



DEPARTAMENTO DE INFORMÁTICA

Introdução às Bases de Dados

**Coletânea de Exercícios
(solução ex. 15)**

Ana Paula Afonso

Docentes de IBD 2021

Ana Paula Afonso

III – Interrogações SQL

15. Departamentos e Empregados

Esquema relacional correspondente ao esquema conceptual:

Categoria (cid, código, designação, salário-base)

Departamento (did, código, nome, localização, *director*)

Empregado (eid, número, nome, *did*, *cid*, *chefe*)

Historial_Chefia(chefe, chefiado, data_posse)

Comissão (eid, data_obtenção, valor-comissão)

Emp-Cat (eid, cid, data-admissão)

Nota: a *itálico* estão representadas as chaves estrangeiras.

Escreva comandos SQL-DML para responder aos seguintes pedidos de dados:

- (a) Nomes dos empregados e nomes dos departamentos onde trabalham.

```
SELECT e.nome, d.nome
FROM Empregado e, Departamento d
WHERE e.did=d.did;
```

- (b) Nomes dos empregados e nomes dos departamentos onde trabalham, ordenados de forma ascendente pelo nome do departamento e por ordem descendente do nome do empregado.

```
SELECT e.nome, d.nome
FROM Empregado e, Departamento d
WHERE e.did=d.did
ORDER BY d.nome ASC, e.nome DESC;
```

- (c) Nomes dos empregados que têm salário-base entre 350 e 450 Euros.

```
SELECT e.nome
FROM Empregado e, Categoria c
WHERE e.cid = c.cid
      AND c.salario_base BETWEEN 350 AND 450;
```

- (d) Códigos das categorias e nome dos departamentos onde existem empregados com essas categorias. Certifique-se que não são mostradas linhas duplicadas.

```
SELECT DISTINCT d.nome, e.cid
FROM Empregado e, Departamento d
WHERE e.did=d.did;
```

- (e) Nomes dos departamentos e nomes dos seus empregados, cujo chefe é o diretor do respetivo departamento.

```
SELECT d.nome, e.nome
FROM Departamento d, Empregado e
WHERE d.director=e.chefe
      AND d.did = e.did;
```

- (f) Nome dos departamentos e nomes dos empregados que são chefes e também diretores do respetivo departamento.

```
SELECT dch.nome, ch.nome
FROM Departamento d, Empregado ch, Empregado e, Departamento dch
WHERE d.director=ch.eid
```

```
AND ch.eid = e.chefe  
AND ch.did = dch.did;
```

Alínea adicional:

(f') Nome dos empregados e nome dos respetivos chefes

```
SELECT e1.nome, e2.nome  
FROM Empregado e1, Empregado e2  
WHERE e1.chefe = e2.eid
```

- (g) Códigos das categorias no histórico, de empregados de departamentos no Porto e que são atualmente as categorias de empregados de departamentos cujo diretor tem nome começado pela letra 'P'. Assuma que os empregados não mudam de departamento.

```
SELECT c.codigo  
FROM Categoria c, Emp_Cat ec, Empregado e, Departamento d, Empregado  
e1, Departamento d1, Empregado e3  
WHERE c.cid = ec.cid  
AND ec.eid = e.eid  
AND e.did = d.did  
AND d.localizacao = 'Porto'  
AND ec.cid = e1.cid  
AND e1.did = d1.did  
AND d1.director = e3.eid  
AND e3.nome LIKE 'P%';
```

- (h) Nomes, salários base, e nomes de departamentos dos chefes que não têm nenhum empregado subordinado a trabalhar em departamentos em Lisboa.

```
SELECT e.nome, c.salario_base, d.nome  
FROM Departamento d, Empregado e, Categoria c  
WHERE d.did=e.did  
AND c.cid = e.cid  
AND e.eid IN  
    (SELECT e.chefe  
     FROM Empregado e)  
AND 'Lisboa' NOT IN  
    (SELECT d.localizacao  
     FROM Empregado e1, Departamento d1  
     WHERE e1.did = d1.did  
     AND e1.chefe = e.eid);
```

- (i) Nome dos departamentos e número de empregados que aí trabalham.

```
SELECT d.nome, COUNT(e.eid)  
FROM Empregado e, Departamento d  
WHERE e.did=d.did  
GROUP BY d.nome;
```

- (j) Códigos das categorias, nomes dos departamentos onde existem empregados com essas categorias, e quantidades de empregados nessas condições.

```
SELECT c.codigo, d.nome, COUNT(eid)  
FROM Empregado e, Departamento d, Categoria c  
WHERE e.did=d.did AND e.cid = c.cid  
GROUP BY d.nome, c.codigo;
```

- (k) Nomes dos empregados e o respetivo salário base, cujo salário base é inferior e superior à média dos salários base da empresa, com um desvio de 25 Euros.

```
SELECT e.nome, c.salario_base
```

```

FROM Empregado e, Categoria c
WHERE e.cid = c.cid
      AND (SELECT AVG(c1.salario_base)
            FROM Empregado e1, Categoria c1
            WHERE e1.cid = c1.cid)
            BETWEEN c.salario_base-25 AND c.salario_base+25;

```

- (l) Nomes dos departamentos e nomes dos seus empregados, cujo salário é inferior e superior à média dos salários base do respetivo departamento, com um desvio de 25 Euros.

```

SELECT d.nome, e.nome
FROM Empregado e, Categoria c, Departamento d
WHERE e.did=d.did
      AND e.cid = c.cid
      AND (SELECT AVG(c1.salario_base)
            FROM Empregado e1, Categoria c1
            WHERE e1.cid = c1.cid
              AND e1.did=e.did)
            BETWEEN c.salario_base-25 AND c.salario_base+25;

```

- (m) Nomes dos departamentos e respetivo número de empregados que ganham um salário base superior ao da média dos salários base da empresa.

```

SELECT d.nome, COUNT(e.eid)
FROM Empregado e, Categoria c, Departamento d
WHERE e.did=d.did
      AND e.cid = c.cid
      AND c.salario_base >
          ( SELECT AVG(c1.salario_base)
            FROM Empregado e1, Categoria c1
            WHERE e1.cid = c1.cid)

GROUP BY d.nome;

```

- (n) Nomes dos departamentos e nomes e salários base dos seus empregados que têm um salário base superior ao da média dos salários base do respetivo departamento.

```

SELECT e.did, e.eid, c.salario_base
FROM Empregado e, Categoria c
WHERE e.cid = c.cid
      AND c.salario_base >
          ( SELECT AVG(c1.salario_base)
            FROM Empregado e1, Categoria c1
            WHERE e1.cid = c1.cid
              AND e1.did = e.did);

```

- (o) Nomes dos departamentos e nomes dos seus empregados que foram os primeiros chefes e em que datas. Assuma que os empregados não mudam de departamento.

```

SELECT d.nome, e.nome, hc.data_posse
FROM Historial_Chefia hc, Empregado e, Departamento d
WHERE hc.chefe = e.eid
      AND d.did = e.did
      AND hc.data_posse =
          (SELECT MIN(hc1.data_posse)
            FROM Historial_Chefia hc1, Empregado e1
            WHERE hc1.chefe = e1.eid
              AND e.did = e1.did);

```

- (p) Nomes dos empregados mais bem pagos de cada departamento.

```
SELECT d.nome, e.nome
FROM Empregado e, Categoria c, Departamento d
WHERE e.cid = c.cid
      AND d.did = e.did
      AND (e.did, c.salario_base) IN
          (SELECT e1.did, MAX(c1.salario_base)
           FROM Empregado e1, Categoria c1
           WHERE e1.cid = c1.cid
           GROUP BY e1.did);
```

Solução alternativa:

```
SELECT d.nome, e.nome
FROM Empregado e, Categoria c, Departamento d
WHERE e.cid = c.cid
      AND d.did = e.did
      AND c.salario_base =
          (SELECT MAX(c1.salario_base)
           FROM Empregado e1, Categoria c1
           WHERE e1.cid = c1.cid
           AND e1.did = e.did);
```

- (q) Nomes dos empregados que têm um salário base superior a todos os salários dos empregados dos departamentos aos quais pertencem os empregados mais recentes da empresa de acordo com as datas de admissão na categoria.

-- assumindo mais recente data admissao em Emp_Cat

```
SELECT e.nome
FROM Empregado e, Categoria c
WHERE e.cid = c.cid
      AND c.salario_base >
          (SELECT MAX(c1.salario_base)
           FROM Empregado e1, Categoria c1
           WHERE e1.cid = c1.cid
           AND e1.did IN
               (SELECT e2.did
                FROM Empregado e2
                WHERE
                    (SELECT MIN(ec.data_admissao)
                     FROM Emp_Cat ec
                     WHERE ec.eid = e2.eid)
                    >= ALL
                    (SELECT MIN(ec.data_admissao)
                     FROM Emp_Cat ec
                     GROUP BY ec.eid)));
```

- (r) Nomes dos departamentos e nomes dos seus empregados que têm um salário superior ao salário médio dos chefes de pessoal do respectivo departamento.

```
SELECT d.nome, e.nome
FROM Empregado e, Categoria c, Departamento d
WHERE e.cid = c.cid
      AND d.did = e.did
      AND c.salario_base >
          (SELECT AVG(c1.salario_base)
           FROM Empregado ch, Categoria c1, Empregado e1
           WHERE ch.cid = c1.cid
                 AND ch.eid = e1.chefe
                 AND e1.did = ch.did)
```

```
AND ch.did = e.did);
```

- (s) Nomes dos departamentos e respetivos códigos de categorias onde existem mais de dois empregados.

```
SELECT d.nome, c.codigo
FROM Empregado e, Categoria c, Departamento d
WHERE e.cid = c.cid
      AND d.did = e.did
GROUP BY e.did, c.cid
HAVING COUNT(eid)>2;
```

- (t) Nomes dos departamentos, cujo salário base médio dos seus empregados é inferior a todas as médias dos salários base dos departamentos cujos diretores têm nome começado por P.

```
SELECT d.nome
FROM Empregado e, Categoria c, Departamento d
WHERE e.cid = c.cid
      AND d.did = e.did
GROUP BY e.did
HAVING AVG(c.salario_base) < ALL
      (SELECT AVG(c.salario_base)
       FROM Empregado e, Categoria c, Departamento d, Empregado cd
       WHERE e.cid = c.cid
             AND d.director = cd.eid
             AND cd.nome LIKE 'P%'
       GROUP BY e.did);
```

- (u) Nomes dos departamentos que têm empregados em todas as categorias existentes.

```
SELECT d.nome
FROM Departamento d
WHERE NOT EXISTS
      (SELECT c.cid
       FROM Categoria c
       WHERE NOT EXISTS
             (SELECT e.eid
              FROM Empregado e
              WHERE c.cid = e.cid
                    AND e.did = d.did));
```

Solução alternativa

```
SELECT d.nome
FROM Departamento d, Empregado e
WHERE e.did = d.did
GROUP BY d.nome
HAVING COUNT(DISTINCT(e.cid)) =
      (SELECT COUNT(c.cid)
       FROM Categoria c)
```

- (v) Nomes dos empregados que já passaram por todas as categorias que existem atualmente nos departamentos localizados no Porto.

```
SELECT e.nome
FROM Empregado e
WHERE NOT EXISTS
      (SELECT e1.cid
       FROM Empregado e1, Departamento d
```

```
WHERE e1.did = d.did
      AND d.localizacao = 'Porto'
      AND NOT EXISTS
          (SELECT *
           FROM Emp_Cat ec
           WHERE ec.eid = e.eid
                 AND ec.cid = e1.cid));
```