais do que um imóvel e um imóvel não tem um empregado de limpeza único.
 - i) Uma empresa de construção civil, mediante um prévio planeamento das duas actividades, necessita de programar correctamente o seu stock de materiais de construção para as várias obras que realiza. Para cada obra é efectuado um registo das suas necessidades ao longo do tempo de decurso da obra, em que é registado qual a quantidade de material necessário em cada data.
 - j) O instituto XPTO responsável pela gestão das pescas pretende construir uma base de dados que permita guardar a informação do peixe descarregado pelas embarcações nas lotas do País. Cada embarcação pode pertencer a um armador e um armador pode possuir no máximo uma embarcação. Cada embarcação pode efectuar várias descargas. Cada descarga é efectuada por uma embarcação e pode contemplar várias espécies de peixe, pretendo-se guardar a quantidade de cada uma das espécies. Uma descarga é sempre efectuada numa lota, que pertence a uma capitania. Cada capitania possui no máximo uma lota. Pretende-se para guardar para espécie qual a quantidade máxima de peixe que cada armador pode pescar.

2. Um **Concessionário de Veículos Automóveis** pretende informatizar os seus serviços para o que necessita de uma aplicação de base de dados cuja concepção deve ter em conta os seguintes pontos:

- O concessionário tem dois tipos de clientes: clientes individuais e empresas. Todos os clientes têm um código unívoco, mas os clientes individuais têm atributos próprios de uma pessoa (nome, estado civil, sexo, morada, telefone, etc.), enquanto os clientes do tipo empresa são descritos pelos atributos adequados para descrever uma empresa (nome, ramo de negócio, morada, telefone, fax, etc.).
- O concessionário faz três tipos de negócios com os seus clientes: vende automóveis, compra automóveis (quando um cliente vende o seu carro usado para o concessionário o voltar a vender a outro cliente) e faz contratos de leasing. Em qualquer dos três casos, é sempre celebrado um contrato com o cliente (contrato de venda, compra ou leasing). Há aspectos da informação contida nesses contratos que depende do tipo de contrato (e.g., nos contratos de venda e de leasing é necessário indicar o prazo da garantia, o que já não acontece quando é o concessionário a comprar o carro; nos contratos de leasing há taxas de juro e apólices obrigatórias o que não acontece nos outros contratos, etc.).
- O concessionário pretende manter informação sobre todos os carros que tem para venda, quer os novos quer os usados. Depois de um carro ser vendido, a informação relativa a esse carro continua ainda a ser importante para o concessionário, quer para fins estatísticos, quer ainda porque a experiência mostra que, muitas vezes, esse carro acaba por voltar ao concessionário (e.g., por uma retoma).

Apresente um diagrama ER para a base do concessionário e o esquema físico correspondente. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada.

3. A **Escola de Condução Chico Esperto** pretende adquirir uma base de dados para ajudar a gestão das aulas ministradas aos seus alunos e cuja concepção deve ter em conta os seguintes aspectos:

- Há dois tipos de aulas: aulas teóricas e aulas de condução. As primeiras são dadas para grupos de alunos e são de participação livre (isto é, não é obrigatória a frequência das aulas teóricas para os alunos poderem fazer o exame de código) enquanto que as aulas de condução são dadas a um único aluno de cada vez e é obrigatório ter pelo menos vinte lições para o aluno poder fazer exame de condução;
- Todos os dias úteis são dadas quatro aulas teóricas, sendo cada aula dada apenas por um único instrutor. Qualquer instrutor da escola pode dar aulas de condução ou aulas práticas. Em todas as aulas teóricas são registados os alunos que foram à aula, o instrutor de deu a aula, a hora em que a aula decorreu e ainda um pequeno sumário com a matéria dada. Para se ajustarem ao regime de participação livre, e poderem ser frequentadas a qualquer momento pelos alunos, todas as aulas teóricas são auto-contidas, não havendo sequência de matérias de aula para aula;
- Cada aluno tem um único instrutor para as aulas de condução, só tendo aulas de condução com esse instrutor. Cada instrutor conduz sempre o mesmo veículo, mas um dado veículo pode ser utilizado por mais do que um instrutor (em horários diferentes) devido ao regime de turnos;

- As aulas de condução duram sempre uma hora, começando sempre a horas certas (às 9 horas, às 10 horas, etc.). O mesmo aluno pode ter mais do que uma aula no mesmo dia;
- A escola prepara alunos para cartas de ligeiros e de pesados, tendo por isso na sua frota veículos dos dois tipos. Claro está que a base de dados contém toda a informação considerada necessária sobre os veículos da escola;
- Um aluno só tem aulas de condução para o tipo de carta que está a tirar, isto é, carta de ligeiros ou carta de pesados;
- A base de dados deve registar informação pessoal típica (nome, BI, morada, telefone, data de nascimento, etc.) quer para os alunos quer para os funcionários da escola. Contudo, funcionários têm mais alguns atributos específicos que importa registar, tais como o salário, data de entrada ao serviço, função, etc.;
- Pretende-se ainda registar todos os exames (de código e de condução) efectuados pelos alunos da escola, sendo relevante reter para cada tipo de exame a data do exame, o aluno que o efectuou e o resultado (aprovado ou reprovado).

Apresente um esquema para a base de dados da escola Chico Esperto utilizando o método da Entidade-Relacionamento. Pode introduzir atributos ou novos elementos que completem a descrição do problema acima efectuada se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados (mas muito sucintamente). A sua resposta deve indicar todas as tabelas, os atributos de cada tabela, as chaves primárias e as chaves forasteiras (mas não é necessário mostrar os comandos para criação das tabelas). A preservação da integridade da base de dados é muito importante, pelo que deve esclarecer todas as situações em que a estrutura da sua base de dados não permite garantir a integridade dos dados.

4. A **Comissão Nacional de Eleições** pretende ter uma base de dados para gerir as próximas eleições em que os eleitores já vão votar por um processo electrónico. O cartão de eleitor passará a ser um cartão magnético (do tipo do Multibanco), que o identificará univocamente, podendo os eleitores votar nos terminais Multibanco (usando o cartão de eleitor) ou pela Internet. Para votar o utilizador terá apenas de ir ao Multibanco mais próximo no dia definido para as eleições, introduzir o cartão na máquina e digitar o seu código secreto recebido da Comissão Nacional de Eleições. Depois basta escolher o partido ou coligação em que deseja votar e sair. Na Internet o eleitor deverá aceder à página da Comissão Nacional de Eleições, digitar manualmente o número do seu cartão de eleitor e o código secreto, votando depois de forma semelhante ao que pode fazer nos terminais Multibanco. Este sistema tem muitas vantagens, tais como eliminar a operação de contagem de votos (que demora bastante tempo e dá azo a erros e a fraudes) ou permitir a um eleitor votar em qualquer ponto do País, independentemente da freguesia onde está recenseado. Deixa também de ser necessário ter o antiquado sistema de mesas de voto. O projecto do esquema de dados desta aplicação deve obedecer aos seguintes pontos:

- O novo sistema destina-se às eleições para a assembleia da república, eleições autárquicas e presidenciais, devendo poder armazenar os resultados de diversas eleições, de forma a tornar fáceis as comparações entre votações em eleições sucessivas.
- A base de dados regista informação sobre todos os eleitores. Para além da informação habitual (nome, endereço postal, data de nascimento, BI, etc.) é necessário registar também o número do cartão do eleitor e o seu código secreto (este atributo será cifrado na tabela mas isso faz parte da matéria de Bases de Dados II). Não se esqueça de que o número de leitores é muito grande, pelo que é importante normalizar as tabelas que registam informação sobre eleitores para minimizar o espaço ocupado pela base de dados.

- Cada eleitor está recenseado numa freguesia, pelo que apesar de poder votar em qualquer lugar, o seu voto ficará associado à freguesia onde o votante está recenseado. Isto não é muito importante para as eleições para a assembleia da república, que tem carácter nacional, mas é muito relevante para as eleições autárquicas. Claro está que a base de dados necessita de ter informação relativa à divisão administrativa do país, ou seja, precisa de conhecer todas as freguesias, concelhos e distritos portugueses.
- A uma dada eleição concorrem partidos ou coligações de partidos. Estes partidos e/ou coligações podem variar ao longo do tempo, ou seja de eleição para eleição. No caso particular das eleições autárquicas, e para simplificar, vamos admitir em todas as freguesias concorrem os mesmos partidos e/ou coligações (como se sabe, não é assim na realidade, visto cada freguesia ter boletins de voto próprios para as autárquicas). Nas eleições presidenciais as candidaturas são, obviamente, em nome individual.
- O sistema tem de garantir o anonimato dos votos, ou seja, na base de dados não poderá ficar armazenada qualquer informação que permita relacionar um eleitor com o seu voto. No entanto, é necessário evitar que o mesmo eleitor vote mais do que uma vez na mesma eleição.

Apresente um diagrama ER que represente correctamente a estrutura da base de dados pretendida. Pode introduzir novos elementos que completem a descrição do problema acima efectuada, se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. Mostre também o esquema físico (tabelas e vistas, se achar necessário ter vistas) e explique os aspectos mais relevantes da preservação da integridade da base de dados.

5. Mostre como poderá ser o esquema da base de dados usada habitualmente para fazer as inscrições nas **Turmas Práticas do DEI**, procurando apresentar uma solução o mais completa possível (esqueça todas as componentes do sistema de informação do DEI que não têm nada a ver com as inscrições nas turmas práticas). Apesar de funcionamento do processo de inscrição nas turmas práticas ser perfeitamente conhecido dos alunos de Bases de Dados I, recorda-se de seguida os aspectos principais a ter em conta:

- É necessário registar as informações básicas sobre cada aluno (nome, BI, número de aluno, ...). De igual modo, também é necessário registar os dados básicos sobre cada disciplina (nome, código, créditos, ano, ...) e sobre cada docente (nome, BI, categoria, data de entrada ao serviço, ...).
- Uma disciplina pode ter várias turmas, sendo necessário registar o número máximo de alunos que se podem inscrever em cada turma, o número de alunos inscritos, o horário da turma, o número de horas, a sala, etc.
- Como se sabe, cada docente pode dar várias turmas de várias disciplinas.
- As turmas práticas funcionam numa determinada sala, pelo que o sistema também precisa de conhecer as salas (descritas pelos atributos adequados ao problema).

A solução apresentada deve incluir o diagrama ER (conceptual) e o esquema físico. É preciso deixar claro como é que aspectos importantes relativos à integridade dos dados são tratados (explicar se as verificações de integridade são declarativas ou se se teria de recorrer a outros processos). É importante que a preservação da integridade seja garantida em todos os dados necessários para a correcta inscrição dos alunos nas turmas práticas, nomeadamente:

- Garantir que um aluno não se inscreve em mais do que uma turma da mesma disciplina no mesmo ano.

- Garantir que não há duas turmas marcadas para a mesma sala e para o mesmo dia da semana.
- Garantir que o mesmo docente não tem atribuídas mais do que uma turma para o mesmo dia e para a mesma hora.
- Garantir que não há duas turmas a funcionar ao mesmo tempo na mesma sala.
- Garantir que a carga horária de um docente nunca é superior a 20 horas por semana;
- Outras verificações de integridade que lhe pareçam relevantes.

Pode introduzir atributos ou novos elementos que completem a descrição do problema acima efectuada (desde que estejam de acordo com as regras usadas no DEI para as turmas práticas). Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados (mas muito sucintamente).

6. A **Liga de Clubes** pretende desenvolver uma aplicação de base de dados para registar informação sobre clubes, jogadores, jogos, resultados dos jogos, aplicações de cartões a jogadores, faltas cometidas pelos jogadores, golos marcados por cada jogador, etc, etc. Esta base de dados servirá essencialmente para disponibilizar informação sobre o campeonato da Liga de Clubes aos próprios clubes e jogadores, mas também a jornais desportivos e ao público em geral. Os pontos que se seguem resumem alguns aspectos a ter em conta na concepção do esquema da base de dados:

- A base de dados deve registar dados sobre várias épocas, de modo a permitir fazer comparações entre clubes e jogadores em diferentes épocas. Só considera a 1ª divisão da Liga.
- Os clubes contratam jogadores antes da época de jogos começar (normalmente em Junho e Julho) ou então, de forma mais limitada, em Janeiro. Por isso o conjunto de jogadores de um clube mantém-se bastante estável durante a época de jogos. Os clubes têm em média 20 a 25 jogadores inscritos (a Liga determina o número máximo, que pode variar de época para época) e são esses os jogadores com que pode contar numa determinada época. Há um conjunto de atributos muito evidentes para os clubes: nome, morada da sede, data de fundação, etc.
- A base de dados precisa de conter também informação sobre todos os jogadores que pertençam aos clubes. Um jogador só pode pertencer a um clube (do ponto de vista da Liga isso significa estar inscrito por esse clube) de cada vez. Mas, como se sabe, os jogadores podem mudar de clube. Os dados a registar sobre cada jogador são os essenciais para ter uma pequena “ficha técnica” do jogador (nome, data de nascimento, BI, etc, etc). Claro está que interessa também saber em que clubes já esteve cada jogador e por quanto tempo.
- O calendário de jogos é definido antes da época pela Liga e deve ser registado na base de dados. Um jogo compreende, naturalmente, duas equipas, a visitante e a visitada, e também uma equipa de arbitragem. A base de dados também precisa de registar informação sobre os árbitros credenciados para arbitrar jogos da 1ª divisão. Para cada jogo são definidos quatro árbitros: o árbitro principal, árbitro auxiliar 1, árbitro auxiliar 2 e o 4º árbitro. O mesmo árbitro pode arbitrar vários jogos (em qualquer dos tipos de árbitros) que envolvam os mesmos clubes. Os árbitros vão sendo nomeados pela Liga com alguma antecedência (mas não são definidos no início da época, como o calendário de jogos).
- Em cada jogo importa registar também que jogadores de cada um dos dois clubes foram convocados para o jogo, que jogadores jogaram efectivamente, quais deles foram titulares (i.e., começaram a jogar ao início do jogo), substituições, tempo de jogo em que ocorre cada substituição. Um objectivo importante é saber em cada momento quanto tempo é que cada

jogador já jogou na época em curso (ou em épocas passadas). Para além disso, há ainda uma longa lista de elementos que devem ser registados em cada jogo:

- Golos, quem marcou cada golo, tempo de jogo em que cada golo foi marcado.
 - Faltas cometidas, quem cometeu a falta, tempo de jogo em que a falta foi cometida.
 - Cartões mostrados, jogador que viu o cartão, tempo de jogo em que o cartão foi mostrado.
 - Etc, etc, para outras ocorrências importantes de um jogo de futebol.
- Cada clube tem em cada momento apenas um treinador principal. No entanto, os treinadores mudam com frequência. A base de dados deve permitir registar informação relativa a treinadores, permitindo saber quem é o treinador de cada clube em cada momento e quais foram os treinadores passados de cada clube.

Apresente um diagrama Entidade-Relacionamento (ER) para a base de dados em causa. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Todos os relacionamentos devem estar correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER (mas não é preciso indicar as tabelas). Se houver aspectos relevantes no que toca à maior ou menor normalização das tabelas que seriam geradas a partir do ER deve esclarecer esses aspectos, justificando as suas opções (especialmente em casos em que há hierarquias de entidades).

7. Um grande hospital pretende informatizar a **gestão da administração de medicamentos** aos seus doentes para o que necessita de uma base de dados cuja concepção deve ter em conta os seguintes pontos:
- Cada unidade do hospital (cardiologia, otorino, medicina interna, etc.) tem a sua farmácia própria. Os medicamentos existentes na farmácia de cada unidade são genericamente determinados pelas necessidades habituais dos doentes internados nessa unidade. Contudo, a farmácia de cada unidade tem também alguns medicamentos de uso mais geral e até medicamentos que também são típicos das farmácias de outras unidades. O hospital tem ainda uma farmácia central cujo stock pretende cobrir os medicamentos mais vulgares de todas as especialidades.
 - Os medicamentos usados no hospital são todos genéricos e são descritos por um vasto conjunto de atributos (código, composição química, apresentação [comprimidos, injeções, etc.], modo de utilização, indicações, etc.). Para cada medicamento específico existente em stock é ainda necessário registar a data limite da validade. As farmácias existentes no hospital também têm outros produtos, tais como próteses, algodão, seringas descartáveis, etc. Estes produtos devem também fazer parte das existências (na base de dados) de cada farmácia;
 - A gestão do stock de medicamentos é global. Isto é, é feita com base na totalidade dos medicamentos existentes no hospital (farmácias das unidades + farmácia central). Para tal, a base de dados deve permitir saber que medicamentos/produtos existem em todo o hospital e em que farmácia é que eles estão;
 - Sempre que um médico prescreve medicamentos a um dado doente (internado no hospital) é feita uma requisição. Tipicamente, cada requisição inclui diversos medicamentos. Sempre que possível, a requisição deve ser satisfeita na farmácia da unidade em que o doente está internado. Os medicamentos (de uma requisição) que não existam na farmácia da unidade

em que foi emitida a requisição são levantados de qualquer outra farmácia do hospital em que eles existam.

Apresente um diagrama ER que represente correctamente a estrutura da base de dados pretendida. Pode introduzir novos elementos que completem a descrição do problema acima efectuada, se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. Mostre também o esquema físico (tabelas e vistas, se achar necessário ter vistas) e explique os aspectos mais relevantes da preservação da integridade da base de dados.

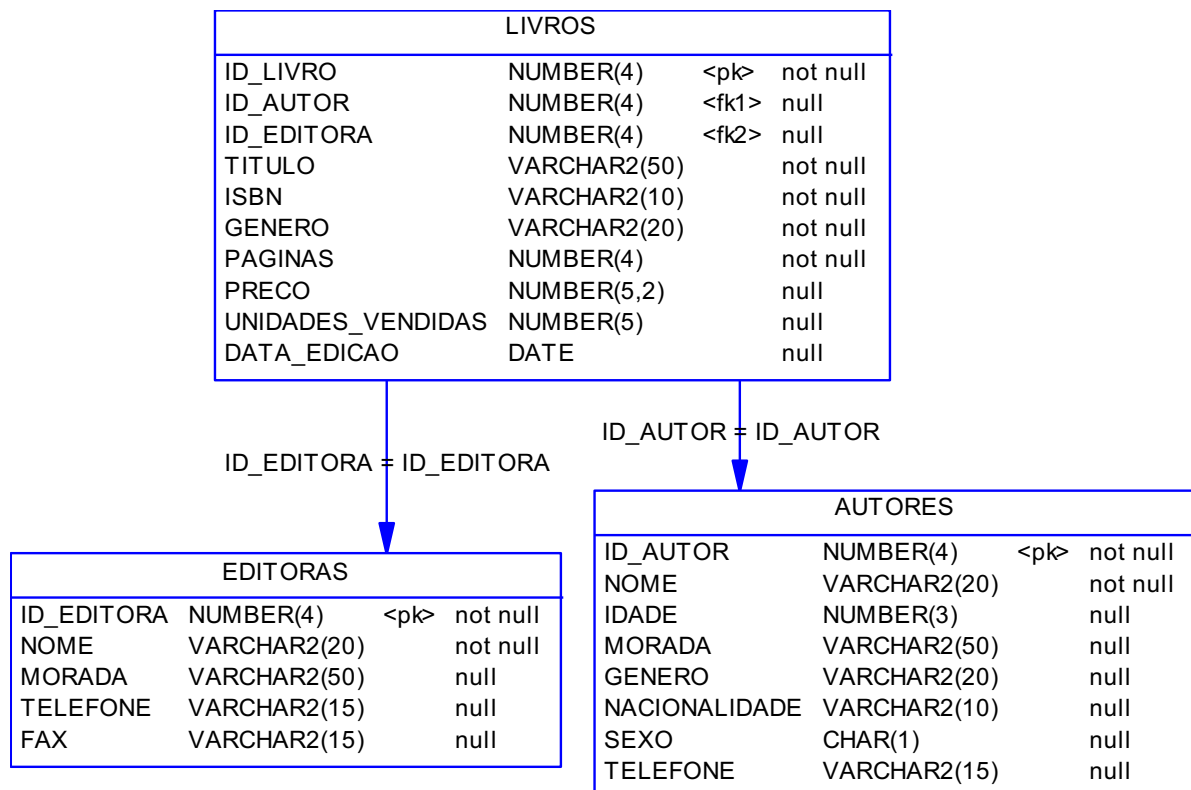
8. Uma loja de produtos electrónicos pretende dispor de uma aplicação de base de dados para duas funções primordiais: fazer a gestão do stock dos produtos e responder rapidamente a pedidos de equivalências entre componentes. Isto é, muitas vezes um cliente pretende adquirir um determinado componente, por exemplo um transistor, mas esse transistor não existe em stock. Neste caso procura-se um transistor (eventualmente de outro fabricante) que seja equivalente (tenha características iguais ou muito aproximadas). Noutros casos, mesmo quando existe em stock o componente pretendido, importa tentar encontrar outro componente equivalente que seja mais barato. O esquema da base de dados deve, portanto, ser pensado para permitir cumprir com eficiência estas duas finalidades (gestão do stock e estabelecimento de equivalências). De seguida indica-se aspectos relevantes para a resolução do problema:

- Todos os componentes electrónicos têm os seguintes atributos genéricos: nome, fabricante, preço unitário, tipo, quantidade de stock mínimo, quantidade de reposição de stock e localização nas prateleiras da loja. O preço de venda ao público de um componente é o preço unitário acrescido de uma margem de comercialização de 30%.
- Os componentes dividem-se em seis famílias principais: resistências, condensadores, transístores, diodos, tirístores e triacs. Todos estes componentes têm um conjunto de atributos específicos (para além dos atributos genéricos referidos anteriormente) que importa registar na base de dados. Por exemplo, qualquer resistência é descrita pelos seguintes atributos específicos: valor, potência, tolerância, tecnologia (ex: resistência de 100K, 1%, 0,5W, fita de carbono). Para as restantes cinco categorias de componentes acontece algo semelhante (não se preocupem em especificar atributos reais; basta admitir que existe, para cada família, um determinado número de atributos específicos).
- Há muitos componentes que não se encaixam nas seis famílias principais acima referidas. Neste caso são apenas registados na base de dados os atributos genéricos.
- Para manter o stock a loja faz encomendas periodicamente aos seus fornecedores. Toda a informação relevante relacionada com estas encomendas deve ser registada na base de dados (data da encomenda, data da entrega, preço total, fornecedor, lista de todos os produtos encomendados nessa encomenda e os respectivos preços unitários). Estas encomendas são numeradas automaticamente pelo sistema.
- A aplicação de base de dados deve registar também os elementos relativos aos fornecedores da loja, sendo os atributos destes fornecedores os atributos habituais que descrevem uma empresa (nome, morada, telefone, etc.).
- Todas as vendas ao público são também registadas, devendo a base de dados guardar a informação habitual que consta do recibo fornecido ao cliente (lista de componentes vendidos e respectivo preço, data, nome do cliente, nº de contribuinte, etc.). O nome do cliente e nº de contribuinte, não são de preenchimento obrigatório, pois muitos clientes não o exigem. Claro está que o funcionamento da base de dados leva sempre a que ao registo de uma venda se siga a actualização da quantidade em stock dos produtos vendidos.

- Um aspecto importante consiste na resposta a pedidos de equivalências de componentes. Como é evidente, a partir do momento em que a base de dados contém as características técnicas dos componentes (os tais atributos específicos de cada família) é possível estabelecer equivalências através de comparação das características técnicas de um componente com os da sua família. Contudo este processo é pouco eficiente, como é fácil de antever. Pretende-se, por isso, introduzir na base de dados a informação relativa a equivalências fornecida pelos fabricantes. Esta informação é muito simples e consiste apenas em dizer, por exemplo, que o componente X é equivalente aos componentes Y, W e Z, tornando desnecessária a comparação das características técnicas. O esquema da base de dados deve permitir a integração desta informação de equivalências.
- a. Apresente um diagrama ER que represente correctamente a estrutura da base de dados pretendida. Pode introduzir novos elementos que completem a descrição do problema acima efectuada, se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. Mostre também o esquema físico (tabelas e vistas, se achar necessário ter vistas) e explique os aspectos mais relevantes da preservação da integridade da base de dados.
- b. O esquema definido na alínea anterior assume que há seis famílias de componentes (resistências, condensadores, transístores, diodos, tiristores e triacs), para além de componentes genéricos. Suponha que depois de a aplicação de base de dados estar a funcionar na loja de componentes surge uma nova família de componentes no mercado que é necessário considerar (e que não pode ser tratada como componentes genéricos). Explique como é que este problema (i.e., o aparecimento de uma nova família de componentes) poderia ser tratado, quer no que toca ao esquema de dados (servidor) quer no que diz respeito à interface com o utilizador (cliente).

3. Linguagem PL/SQL

As tabelas seguintes pretendem modelar uma livraria com uma base de dados muito simples. Existe apenas informação sobre editoras, livros e autores. Todos os livros são editados por uma editora. Essa relação é indicada através da coluna de chave forasteira, *id_editora*, na tabela *livros*, que indica um valor existente na coluna de chave primária *id_editora* da tabela *editoras*. Cada livro é escrito por um, e apenas um autor. Essa relação é indicada através da coluna de chave forasteira, *id_autor*, na tabela *livros*, que indica um valor existente na coluna de chave primária *id_autor* da tabela *autores*.



Os comandos necessários para criar as tabelas e inserir dados de teste encontram-se na página da disciplina.

Baseando-se nas tabelas anteriores, resolva os seguintes problemas:

- Usando a linguagem SQL, crie a seguinte tabela:

```
temp(col1 number(10),col2 number(20),message varchar2(100))
```

- Faça um programa PL/SQL que insira 100 registos na tabela *temp* com os seguintes dados:

COL1	COL2	MESSAGE
1	100	Col1 é impar
2	200	Col1 é par
...
100	10000	Col1 é par

3. Faça um programa em PL/SQL que peça o número de um livro (*id_livro*) ao utilizador e que aumente o preço desse livro de acordo com o seguinte critério:
- se ele custar menos de €25 é aumentado 10%;
 - se custar €25 ou mais é aumentado 6%;
 - se for um livro de aventuras não é aumentado o seu preço.

Utilize a seguinte linha de código para ler o valor de *id_livro*:

```
i_id_livro=&Introduza_ID_Livro;
```

4. Analise e complete o seguinte programa em PL/SQL. Este programa aumenta os preços dos livros dos géneros romance e aventuras, de acordo com o seguinte critério:
- os livros que custem menos de €25 são aumentados em 10%;
 - os livros que custem mais de €25 e menos de €50 são aumentados em apenas 6%;
 - os livros que custem mais de €50 não são aumentados.

```
declare
    v_id_livro livros.id_livro%type;
    v_preco livros.preco%type;
    cursor c1 is
        select id_livro,preço
        from livros
        where ...
        for update of preco;
begin
    open c1;
    loop
        fetch ...;
        exit ...;
        if v_preco<=25 then
            update livros set preco=preco*1.1 where current of c1;
        else
            ...
        end if;
    end loop;
    close c1;
end;
```

5. Volte a escrever o programa anterior utilizando um ciclo “*for de cursor*”.
6. Faça um programa em PL/SQL que peça um género ao utilizador e que aumente os preços dos livros com esse género em 10%, de acordo com o seguinte critério:
- Só aumenta preços se o preço total de todos os livros com esse género for inferior a €150;
 - Se obedecer à condição anterior, então começa por aumentar os livros que tenham o preço mais baixo;
 - Pára assim que o aumento de um livro levar à violação da primeira condição.

Faça um programa que copie os campos *id_livro*, *preco* e *titulo* dos 8 livros mais caros da tabela *livros* para a tabela *temp* (criada na pergunta 1).

7. Faça um programa utilizando cursores, que calcule o preço total dos livros de informática e que conte quantos livros deste género têm preços superiores a €25 e quantos têm um número de páginas maior que 500. Os resultados devem ser armazenados na tabela *temp* do seguinte modo:
- em *col1* fica o número de livros com preço superior a €25;
 - em *col2* fica o número de livros com um número de páginas maior que 500;
 - em *message* fica ‘Preço Total dos Livros de Informática=€valor’.

8. Faça um programa que para cada autor que tenha publicado livros de informática, copie os seguintes dados para a tabela *temp*:
- em *col1* fica o *id_autor*;
 - em *col2* fica o número de livros que publicou;
 - em *message* fica nome do autor invertido.
9. Faça um programa PL/SQL que peça o número de um livro (*id_livro*) ao utilizador e se este pertencer ao autor 17 o mude para o autor 90. Como este autor ainda não existe na tabela autores, o programa deverá começar por criá-lo com a seguinte informação:
- *id_autor* = 90
 - *nome* = João Varajão
 - *morada* = Braga
 - *sexo* = M
 - *nacionalidade* = Portuguesa
 - *genero* = Informática

Para experimentar este programa só podem ser considerados os livros com os números (*id_livro*): 5, 8 e 16.

10. Considerando um livro com o *id_livro* diferente do usado no exercício anterior (mas dentro do conjunto 5, 8 e 16), execute o programa anterior outra vez e verifique o erro obtido. A que se deve tal erro?
11. Execute o programa da pergunta 4 mais uma vez considerando o livro com o *id_livro* 1000. A que se deve o erro obtido?
12. Corrija o programa da pergunta 4 através do tratamento das exceções geradas (*dup_val_on_index*, *no_data_found* e *others*).

As mensagens de erro devem ser armazenadas numa tabela denominada erros (deve ser criada previamente), que tem como colunas o código do erro (*number(10)*), a mensagem do erro (*varchar2(250)*) e a data/hora em que ocorreu (*date*). Para obter o código do erro utilize a função *SQLCODE*, e para obter a mensagem do erro utilize a função *SQLERRM*.

13. Observe agora o código apresentado abaixo e complete-o, Observe a solução e compare-a com a que construiu na pergunta 7. Qual as vantagens e/ou desvantagens desta solução relativamente à anterior.


```
declare
    v_id_livro livros.id_livro%type;
    v_id_autor livros.id_autor%type;

    cod number(10);
    mens varchar2(250);
begin
    select ...;

    if v_id_autor=17 then
        begin
            insert ...;
        exception
            when ...
                null;

        end;

        update ...;
    end if;
exception
    when no_data_found then
        cod:=SQLCODE;
        mens:='Livro não existe';
        insert ...;
    when others then
        cod:=SQLCODE;
        mens:=SQLERRM;
        insert into erros values(cod,mens,sysdate);
end;
/
```

14. Faça um programa que para cada autor, contabilize o número total de livros editados e o número total de livros do seu género preferido. Insira essa informação na tabela *temp* do seguinte modo:
- em *col1* fica o número total de livros;
 - em *col2* fica o número de livros dos seu género preferido;
 - em *message* fica o primeiro nome do autor.
15. Faça um **procedimento** que limpe o conteúdo da tabela *temp*, e calcule o número e o preço total dos livros de cada género. Os resultados devem ser armazenados na tabela *temp* do seguinte modo:
- em *col1* fica o preço total dos livros de cada género;
 - em *col2* fica o número de livros de cada género;
 - em *message* fica o género.
16. Implemente um **trigger** que mantenha actualizado o registo da tabela *temp*, relativo aos dados guardados na questão anterior. Deste modo, sempre que existam alterações aos registos de livros, estas terão de ser reflectidas na tabela *temp*.
17. Pretende-se implementar uma solução que permita controlar o número de livros editados por cada editora. Para isso resolva as seguintes alíneas:
- a) Adicione a coluna *nlivros_editados* à tabela *editoras*. Esta coluna deverá ser do tipo **NUMBER(5)**.
 - b) Implemente um comando **SQL** que preencha a coluna criada na alínea anterior, através dos dados existentes na tabela *livros*.
 - c) Crie um **trigger** que permita manter actualizada a coluna *nlivros_editados*.

18. Crie uma *package* que inclua todos os programas resolvidos na presente ficha.
19. Analise e execute o código seguinte. Este procedimento aumenta o preço dos livros do género fornecido por parâmetro e devolve a lista dos livros modificados.

```
create or replace package types is
    type refcursor is ref cursor;
end;
/
```

Criação de um tipo de dados do tipo ref cursor que vai ser usar para permitir que o procedimento devolva um cursor

```
create or replace procedure
aumenta_preco
(p_genero livros.genero%type,
res1 in out types.refcursor)
is
    cursor c1 is
        select id_livro,preco
        from livros
        where genero=p_genero
        for update of preco;
begin
    for r in c1
    loop
        if r.preco < 20 then
            update livros
            set preco=preco*1.1
            where current of c1;
        elsif r.preco<40 then
            update livros
            set preco=preco*1.06
            where current of c1;
        else
            update livros
            set preco=preco*1.02
            where current of c1;
        end if;
    end loop;

    open res1 for
        select id_livro,titulo,preco
        from livros
        where genero=p_genero;
end;
/
```

O cursor vai ser devolvido através do parâmetro de entrada/saída (*in out*) denominado *res1*

Abertura do cursor que vai ser devolvido por parâmetro

20. Analise, compile e execute o seguinte programa em JAVA.

```
import java.sql.*;
import oracle.jdbc.driver.*;

public class OracleStoredProc {

    public OracleStoredProc() {
    }

    public static void main(String[] args) {
        System.out.println("Executing...");

        try {
            // Connect to the Database
            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection conn =
                DriverManager.getConnection("jdbc:oracle:oci8:bd2/bd2@webstore");

            String query = "begin  aumenta_preco(?,?); end;";

            CallableStatement stmt = conn.prepareCall(query);

            // register the type of the out param - an Oracle specific type
            stmt.registerOutParameter(2, OracleTypes.CURSOR);

            // set the in param
            stmt.setString(1, args[0]);

            // execute and retrieve the result set
            stmt.execute();
            ResultSet rs = (ResultSet)stmt.getObject(2);

            // print the results
            while (rs.next()) {
                System.out.println(rs.getInt(1) + "\t"+
                    rs.getFloat(2));
            }
        } catch (java.lang.ClassNotFoundException e) {
            System.out.println("Error connecting");
            return;
        } catch (java.sql.SQLException e) {
            System.out.println("Error executing the command: "+e);
            return;
        }
    }
}
```

21. Faça um **procedimento** que copie para a tabela *temp* o preço e o título de todos os livros que custem mais de um valor a passar por parâmetro. O preço deve ficar no campo *coll* e o título no campo *message*.
22. Implemente uma **função** que receba por parâmetro o código de um livro (*id_livro*) e devolva o seu preço.
23. Implemente um **trigger** que ao remover um autor da tabela *autores*, apague da tabela *livros* todos os livros publicados pelo autor removido.
24. Pretende-se criar uma solução que permita armazenar todos os livros removidos da tabela *livros*. Para isso resolva as seguintes alíneas:

- a) Crie a tabela *livros_removidos* com a mesma estrutura da tabela *livros*, retirando as colunas *id_autor* e *id_editora*.
- b) Crie um **trigger** que mantenha a tabela de livros removidos actualizada, ou seja, quando se remove um livro da tabela *livros*, este deverá ser inserido na tabela *livros_removidos*.
- c) Adicione à tabela *livros_removidos* as seguintes colunas:
 - *utilizador* – coluna para armazenar o utilizador que removeu o livro. Esta coluna deverá ser do tipo VARCHAR2(40).
 - *data* – coluna para armazenar a data em que o livro foi removido.
- d) Altere o **trigger** criado na alínea b) de forma a que este, para além de tratar o livro removido, passe a registar o utilizador e a hora em que foi feita a remoção.
- e) Remova o autor “Cláudio Tereso” da tabela *autores*. Verifique se o autor e os seus livros foram removidos e se os mesmos constam da tabela *livros_removidos*.

25. Crie uma *package* que inclua todos os programas resolvidos na presente ficha.

4. Transacções e controlo de concorrência

Resolva os seguintes problemas:

1. Crie uma tabela DEPl com os mesmos dados que a tabela DEP (*create table dep1 as select * from dep*). Se a tabela DEPl já existir, remova-a (*drop table dep1*).
2. Abra duas janelas SQL (SQL Plus ou iSQLPlus) usando a mesma conta. Com uma delas faça:

```
SQL> select * from dep1;
```

e de seguida faça o mesmo com a outra. Ambas produzem o mesmo resultado, certo?

Chamemos a uma das janelas a Janela A e à outra a Janela B. Execute o seguinte código em A:

```
SQL> update dep1 set local='VISEU' where ndep = 10;
```

```
1 row updated.
```

```
SQL> select * from dep1;
```

Agora faça em B:

```
SQL> select * from dep1;
```

O que aconteceu? Em A viu-se o valor da localização do departamento 10 alterada enquanto que em B ainda se vê a localização antiga. Como justifica isso sabendo que tanto A como B estão ligadas ao mesmo utilizador?

3. Ainda em B digite:

```
SQL> update dep1 set local='LISBOA' where ndep = 20;
```

```
1 row updated.
```

Em A e em B execute:

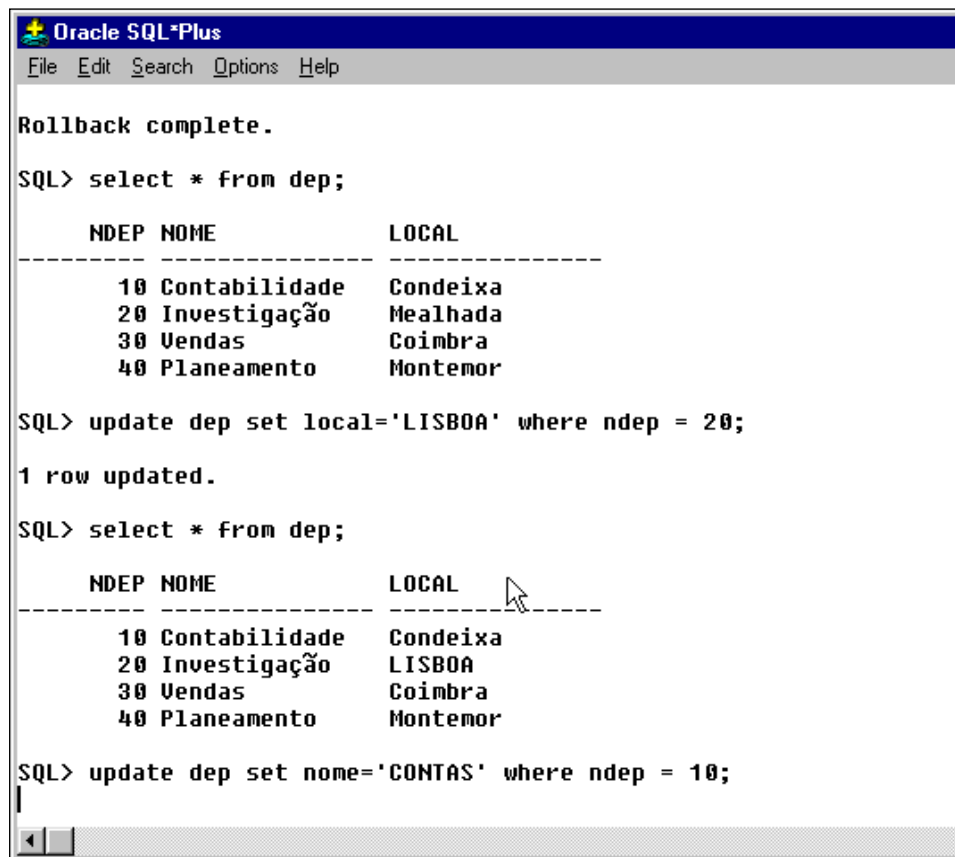
```
SQL> select * from dep1;
```

Tanto A como B só vêem as suas próprias alterações.

Continuando em B faça:

```
SQL> update dep1 set nome='CONTAS' where ndep = 10;
```

O resultado é o da figura seguinte (o facto de a janela ter bloqueado na última linha é que é importante)



```

Oracle SQL*Plus
File Edit Search Options Help

Rollback complete.

SQL> select * from dep;

      NDEP NOME          LOCAL
-----
      10 Contabilidade  Condeixa
      20 Investigação   Mealhada
      30 Vendas         Coimbra
      40 Planeamento   Montemor

SQL> update dep set local='LISBOA' where ndep = 20;

1 row updated.

SQL> select * from dep;

      NDEP NOME          LOCAL
-----
      10 Contabilidade  Condeixa
      20 Investigação   LISBOA
      30 Vendas         Coimbra
      40 Planeamento   Montemor

SQL> update dep set nome='CONTAS' where ndep = 10;

```

Justifique a situação. Como é que se pode desbloquear B?

- Em A execute *commit*. Quais as consequências da execução deste comando? Existe mais do que uma e algumas não são visíveis!...

Faça *rollback* em B. Verifique que as alterações em B foram desfeitas.

- E se em vez de fazer *commit* no passo anterior se tivesse feito *rollback* também desbloquearia? Tente reproduzir a situação anterior e use *rollback* em vez de *commit*. Diga o que observou.
- E se fizesse um *update* mas se o valor a alterar fosse rigorosamente igual ao valor que já lá estava (por exemplo mudar a localização de “Coimbra” para “Coimbra”)? Será que produziria bloqueio do registo “alterado”?

Experimente e responda.

- Reponha o estado das tabelas como estavam no princípio dos exercícios, fazendo *rollbacks* e *update* com os valores iniciais. Execute *commit* em ambas as janelas.

Agora em A execute:

```

SQL> update dep1 set local='VISEU' where ndep = 10;

1 row updated.

```

E em B execute:

```
SQL> update dep1 set local='LISBOA' where ndep = 20;
```

```
1 row updated.
```

```
SQL> update dep1 set nome='CONTAS' where ndep = 10;
```

B bloqueou porque está a tentar alterar um registo bloqueado por A.

Agora em A tente alterar um registo bloqueado por B fazendo:

```
SQL> update dep1 set nome='PESCAS' where ndep = 20;
```

O Oracle detecta o impasse (*deadlock* em inglês) e passado uns segundos quebra-o. Quando um impasse é detectado e quebrado pelo sistema existe sempre informação que se perde. O que é que se perdeu aqui? Note que uma das janelas continua bloqueada!

8. Depois de desbloquear a janela abra agora uma terceira janela, C, com o mesmo *login* e *password* e tente criar uma situação de impasse com 3 sessões.
9. Através de *commits* e *updates* reponha os dados nas tabelas como estavam no princípio da aula. Na janela A inicie uma transacção só de leitura fazendo:

```
SQL> set transaction read only;
```

```
Transaction set.
```

Tente alterar o valor de um registo em A. Execute, por exemplo, o seguinte comando:

```
SQL> update dep1 set local = 'VISEU' where ndep = 10;
```

```
update dep1 set local = 'VISEU' where ndep = 10
```

```
*
```

```
ERROR at line 1:
```

```
ORA-01456: may not perform insert/delete/update operation inside a  
READ ONLY transaction
```

Como pode verificar, não pode alterar dados numa transacção só de leitura.

Em B execute:

```
SQL> update dep1 set local='LISBOA' where ndep = 20;
```

```
1 row updated.
```

Verifique o que as sessões A e C vêm fazendo em ambas:

```
SQL> select * from dep1;
```

Agora volte a B e execute *commit*:

```
SQL> commit;
```

Verifique novamente o que as sessões A e C vêm através de:

```
SQL> select * from dep1;
```

- a. Como justifica?
- b. Quais as consequências do *commit* em B?
- c. Como é que A ainda vê os valores antigos?
- d. Onde é que essa informação está guardada?
- e. O que é que acontece se A não fizer *commit* (ou *rollback*) ou demorar muito tempo a fazê-lo?
- f. Se em vez de termos 3 transacções (correspondentes às 3 sessões) abertas tivéssemos um elevado número de transacções por segundo quais eram as diferenças do ponto de vista da gestão da base de dados Oracle? Parta do princípio que uma delas iniciou também uma transacção só de leitura e ainda não fez o *commit*.

10. Reponha novamente os dados como no princípio e faça em A:

```
SQL> update dep1 set local = 'VISEU' where ndep = 10;  
  
1 row updated.
```

Execute o seguinte código em B:

```
SQL> select * from dep1 where ndep in (10, 20) for update;
```

Porque é que B ficou bloqueado?

11. Faça *commit* em A. B desbloqueia. Agora, ainda em A tente alterar novamente o registo fazendo:

```
SQL> update dep1 set nome = 'CONTAS' where ndep = 10;
```

Justifique o sucedido.

12. Em B execute *commit*. Ao fazer isto A é libertado. Agora em B faça:

```
SQL> select * from dep1 where ndep in (10, 20) for update nowait;  
ERROR:  
ORA-00054: resource busy and acquire with NOWAIT specified  
  
no rows selected
```

Porque é que isto aconteceu?

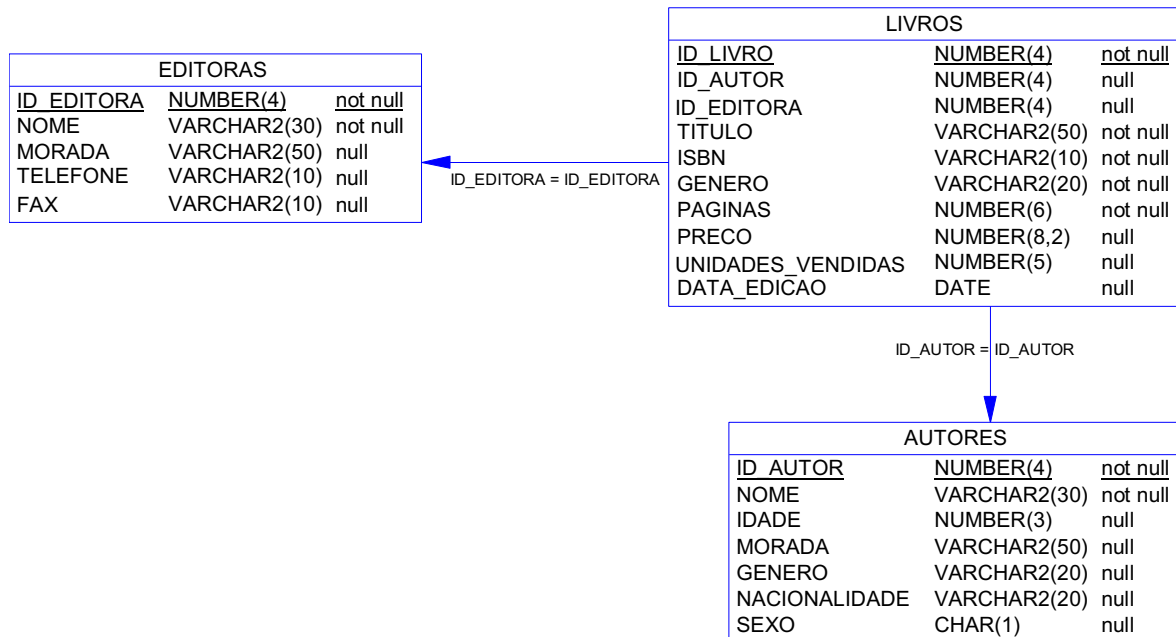
13. Faça *commit* em A e repita o comando em B:

```
SQL> select * from dep1 where ndep in (10, 20) for update nowait;
```

Quais as diferenças de resultados? Os registos estão bloqueados? Por quem?

5. Exercícios extra - SQL

As tabelas seguintes pretendem modelar uma livraria com uma base de dados muito simples. Existe apenas informação sobre editoras, livros e autores. Todos os livros são editados por uma editora. Essa relação é indicada através da coluna de chave forasteira, *id_editora*, na tabela *livros*, que indica um valor existente na coluna de chave primária *id_editora* da tabela *editoras*. Cada livro é escrito por um, e apenas um autor. Essa relação é indicada através da coluna de chave forasteira, *id_autor*, na tabela *livros*, que indica um valor existente na coluna de chave primária *id_autor* da tabela *autores*.



Os comandos necessários para criar as tabelas e inserir dados de teste encontram-se na página da disciplina.

Baseando-se nas tabelas apresentadas na secção 10, resolva os seguintes problemas:

1. Selecione toda a informação relativa aos autores.
2. Mostre os títulos dos livros publicados.
3. Selecione todos os géneros de livros editados, evitando as repetições.
4. Selecione todos os livros cujo preço esteja entre 5000 e 7000.
5. Mostre todos os livros que sejam do género Informática.
6. Selecione os livros policiais com mais de 500 páginas.
7. Selecione os géneros dos livros que tenham mais de 700 páginas ou preço superior a 5000.
8. Obtenha a lista dos livros ordenada de forma descendente por preço.

9. Obtenha uma lista de ISBNs e títulos de livros ordenada ascendentemente pelas vendas e descendentemente pelo preço.
10. Seleccione os IDs dos autores que tenham livros editados.
11. Seleccione todos os livros excepto os policiais.
12. Obtenha a lista de autores ordenada por nome, cuja idade seja superior a 30 anos e o género principal seja os policiais.
13. Seleccione todos os livros excepto os policiais usando o NOT.
14. Apresente a lista dos autores em cujos nomes aparecem as letras 'a' e 'b'.
15. Seleccione os títulos dos livros que começam por 'O' e contém a sequência de letras 'st' e também os que começam por 'M' e contém a sequência 'to'.
16. Seleccione os títulos dos livros que sejam dos géneros policiais, romances e informática.
17. Obtenha o seguinte *output*:

```
DESCRICAO_PRECO
-----
O livro "Microsoft Office 2000 para todos" custa 5500 escudos.
O livro "Microsoft Office 2000 sem fronteiras" custa 6400 escudos.
O livro "Fundamental do Word 2000" custa 4800 escudos.
O livro "Domine a 110% Word 2000" custa 4400 escudos.
O livro "Windows NT Server 4 para profissionais" custa 7900 escudos.
O livro "Windows NT Server 4 Curso Completo" custa 6450 escudos.
O livro "Tecnologia dos Sistemas Distribuidos" custa 4950 escudos.
O livro "TCP/IP em redes Microsoft para profissionais" custa 6500 escudos.
O livro "Redes Locais em Windows 98 e 95" custa 4400 escudos.
O livro "Tecnologia de Base de Dados" custa 4950 escudos.
O livro "O mistério da porta trancada" custa 5750 escudos.
O livro "Os cinco na praia" custa 1950 escudos.
O livro "A outra cara metade" custa 14350 escudos.
O livro "A balada de Hill Street" custa 9950 escudos.
O livro "A balada de Nova Iorque" custa 3300 escudos.
O livro "O Corsário Negro" custa 5470 escudos.
O livro "A lenda do véu dourado" custa 7650 escudos.
O livro "Tu, eu e mais uns quantos" custa 7590 escudos.
O livro "Oracle 8 - Curso completo" custa 4960 escudos.
O livro "A vizinha do lado" custa 4950 escudos.
```

18. Seleccione os títulos e o género dos livros que não sejam dos géneros policiais e romances ou comecem pela letra 'O'.
19. Supondo que o autor recebe 30% das vendas, quanto rendeu cada livro ao seu autor?
20. Obtenha os livros que renderam mais de 10000000 de escudos ao seu autor, ordenados por género e preço.
21. Obtenha os títulos editados em 1999.

22. Construa o seguinte *output*:

```

Titulo Centrado
-----
>>>>>>> Microsoft Office 2000 para todos <<<<<<<
>>>>>>> Microsoft Office 2000 sem fronteiras <<<<<<
>>>>>>>>> Fundamental do Word 2000 <<<<<<<<<<
>>>>>>>>>> Domine a 110% Word 2000 <<<<<<<<<<
>>>>> Windows NT Server 4 para profissionais <<<<<
>>>>>>> Windows NT Server 4 Curso Completo <<<<<<
>>>>>>> Tecnologia dos Sistemas Distribuidos <<<<<<
>> TCP/IP em redes Microsoft para profissionais <<
>>>>>>>>> Redes Locais em Windows 98 e 95 <<<<<<<
>>>>>>>>>>> Tecnologia de Base de Dados <<<<<<<<<
>>>>>>>>>>>>> O mistério da porta trancada <<<<<<<<<
>>>>>>>>>>>>>>> Os cinco na praia <<<<<<<<<<<<
>>>>>>>>>>>>>>> A outra cara metade <<<<<<<<<<<
>>>>>>>>>>>>>>> A balada de Hill Street <<<<<<<<<<
>>>>>>>>>>>>>>> A balada de Nova Iorque <<<<<<<<<<
>>>>>>>>>>>>>>> O Corsário Negro <<<<<<<<<<<<
>>>>>>>>>>>>>>> A lenda do véu dourado <<<<<<<<<<
>>>>>>>>>>>>>>> Tu, eu e mais uns quantos <<<<<<<<<
>>>>>>>>>>>>>>>>> Oracle 8 - Curso completo <<<<<<<<<
>>>>>>>>>>>>>>>>>>> A vizinha do lado <<<<<<<<<<<<

```

23. Selecciona os títulos, preços e datas de edição dos livros editados há menos de 365 dias.**24. Por cada livro calcule o seu custo por página arredondado à 1ª casa decimal e o valor inteiro em escudos superior e inferior, para os livros cujo número de páginas esteja entre 400 e 700. Os resultados devem ser ordenados por custo.****25. Obtenha o seguinte *output*:**

Titulo	Preço em Escudos	Preço em Euros
Microsoft Office 2000 para todos	5500	27.433884
Microsoft Office 2000 sem fronteiras	6400	31.923065
Fundamental do Word 2000	4800	23.942299
Domine a 110% Word 2000	4400	21.947107
Windows NT Server 4 para profissionais	7900	39.405034
Windows NT Server 4 Curso Completo	6450	32.172464
Tecnologia dos Sistemas Distribuidos	4950	24.690496
TCP/IP em redes Microsoft para profissionais	6500	32.421863
Redes Locais em Windows 98 e 95	4400	21.947107
Tecnologia de Base de Dados	4950	24.690496
O mistério da porta trancada	5750	28.680879
Os cinco na praia	1950	9.726559
A outra cara metade	14350	71.577498
A balada de Hill Street	9950	49.630391
A balada de Nova Iorque	3300	16.460331
O Corsário Negro	5470	27.284245
A lenda do véu dourado	7650	38.158039
Tu, eu e mais uns quantos	7590	37.85876
Oracle 8 - Curso completo	4960	24.740376
A vizinha do lado	4950	24.690496

26. Obtenha o título em maiúsculas, minúsculas e com a primeira letra (de cada palavra) maiúscula e restantes minúsculas dos livros editados.**27. Obtenha os livros (título e género) que sejam do género informática, independentemente do modo como o género foi escrito (maiúsculas ou minúsculas).****28. Para cada autor, obtenha o seu primeiro e último nome.****29. Mostre o título, género e o nome do autor dos livros editados pela editora 'FCA - EDITORA'. O título do livro deverá aparecer em maiúsculas, o género em minúsculas e o nome do autor com a primeira letra maiúscula e o resto em minúsculas. Ordene o resultado por nome do autor.**

30. Escreva um comando que devolva o número de livros editados. O resultado deve ser semelhante ao que se segue.

```
Total de livros
-----
20
```

31. Escreva um comando que determine quantos livros de Informática foram editados. O resultado deve ser semelhante ao que se segue.

```
Total de livros de Informática
-----
11
```

32. Escreva um comando que conte o número de livros editados, calcule o preço médio dos livros e o total de livros vendidos.

```
Total de Livros Preço Médio Total de Livros Vendidos
-----
20 6108.5 171376
```

33. Mostre a lista dos livros Policiais e o respectivo preço e com um aumento de 13,55% para estes livros. O preço depois do aumento deverá ser arredondado na primeira casa decimal.

TITULO	GENERO	PRECO	PRECO_COM_AUMENTO
O mistério da porta trancada	Policia	5750	6529.1
A balada de Hill Street	Policia	9950	11298.2
A balada de Nova Iorque	Policia	3300	3747.2

34. O mesmo que na pergunta anterior, mas agora com o preço depois do aumento arredondado para um número inteiro.

TITULO	GENERO	PRECO	PRECO_COM_AUMENTO
O mistério da porta trancada	Policia	5750	6529
A balada de Hill Street	Policia	9950	11298
A balada de Nova Iorque	Policia	3300	3747

35. Encontre o preço mais baixo, mais alto e o preço médio de todos os livros editados pela FCA.

```
PRECO MAIS BAIXO PRECO MAIS ALTO PRECO MÉDIO
-----
4400 14350 6294.6667
```

36. Encontre a diferença entre o preço mais alto e o preço baixo de cada género.

Género	Diferença
Aventura	5640
Informática	3500
Policia	6650
Romance	9400

37. Mostre quantos livros existem para cada género. Ordene o resultado pela género.

Género	Quantidade
-----	-----
Aventura	3
Informática	11
Policial	3
Romance	3

38. Repita o comando anterior, mas apenas dos autores que possuam um ‘a’ no nome ou dos autores que sejam de Coimbra.

Género	Quantidade
-----	-----
Aventura	3
Informática	10
Policial	1
Romance	3

39. Mostre o preço mais baixo dos livros editados por cada autor. Exclua os autores em que o preço mínimo seja inferior a 6000 e ordene o resultado por preço.

NOME	Preço Mínimo
-----	-----
Samuel Santos	6450
Robert Cowart	7590
Eurico Fonseca	7650
Rui Lemos	9950
Ana Capucho	14350

40. Mostre o preço médio de cada género de livro, ordenando o resultados por ordem crescente dos preços médios. O preço médio deverá ser arredondado para o menor inteiro possível que seja superior ou igual ao seu valor. Apenas deverão ser mostradas os géneros com mais do que 4 títulos.

Género	Preço Médio
-----	-----
Informática	5565

41. Indique o número máximo, mínimo e médio de unidades vendidas e a quantidade de livros de cada género e de cada editora. Exclua os títulos cujo nome do autor seja ‘Paulo Loureiro’ e resultados colectivos que apresentem um número máximo inferior ou igual a 10000. Ordene por editora, género e depois por número máximo de unidades vendidas.

EDITORIA	GENERO	MAXIMO	MINIMO	MEDIO	Num.Livros
-----	-----	-----	-----	-----	-----
FCA-EDITORIA	Informática	15500	3000	9553.75	8
FCA-EDITORIA	Romance	12726	2100	7342	3
PORTO EDITORA	Policial	20520	20000	20260	2

42. Mostre o nome dos autores que não tenham livros publicados ou que só tenham publicado em 1999.

```
NOME
-----
Ana Capucho
Christian Crumush
Fernando Tavares
Luis Gomes
Norberto Candeias
Pedro Coelho
Rui Lemos
Rui Vega
Tânia Azevedo
Vitor Gonçalves
```

43. Construa o seguinte output:

NOME	Num. Livros
-----	-----
Alves Marques	1
Ana Capucho	1
Carlos Milheiro	1
Christian Crumush	nenhum
Cláudio Tereso	2
Eurico Fonseca	1
Fernando Tavares	1
José Luis Pereira	1
Luis Gomes	nenhum
Maria José Sousa	1
Norberto Candeias	nenhum
Paulo Loureiro	3
Pedro Coelho	nenhum
Robert Cowart	1
Rui Lemos	1
Rui Vega	1
Samuel Santos	1
Sérgio Sousa	2
Tânia Azevedo	nenhum
Vitor Beça	1
Vitor Gonçalves	1

44. Obtenha o livro de Informática mais caro:

TITULO	PRECO
-----	-----
Windows NT Server 4 para profissionais	7900

- a) Usando o *MAX*
- b) Usando o *ALL*
- c) Usando o *EXISTS*

45. Quais os autores que publicaram livros com mais páginas do que o número médio de páginas por livro.

NOME

 Alves Marques
 Ana Capucho
 Cláudio Tereso
 Fernando Tavares
 José Luis Pereira
 Paulo Loureiro
 Robert Cowart
 Rui Lemos
 Sérgio Sousa
 Vitor Beça

46. Construa o seguinte output. (Os livros de informática, o seu preço, o preço médio dos livros editados pelo mesmo autor (só de informática) e a diferença entre o preço do livro e o preço médio).

TITULO	PRECO	PRECO_MED	Diferença
Microsoft Office 2000 para todos	5500	5950	-450
Microsoft Office 2000 sem fronteiras	6400	5950	450
Fundamental do Word 2000	4800	4800	0
Domine a 110% Word 2000	4400	4400	0
Oracle 8 - Curso completo	4960	4960	0
Windows NT Server 4 para profissionais	7900	7200	700
TCP/IP em redes Microsoft para profissionais	6500	7200	-700
Windows NT Server 4 Curso Completo	6450	6450	0
Tecnologia dos Sistemas Distribuidos	4950	4950	0
Redes Locais em Windows 98 e 95	4400	4400	0
Tecnologia de Base de Dados	4950	4950	0

47. Construa o seguinte output.

GENERO	TITULO
Informática	#####
"	Domine a 110% Word 2000
"	Fundamental do Word 2000
"	Microsoft Office 2000 para todos
"	Microsoft Office 2000 sem fronteiras
"	Oracle 8 - Curso completo
"	Redes Locais em Windows 98 e 95
"	TCP/IP em redes Microsoft para profissionais
"	Tecnologia de Base de Dados
"	Tecnologia dos Sistemas Distribuidos
"	Windows NT Server 4 Curso Completo
"	Windows NT Server 4 para profissionais

48. Obtenha o seguinte output:

Hora Actual	Data Actual
12:29:55	17th, of November , 1999

49. Obtenha o dia da semana correspondente a 1 de Janeiro de 2000.

```
Dia da Semana
-----
Saturday
```

50. Obtenha o seguinte output.

NOME	TITULO
Alves Marques	Tecnologia dos Sistemas Distribuídos
Ana Capucho	A outra cara metade
Carlos Milheiro	Domine a 110% Word 2000
Christian Crumush	
Cláudio Tereso	Redes Locais em Windows 98 e 95
Cláudio Tereso	A balada de Nova Iorque
Eurico Fonseca	A lenda do véu dourado
Fernando Tavares	Os cinco na praia
José Luis Pereira	Tecnologia de Base de Dados
Luis Gomes	
Maria José Sousa	O mistério da porta trancada
Norberto Candeias	
Paulo Loureiro	Windows NT Server 4 para profissionais
Paulo Loureiro	O Corsário Negro
Paulo Loureiro	TCP/IP em redes Microsoft para profissionais
Pedro Coelho	
Robert Cowart	Tu, eu e mais uns quantos
Rui Lemos	A balada de Hill Street
Rui Vega	A vizinha do lado
Samuel Santos	Windows NT Server 4 Curso Completo
Sérgio Sousa	Microsoft Office 2000 para todos
Sérgio Sousa	Microsoft Office 2000 sem fronteiras
Tânia Azevedo	
Vitor Beça	Oracle 8 - Curso completo
Vitor Gonçalves	Fundamental do Word 2000
25 rows selected.	

51. Resolva o exercício 15 da aula 3 usando OUTER JOINS.

52. Insira um novo livro com as seguintes características:

Livro 'Informática para todos' da editora FCA, do Sérgio Sousa, editado hoje. O livro custa 4800\$00 e tem 430 páginas, sendo-lhe atribuído isbn 132434 e o id_livro 51, não tendo sido vendido nenhuma unidade.

53. Insira um novo autor 'José de Magalhães' com o código 45.

54. Apague o livro de código 51.

55. Actualize os preços dos livros de aventuras através do seguinte fórmula:

$\text{preco} = \text{preco} * (1 + (1 / \text{n}^\circ \text{ de páginas}))$

56. Visualize as tabelas existentes no esquema do utilizador.

```
TABLE_NAME
-----
AUTORES
EDITORAS
LIVROS
```


57. Visualize todas as *constraints* existentes sobre as tabelas do utilizador.

TABLE_NAME	CONSTRAINT_NAME
AUTORES	SYS_C002777
AUTORES	SYS_C002778
AUTORES	PK_ID_AUTOR
EDITORAS	SYS_C002780
EDITORAS	SYS_C002781
EDITORAS	PK_ID_EDITORA
LIVROS	SYS_C002783
LIVROS	SYS_C002784
LIVROS	SYS_C002785
LIVROS	PK_ID_LIVRO
LIVROS	FK_LIVROS_ID_EDITOR_EDITORAS
LIVROS	FK_LIVROS_ID_LIVRO_AUTORES

58. Crie a tabela *Autores2*, análoga à tabela de *Autores* contendo as seguintes restrições:

- Os nomes devem ser em maiúsculas;
- A idade tem que ser um valor positivo menor 200.

59. Crie uma tabela, *Esbocos*, com as colunas, *nesboco* (number(4)), *nome* (varchar2(20)), e *descricao* (varchar2(60)). A chave primária deve ser *nesboco*. A coluna *nome* deve ser *NOT NULL* e *UNIQUE*. A coluna *descricao* deve ser *NOT NULL*.

60. Crie uma sequência, *seq_esboco* para usar nos valores de *nesboco*. A sequência deve começar em 500 e avançar de 10 em 10. Não deverá voltar ao princípio quando terminar e não deve estar limitada a nenhum valor máximo.

61. Insira vários registos na tabela *esbocos* (usando a sequência *seq_esboco* para preencher o campo *nesboco*) . Insira por exemplo os projectos:

Nome	Descrição
"SQL21Dias"	"Aprenda SQL em 21 dias"
"SQLTodos"	"SQL Para Todos"
"Norm"	"Processos de Normalização"

62. Faça uma cópia da tabela *livros* para uma nova tabela *livros_backup*. Use o comando *CREATE TABLE* com subconsulta. Na cópia de valores transforme os géneros dos livros para maiúsculas e garanta que todos os livros custam mais que 5000 (os que custam menos devem ser alterados de forma a passarem a custar 5000). Relembre o uso das funções *UPPER* e *GREATEST*. Note que como está a produzir alterações sobre algumas colunas irá ter que usar pseudónimos para a tabela *livros_backup* possuir colunas com nomes válidos.

63. Como ao criar a nova tabela *livros_backup* as restrições não são copiadas altere a tabela de modo a incluir as restrições que existiam na tabela *livros* (chave primária e chaves forasteiras). Use o comando *ALTER TABLE*.

64. Acrescente uma coluna, *Edicao*, do tipo *number(7)* a *livros_backup*.

65. Construa a vista autor_livro contendo o nome do autor e o título dos livros editados por eles, utilizando a tabela de autores e a de livros_backup. Ambos os campos devem estar em maiúsculas.
66. Visualize todos os dados da vista construída. De seguida apague todos os registos da tabela livros_backup. Volte a visualizar os dados da vista. O que aconteceu?
67. Apague a tabela livros_backup. Volte a visualizar os dados da vista. O que aconteceu?

6. Exercícios extra - Diagramas ER

1. Um **Laboratório de Análises Clínicas** pretende ter uma base de dados cuja concepção deve ter em conta os seguintes pontos:

- As análises efectuadas no laboratório são de dois grandes tipos: análises ao sangue e análises à urina. Para cada um destes tipos de análises há um conjunto de parâmetros que é medido. Por exemplo, numa análise à urina medem-se coisas como a glicose, a ureia, etc. Qualquer que seja o parâmetro (quer se refira ao sangue quer à urina), e independentemente do doente, tem sempre associada a seguinte informação: nome do parâmetro, valor mínimo (considerado normal) para homens, valor mínimo para mulheres, valor máximo para homens, valor máximo para mulheres, unidades em que é medido e o preço da análise ao parâmetro.
- Quando um médico prescreve uma análise ao sangue ou à urina de um doente, normalmente manda apenas analisar alguns destes parâmetros. O conjunto de parâmetros a analisar varia bastante com a situação do doente, variando quer o número quer o tipo de parâmetros analisados. De qualquer modo, a prescrição médica indica sempre que parâmetros devem ser analisados.
- Como as análises ao sangue e à urina são prescritas em impressos diferentes, a base de dados pretende manter essa diferença. Portanto, se o mesmo doente fizer simultaneamente uma análise ao sangue e uma à urina, do ponto de vista do laboratório estas análises são tratadas em separado.
- As análises são efectuadas pelos técnicos do laboratório. A medição de um dado parâmetro é feita sempre por um único técnico. Contudo, como uma análise pode consistir na medição de vários parâmetros, significa que diversos técnicos participam na execução de uma análise. Por cada parâmetro analisado é necessário registar, para além do valor observado, o técnico que efectuou a análise do parâmetro.
- O laboratório também dispõe de um corpo de médicos. Todas as análises, quer ao sangue quer à urina, são observadas no final por um médico (e só um) que observa os resultados dos diversos parâmetros e escreve um pequeno relatório para o médico assistente. O preenchimento deste relatório é obrigatório para todas as análises.
- Para além dos técnicos e dos médicos o laboratório tem ainda funcionários administrativos. Importa registar os atributos mais relevantes de todos estes empregados (nome, BI, morada, telefone, salário, data de nascimento, data de entrada). Há alguns atributos específicos dos médicos, tais como o ano de licenciatura, média final, número de ordem. Do mesmo modo, também os técnicos têm atributos específicos, como, por exemplo, categoria e especialidade.
- Por razões relacionadas com os serviços de cobrança de impostos, a base de dados deve guardar todas as análises efectuadas nos últimos cinco anos e os respectivos preços.

Apresente um diagrama ER que represente correctamente a estrutura da base de dados pretendida. Pode introduzir novos elementos que completem a descrição do problema acima efectuada, se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. Mostre também o esquema físico (tabelas e vistas, se achar necessário ter vistas) e explique os aspectos mais relevantes da preservação da integridade da base de dados.

2. A **Folha em Branco**, editora especializada em manuais escolares para todos os graus de ensino com excepção do universitário, pretende ter uma base de dados cuja concepção deve ter em conta os seguintes pontos:

- Cada livro pode ter mais do que um autor, sendo cada autor caracterizado (pelo menos) pelos seguintes atributos: nome, nº contribuinte, endereço, telefone, e-mail, etc.
- Cada livro destina-se a uma disciplina do ensino básico ou secundário. Para além disso, todos os livros têm um título, um ou mais autores, ISBN, preço de capa e ano de edição.
- A editora pretende ter na sua base de dados a lista de todas as escolas do ensino básico e secundário existentes no país, sendo importante saber que disciplinas é que funcionam em cada escola e o número de alunos previsto para cada disciplina. Esta informação é importante, pois a editora precisa de publicitar os seus livros junto das escolas e não lhe interessa enviar publicidade a um determinado livro (normalmente uma amostra do livro) para uma escola onde a disciplina a que esse livro se destina não funciona. Para além disto, é preciso saber o nome de cada escola e o seu endereço.
- Cada escola escolhe, de entre todos os livros existentes no mercado para cada disciplina, um livro para cada disciplina. Esta escolha é feita no final de cada ano lectivo para o ano lectivo seguinte. Deste modo a editora vai receber das escolas, no final de cada ano lectivo, a lista dos livros por ela editados que foram adoptados em cada escola. Esta informação é muito importante pois é com base nela que a editora obtém uma estimativa do número de exemplares que é necessário fazer para cada livro (notar que a editora inicialmente só faz amostras dos livros para publicidade, e só quando sabe quantas escola é que adoptaram cada um dos seus livros é que faz a primeira tiragem).
- A editora aceita encomendas das livrarias. Normalmente uma encomenda é composta por vários exemplares de mais do que um livro. A factura correspondente a cada encomenda deve indicar todos os livros que constam da encomenda, o número de exemplares de cada livro, o preço unitário e o preço total, assim como o nome da livraria que fez a encomenda, o seu endereço e número fiscal.
- A editora paga direitos de autor aos autores dos livros por ela editados. Os direitos de autor são definidos como uma percentagem do preço de capa do livro, podendo os direitos serem diferentes de livro para livro e de autor para autor.

Apresente um diagrama Entidade-Relacionamento (ER) para a base de dados da editora em causa. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Todos os relacionamentos devem estar correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER (mas não é preciso indicar as tabelas). Se houver aspectos relevantes no que toca à maior ou menor normalização das tabelas que seriam geradas a partir do ER deve esclarecer esses aspectos, justificando as suas opções.

3. A **Conservatória do Registo Civil** pretende informatizar os seus serviços para o que vai encomendar o desenvolvimento de uma aplicação de base de dados aos alunos do DEI. Essa base de dados deverá registar informação sobre cidadãos nacionais, nomeadamente nascimentos, casamentos, divórcios e óbitos. Os pontos que se seguem resumem os aspectos a ter em conta na concepção do esquema da base de dados:

- A principal função da base de dados é registar informação sobre pessoas. Todas as pessoas nascidas em Portugal têm de ser obrigatoriamente registadas. Para além disso, sempre que

- há um casamento ou um óbito de um estrangeiro residente em Portugal, essa pessoa também tem de ser registada na base de dados (para que se possa registar o casamento ou o óbito).
- O registo de informação sobre pessoas inclui os atributos típicos de pessoas. Seguem alguns exemplos ilustrativos: nome, BI (ou N° de Passaporte para estrangeiros que casem ou morram em Portugal, ou N° de Cédula para recém-nascidos), data de nascimento, lugar de nascimento e país de nascimento, estado civil, etc. Para os nascidos em Portugal o lugar de nascimento é indicado por freguesia, concelho e distrito. No caso de um estrangeiro residente o lugar de nascimento é indicado por cidade, estado (opcional) e país.
 - A informação sobre as pessoas muda ao longo do tempo, pelo que a base de dados deverá permitir registar essas alterações. Em geral, estas alterações são feitas nos momentos em que há actos a registar. Por exemplo, quando é registado um casamento o estado civil das pessoas que casam é actualizado (o mesmo para divórcios). Quando há um óbito é obviamente registado que essa pessoa já não está viva. Há ainda alterações do processo de identificação pessoal quando uma criança deixa de ter N° de Cédula e passa a ter BI.
 - Quando é registada uma pessoa (um recém-nascido), para além de preencher os atributos referentes à pessoa (terá N° de Cédula pessoal em vez de BI), é preciso indicar quem são os pais. Pretende-se aliás que a base de dados permita seguir geneallogicamente a família de uma pessoa (i.e., saber quem foram os avós, bisavós, etc, e o mesmo por ordem inversa para saber quem foram os filhos, netos, bisnetos, etc). O esquema deverá também resolver os casos (raros) em que o recém-nascido é filho de pais incógnitos (pai ou mãe).
 - Como já se pode antever pelos pontos anteriores, a base de dados também terá de registar casamentos. A informação a reter é a data do casamento, o tipo de casamento (comunhão de adquiridos, separação de bens, etc.) e, obviamente, quem são as duas pessoas que se casam. Para além dos casamentos, também é preciso registar as uniões de facto. No entanto, estas só são normalmente registadas quando resultam em filhos, pois para registar uma criança torna-se necessário registar a união de facto dos pais, se for esse o caso. De um modo semelhante aos casamentos, a base de dados também deverá registar os divórcios e separações. Estes também têm uma data e um tipo (de comum acordo ou litigioso), envolvendo também duas pessoas.
 - Por fim, importa ainda registar os óbitos. Neste caso o que é preciso saber é a data e a causa do óbito (há uma lista de causas fornecida pela medicina legal, que inclui a “causa indeterminada”).

Apresente um diagrama Entidade-Relacionamento (ER) para a base de dados em causa. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Repare que neste caso há bastantes verificações de integridade importantes (não permitir que uma pessoa que já morreu se case, não por recém nascidos como pais de outras pessoas, não divorciar pessoas que são solteiras, etc., etc., etc.). Nem todas essas verificações de integridade podem ser garantidas pelo esquema mas é importante que comente o que é (e não é) tratado pelo esquema de dados. É muito importante também que todos os relacionamentos estejam correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER (mas não é preciso indicar as tabelas). Se houver aspectos relevantes no que toca à maior ou menor normalização das tabelas que seriam geradas a partir do ER deve esclarecer esses aspectos, justificando as suas opções (especialmente em casos em que há hierarquias de entidades).

4. Um **Jornal Diário** de grande circulação pretende ter uma base de dados para ajudar a gerir os seus funcionários e colaboradores, assim como os seus clientes (que compram espaço para publicidade) e assinantes. De seguida descreve-se um conjunto de pontos que devem ser considerados no projecto da base de dados:

- Os funcionários do jornal são descritos pelos seguintes atributos: nome, nº de funcionário, morada, telefone, categoria, salário e data de entrada no jornal;
- Para além dos funcionários, o jornal tem um grande número de colaboradores. Estes fazem diversos tipos trabalhos, quer de natureza esporádica quer trabalhos regulares. Alguns exemplos de trabalhos efectuados pelos colaboradores são: crónicas periódicas, crónicas temáticas, entrevistas, reportagens, grandes reportagens, notícias nacionais, fotografia, etc.
- Cada um dos tipos de trabalhos efectuados pelos colaboradores é pago de acordo com um valor preestabelecido. Por exemplo, uma reportagem vale 500 euros, uma grande reportagem vale 1000 euros, uma crónica 50 contos, cada fotografia publicada vale 50 euros, etc. Contudo, dependendo da qualidade do trabalho efectuado, o colaborador pode ainda receber um bónus adicional. Um trabalho pode ser efectuado por vários colaboradores, mas, neste caso, eles têm funções distintas e são pagos independentemente. Por exemplo, uma reportagem pode ter um colaborador que faz a reportagem e outro colaborador que faz as fotografias;
- Os trabalhos efectuados pelos colaboradores podem levar ao pagamento de despesas (e.g., viagens, hotéis, etc.). Neste caso, cada colaborador paga as despesas do seu bolso, entregando depois ao jornal o comprovativo dessas despesas para fins de reembolso. A base de dados deve permitir registar todas as despesas efectuadas com cada trabalho, sendo necessário registar o tipo de despesa (pequeno texto a descrever o motivo da despesa), a importância e a data;
- Os clientes do jornal e os assinantes tanto podem ser pessoas individuais como empresas e/ou instituições. Contudo, os atributos necessários para qualquer dos casos são praticamente os mesmos. Alguns desses atributos (que tantos são válidos para clientes e assinantes individuais como para empresas) são: nome, morada, telefone, número fiscal e tipo (individual ou colectivo);
- Os clientes fazem contratos publicitários com o jornal. Um contrato deve indicar o espaço ocupado, o número de cores a usar, a página e as datas em que o anúncio deve ser publicado. Notar que o contrato pode estipular que o anúncio é publicado em dias não consecutivos;
- As assinaturas são feitas por períodos de seis meses, um ano ou dois anos. A assinatura tem uma data de início e um preço.

Apresente um diagrama Entidade-Relacionamento (ER) para a base de dados em causa. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Nem todas essas verificações de integridade podem ser garantidas pelo esquema mas é importante que comente o que é (e não é) tratado pelo esquema de dados. É muito importante também que todos os relacionamentos estejam correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER (mas não é preciso indicar as tabelas). Se houver aspectos relevantes no que toca à maior ou menor normalização das tabelas que seriam geradas a partir do ER deve esclarecer esses aspectos, justificando as suas opções (especialmente em casos em que há hierarquias de entidades).

5. Uma **Empresa de Aluguer de Veículos** pretende adquirir uma base de dados que deve funcionar de acordo com os seguintes princípios:

- A empresa trabalha quer com clientes individuais quer com empresas e instituições. Por isso é necessário guardar na base de dados informação típica necessária para identificar estes clientes. Não há grande diferença no tratamento dos diferentes tipos de clientes, embora seja habitual as empresas terem desconto no aluguer dos carros, o que não acontece com os clientes particulares;
- A empresa tem uma frota bastante vasta de veículos. Estes são basicamente de três tipos: ligeiros de passageiros, motos e máquinas industriais (bulldozers, retro-escavadoras, empilhadores, etc). Para mais, dentro dos ligeiros de passageiros a empresa oferece uma enorme variedade de veículos, desde os pequenos utilitários até carros de luxo, passando pelas carrinhas comerciais. Qualquer que seja o tipo de veículo é necessário registar na base de dados atributos tais como a matrícula, a marca, a data de aquisição, tipo de combustível, etc. Claro está que também há atributos que são específicos de cada tipo de veículo. Por exemplo, nos ligeiros de passageiros importa registar o número de lugares, número de portas e cilindrada, nas motos o tipo de moto e a cilindrada e nas máquinas industriais é necessário registar o tipo de máquina, o peso e a potência;
- Os clientes podem alugar os veículos no momento em que necessitam deles ou podem fazer uma reserva. Em qualquer dos casos a informação (a indicar que o veículo foi alugado ou foi reservado) só é inserida na base de dados se o aluguer ou a reserva forem efectivamente efectuadas;
- A informação a ter em consideração quando é alugado um veículo consiste em quem o alugou, que veículo foi alugado, data de início de aluguer, data de entrega do veículo, tipo de pagamento, tipo de aluguer, valor no conta-quilómetros no momento de alugar, valor no conta-quilómetros na entrega e preço. No caso das reservas a informação relevante é algo semelhante pois também neste caso é necessário ter em conta quem fez a reserva, que tipo de carro se pretende reservar, qual a data de início do aluguer, a duração do aluguer, o número do cartão de crédito e, no caso do cliente não ter cartão de crédito, o montante deixado como garantia da reserva. Claro está que as reservas podem ser canceladas, nunca chegando a originar um aluguer. Contudo, se a reserva for cancelada no próprio dia de início do aluguer o cliente paga o equivalente ao montante estabelecido como garantia de reserva (no caso do pagamento ser por cartão de crédito é-lhe debitado esse montante);
- Como os veículos a alugar são muito diferentes os preços do aluguer também são muito variados. Para mais a empresa pratica dois tipos de aluguer: aluguer com quilometragem ilimitada em que o cliente paga um determinado montante (dependente do veículo) por dia de aluguer, independentemente dos quilómetros percorridos, e aluguer ao quilómetro em que é pago um montante pequeno por dia de aluguer, ao qual é acrescido o preço por quilómetro percorrido. Qualquer que seja o tipo de aluguer é também necessário pagar um montante diário para seguro. Este montante depende apenas do veículo. Para facilitar a gestão dos preços de aluguer os veículos (qualquer que seja o tipo) estão divididos em categorias. Por exemplo, os ligeiros estão divididos nas seguintes categorias: utilitários, gama média, gama alta, luxo e mini-autocarros. Nas motos e nas máquinas acontece algo semelhante. Para cada categoria os preços do aluguer são sempre iguais, mesmo que os veículos tenham marcas diferentes.

Apresente um diagrama Entidade-Relacionamento (ER) para a base de dados em causa. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Nem todas essas verificações de integridade podem

ser garantidas pelo esquema mas é importante que comente o que é (e não é) tratado pelo esquema de dados. É muito importante também que todos os relacionamentos estejam correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER (mas não é preciso indicar as tabelas). Se houver aspectos relevantes no que toca à maior ou menor normalização das tabelas que seriam geradas a partir do ER deve esclarecer esses aspectos, justificando as suas opções (especialmente em casos em que há hierarquias de entidades).

6. A Empresa de Construção Civil Pato & Bravo Ltd. precisa de uma aplicação de bases de dados para ajudar a gerir as máquinas e trabalhadores adstritos a cada obra. Os pontos que se seguem definem genericamente o problema a tratar:

- A empresa tem diversas obras de construção civil em diferentes pontos do país. Estas obras são de vários tipos, incluindo obras públicas, construção de edifícios de apartamentos, construção de moradias ou reparações em edifícios. Para além disso, é importante registar a data de início, a data prevista para a conclusão, a data de conclusão efectiva, o estado da obra e outros atributos que a análise mais detalhada determinará.
- Uma obra pode ter um número variável de trabalhadores adstritos à obra. Um trabalhador não trabalha ao mesmo tempo em mais do que uma obra. No entanto, um trabalhador pode trabalhar apenas durante parte do tempo que a obra dura, sendo por isso necessário registar o período de tempo em que o trabalhador esteve efectivamente adstrito à obra. Para além disso, por razões de mobilidade, um trabalhador pode trabalhar numa dada obra durante um determinado período, depois transitar para outra obra em curso e, passado algum tempo, voltar à obra inicial. É importante manter o registo de todas as obras em que cada trabalhador trabalhou bem como todos os trabalhadores que trabalharam numa dada obra e os períodos de tempo em que estiveram adstritos à obra.
- Cada obra tem um único encarregado. No entanto, a mesma pessoa poder ser encarregado de mais do que uma obra. Ao contrário dos trabalhadores, que transitam de obra para obra com alguma frequência, o encarregado designado para uma obra mantém-se como encarregado obra até ao fim, só sendo substituído por razões de força maior. Nesse caso simplesmente substitui-se o encarregado, não sendo necessário manter o registo de todos os trabalhadores que foram encarregados da obra.
- Os trabalhadores são descritos por um conjunto de atributos que incluem o nome, função (usado, entre outras coisas, para distinguir os encarregados de outros trabalhadores), data de entrada na empresa, categoria, etc. Os atributos que relevantes são basicamente os mesmos, qualquer que seja a função dos trabalhadores.
- A empresa tem um parque de máquinas que usa nas obras. Estas máquinas dividem-se em dois grandes grupos:
 - O grupo auto, que inclui máquinas do tipo camiões, carrinhas, tractores, empilhadores, etc., que se caracterizam por terem chapa de matrícula e, consequentemente, poderem circular na via pública. Estas máquinas podem ser movidas a gásóleo ou a gás.
 - O grupo construção, que inclui guindastes, betoneiras e outras máquinas usadas na construção civil. Estas distinguem-se pela sua função, por precisarem ou não de operador especializado e pela fonte de energia usada (electricidade, gás ou gásóleo).

Independentemente do tipo de máquina, todas as máquinas são descritas por atributos tais como marca, data de aquisição, estado, etc.

- Uma obra pode usar várias máquinas mas uma máquina só pode ser usada numa obra de cada vez. No entanto, uma máquina pode estar adstrita a uma obra apenas durante parte do

tempo que a obra dura. Importa por isso registar exactamente em que obras é que as máquinas estiverem e em que períodos de tempo.

Apresente um diagrama Entidade-Relacionamento (ER) para a base da Pato & Bravo Ltd. Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada. Todos os relacionamentos devem estar correctamente indicados, para que não haja dúvidas sobre as tabelas que devem ser geradas a partir do ER.

7. O Jardim Zoológico de Lisboa pretende adquirir uma aplicação de base de dados para gerir toda a bicharada nele existente: bichos propriamente ditos, funcionários e patrocinadores. Segue-se um conjunto de considerações que definem genericamente o problema a tratar:

- Os bichos existentes no Jardim estão divididos em três categorias: mamíferos, répteis e aves (isto não segue uma taxinomia perfeita pois o Jardim dá-se a algumas liberdades como, por exemplo, incluir os marsupiais no grupo dos mamíferos).
- Todos animais existentes, independentemente da categoria a que pertencem, têm alguns atributos comuns (nº identificação, nome, espécie, data de nascimento, data de entrada). A data de nascimento pode ser desconhecida, mas esse facto deve constar da base de dados. Do mesmo modo, a data de entrada no Jardim pode coincidir com a data de nascimento, o que significa que o animal nasceu no Jardim. O atributo nome corresponde a um nome próprio dado a alguns animais (e.g., há um gorila chamado Pimpão, um elefante fêmea chamado Natacha, etc.), embora alguns animais nunca cheguem a ter nome.
- Cada categoria de animais tem alguns atributos específicos, muito particularmente os atributos relativos à ficha veterinária. Todos os animais são vistos periodicamente pelo veterinário sendo muitos dos atributos específicos para cada categoria. A título de exemplo, e considerando apenas aspectos relativos às medidas, as aves têm peso, envergadura e altura; os répteis têm peso e comprimento; os mamíferos têm peso, comprimento e altura. Muitos outros dados de interesse para o veterinário são específicos de cada categoria. Para alguns animais estas medidas são estimadas por observação (porque não é muito prático pesar, por exemplo, um urso polar) e sempre que são registados dados veterinários sobre um animal é também registada a data da observação.
- Os animais ocupam instalações (jaulas, recintos, etc) e estas são caracterizadas por um código. Para simplificar, considera-se que os atributos de interesse são a área útil de cada instalação e o tipo de instalação. Pode haver instalações temporariamente vazias e, geralmente, vários animais partilham uma mesma instalação.
- Cada tratador tem a seu cargo um determinado número de instalações. Todos os tratadores estão ocupados e, nalguns casos, há mais do que um tratador para a mesma jaula;
- Cada veterinário existente no Jardim é especializado numa categoria de animais (mamíferos, aves ou répteis) só tratando animais dessa categoria;
- Cada animal pode ter um patrocinador que ajuda nas despesas de alimentação e tratamento dos animais a troco de publicidade aos seus produtos, associada ao animal patrocinado. Por exemplo, os automóveis Peugeot e o Sporting Clube de Portugal patrocinam os leões, o ar condicionado Sanyo patrocina os pinguins, etc.). Nem todos os animais têm patrocínio, mas alguns podem ter vários patrocinadores. Os patrocinadores têm, entre outros, os seguintes atributos: nome da empresa (ou entidade), morada, telefone, produto publicitado, montante anual que a empresa se compromete doar ao Jardim.

- Para além dos tratadores e dos veterinários (que lidam directamente com os animais) o Jardim tem também outros funcionários, que também interessa manter na base de dados. Os funcionários têm pelo menos os seguintes atributos: nº segurança social, nome, endereço, telefone, função, data de nascimento e data de entrada ao serviço do Jardim.

Apresente um diagrama Entidade-Relacionamento para a base do Jardim Zoológico de Lisboa e o esquema físico correspondente (conjunto de tabelas). Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada.

8. A Escola de Natação Bazófiás (EsNaBa) pretende informatizar os seus serviços para o que necessita de uma aplicação de base de dados cuja concepção deve ter em conta os seguintes pontos:

- A escola dispõe de diversos monitores especializados, sendo necessário ter na base de dados, para cada um deles, a seguinte informação: nome, morada, telefone, data de admissão, nº segurança social, nota final no curso de monitores de natação.
 - Os monitores existentes na escola podem dar aulas a qualquer dos níveis etários (infantil, juvenil ou adultos).
 - Dentro de cada nível etário os alunos são divididos por turmas que podem ter no máximo 12 alunos.
 - Todos os alunos existentes na escola têm de estar inscritos numa das turmas existentes.
 - Um monitor pode leccionar mais do que uma turma e, nalguns casos (e.g., as turmas grupo infantil), pode haver mais do que um monitor por turma.
 - Para fins de matrícula, a escola necessita da seguinte informação para os alunos: número, nome, morada, telefone, data de nascimento, peso, altura, data de entrada na escola.
 - Como a escola tem três piscinas e, para a mesma piscina, não pode haver turmas com horários sobrepostos.
 - A escola tem diversos funcionários (administrativos, limpeza, etc) e também pretende ter informação sobre esses funcionários na base de dados (nome, morada, telefone, data de admissão, nº segurança social, departamento).
- a. Apresente um diagrama ER para a base da Bazófiás e o esquema físico correspondente (conjunto de tabelas). Pode introduzir atributos não mencionados no texto acima sempre que isso lhe pareça necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. É particularmente importante comentar o diagrama ER e esclarecer como é que a integridade vai ser preservada.
- b. Mostre que o esquema proposto está adequadamente normalizado. Se houver tabelas que não estejam na FNBC justifique porque é que as deixava normalizadas numa forma normal inferior à FNBC.
- c. Um dos aspectos referidos no enunciado do problema é que na altura da matrícula é registado o peso e a altura dos alunos. No entanto, o peso e a altura dos alunos varia ao longo do tempo, especialmente nos mais jovens que estão em fase de crescimento. Se fosse interessante registar a evolução do peso e altura dos alunos como é que alterava o esquema para acomodar este novo requisito.

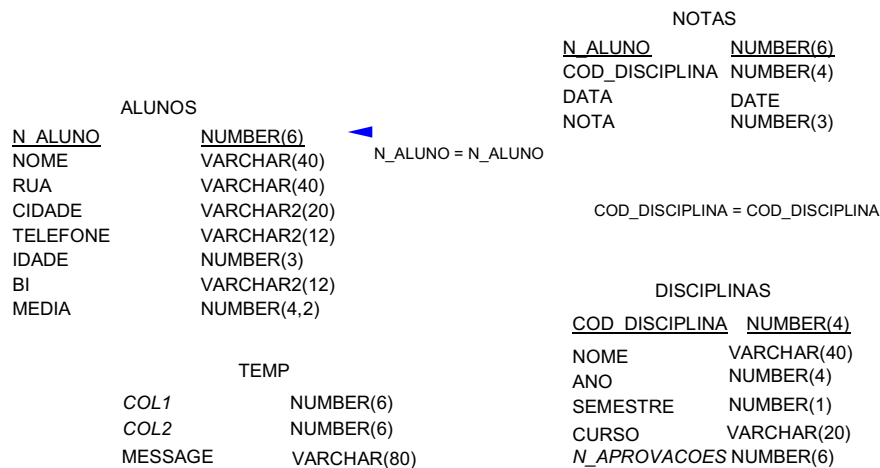
9. O **Aeródromo de Cernache**, nos arredores de Coimbra, precisa de uma base de dados para ter o registo dos aviões nele estacionados, dos seus donos, dos funcionários do aeródromo e dos pilotos. Foi feito o levantamento dos requisitos necessários à base de dados tendo-se obtido a seguinte informação:

- Os aviões são agrupados por categorias (tipos de avião) por razões relativas aos regulamentos da aviação civil, tendo cada tipo de avião os seguintes atributos mais importantes: modelo, capacidade, peso e raio de acção.
- Cada avião tem um número de registo, pertence a um determinado tipo de aviões e tem um lugar fixo num determinado hangar.
- Cada hangar existente no aeródromo é designado por um número, tem uma dada capacidade e uma localização.
- Um avião pode pertencer a uma pessoa singular ou a uma empresa. Como o mercado de compra e venda de aviões usados é muito activo é muito importante registar a data de compra do avião.
- O aeródromo dispõe de um conjunto de mecânicos que fazem a manutenção dos aviões. Os mecânicos têm diversos níveis de especialização, pelo que um dado mecânico só trabalha em certos tipos de avião. Alguns mecânicos trabalham em mais do que um tipo de avião, mas outros só fazem manutenção a um único tipo de avião.
- O serviço de manutenção feito pelos mecânicos nos aviões obedece a regras da aeronáutica civil muito rigorosas. Há um plano de manutenções pré-definido para cada tipo de avião, sendo cada manutenção identificada por um código. Cada manutenção é composta por um determinado número de operações pré-definidas (têm um código que as identifica universalmente), sendo registado sempre o mecânico que efectuou a operação, a data e o tempo gasto pelo mecânico. Para cada serviço de manutenção é necessário saber a data e hora em que o avião deu entrada na oficina e a data e hora de saída.
- Há alguns pilotos que também são funcionários do aeródromo. Cada piloto só tem licença para voar em certos tipos de aviões, podendo um dado piloto ter licença para voar em mais do que um tipo de avião. A licença do piloto tem um número, data de emissão e, eventualmente, restrições impostas pelas autoridades aeronáuticas.
- Todas as pessoas relacionadas com o aeródromo (donos de aviões, mecânicos ou pilotos) têm de ter uma ficha completa na base de dados, contemplando os atributos habituais para a identificação de funcionários.

Apresente um diagrama ER que represente correctamente a estrutura da base de dados pretendida. Pode introduzir novos elementos que completem a descrição do problema acima efectuada, se tal lhe parecer necessário e/ou conveniente. Todos os pressupostos em que se basear, que não tenham sido referidos no texto, têm de ser justificados. Mostre também o esquema físico (tabelas e vistas, se achar necessário ter vistas) e explique os aspectos mais relevantes da preservação da integridade da base de dados.

7. Exercícios extra - PL/SQL

As tabelas seguintes armazenam informação sobre disciplinas, alunos e os exames que os alunos realizaram às várias disciplinas. Os campos que se encontram sublinhados são a chave primária de cada tabela.



Usando a linguagem SQL crie as tabelas apresentadas acima e insira alguns dados para teste.

Usando a linguagem PL/SQL resolva as seguintes questões:

1. Implemente uma **função** que receba por parâmetro o nome de um aluno e devolva o seguinte resultado:
 - se o aluno não existir deve devolver -1;
 - se o aluno existir e for o aluno com melhor média então deve devolver 1;
 - se o aluno existir e não for o aluno com melhor média então deve devolver 0.

A função deverá fazer o tratamento de exceções.

2. Implemente um **procedimento** que comece por apagar todo o conteúdo da tabela *temp*. Depois de apagar o conteúdo da tabela *temp*, o procedimento deverá inserir as médias de todas as disciplinas em cada ano civil (2000, 2001, etc.). Em *col1* deverá ficar o ano e em *col2* a média de todas as disciplinas nesse ano. O procedimento deverá também inserir em *temp* um registo com *col1* igual a -1 e *col2* igual à média de todas as disciplinas em todos os anos. O procedimento deverá usar cursores e fazer o tratamento de exceções.
3. Pretende-se implementar um **trigger de comando** que mantenha actualizada a coluna *n_aprovacoes* (número de aprovações) da tabela disciplinas. Tendo em consideração que a remoção e alteração da nota de um aluno a uma disciplina também afecta o número de aprovações a essa disciplina, será que o cabeçalho seguinte é correcto? Justifique.

```

create or replace trigger t1
before insert or delete or update of data on notas
for each row
declare
...
begin
...
end;
```

4. Implemente um **trigger de comando** que mantenha actualizada a média de cada aluno.
5. Implemente um **trigger de linha** que mantenha actualizada a coluna *n_aprovacoes* (número de aprovações) da tabela disciplinas. Tenha em atenção que a remoção e alteração da nota de um aluno a uma disciplina também afecta o número de aprovações a essa disciplina.
6. Pretende-se implementar um **trigger de linha** que mantenha actualizada a média de cada aluno. Complete o código seguinte. Diga se, para este caso, existe alguma diferença entre o trigger ser *before* ou *after*.

```
create or replace trigger t1
after insert or delete or update on alunos
for each line
declare
...
begin
...
end;
```

7. Implemente uma **função** que receba por parâmetro o nome de uma disciplina e que devolva a posição em que a disciplina se encontra em termos de média, ou seja, para a disciplina com melhor média a função deve devolver 1, para a disciplina com a segunda melhor média a função deve devolver 2, e assim sucessivamente. Caso o nome da disciplina passado por parâmetro não exista, a função deve devolver -1. A função deve usar cursores e fazer o tratamento de excepções.
8. Implemente um **trigger** de comando que mantenha actualizada a coluna *n_aprovacoes* (número de aprovações) da tabela disciplinas. Tenha em atenção que a remoção e alteração da nota de um aluno a uma disciplina também afecta o número de aprovações a essa disciplina. Sempre que um registo da tabela notas é apagado, o *trigger* deverá copiar o seu conteúdo para a tabela *notas_apag* (tem a mesma estrutura da tabela *notas*).
9. Pretende-se implementar um **trigger de comando** que mantenha actualizada a tabela *temp*. Esta tabela contém em *col1* o código de uma disciplina e em *col2* o número de alunos que se submeteram a exame a essa disciplina. Analise o código seguinte e diga se responde ao problema colocado. Diga também se, para este caso, existe alguma diferença entre o *trigger* ser *before* ou *after*.

```
create or replace trigger t1
before insert or delete or update on alunos
for each line
begin
...
end;
```