

21053 - Fundamentos de Bases de Dados

2015-2016

e-fólio A

Resolução e Critérios de Correção

PARA A RESOLUÇÃO DO E-FÓLIO, ACONSELHA-SE QUE LEIA ATENTAMENTE O SEGUINTE:

- 1) O e-fólio é constituído por 4 alíneas, com a cotação 0,5 valores cada. A cotação global é de 2 valores.
- 2) O e-fólio deve ser entregue num único ficheiro PDF, não zipado, com fundo branco, com perguntas numeradas e sem necessidade de rodar o texto para o ler. Penalização de 1 a 2 valores.
- 3) Não são aceites e-fólios manuscritos, i.e. tem penalização de 100%.
- 4) O nome do ficheiro deve seguir a normal “eFolioA” + <nº estudante> + <nome estudante com o máximo de 3 palavras>
- 5) Durante a realização do e-fólio, os estudantes devem concentrar-se na resolução do seu trabalho individual, não sendo permitida a colocação de perguntas ao professor ou entre colegas.
- 6) A interpretação das perguntas também faz parte da sua resolução, se encontrar alguma ambiguidade deve indicar claramente como foi resolvida.
- 7) A legibilidade, a objectividade e a clareza nas respostas serão valorizadas, pelo que, a falta destas qualidades serão penalizadas.

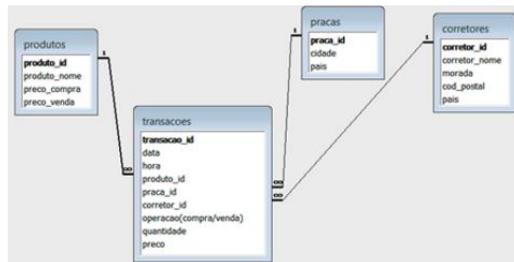
Vetor de cotação:

1.1	1.2.a	1.2.b	1.2.c	pergunta
5	5	5	5	décimas

1) Considere a base de dados de clientes de uma cadeia de restaurantes:

- clientes (IdCliente -> nome, data_nascimento, género, contacto)
- restaurantes (IdRestaurante -> nome, morada, cod_postal)
- menus (IdMenu -> nome, preço, tipo)
- transações (IdCliente, data, hora -> IdRestaurante, IdMenu)

1.1) Defina chave principal e chave estrangeira. Represente graficamente a base de dados relacional, com as respetivas tabelas e ligações de chaves estrangeiras. Siga a seguinte regra para a representar: nas ligações de 1:N a tabela com uma única linha é desenhada em cima e da tabela com várias linhas é desenhada por baixo, conforme a figura seguinte:



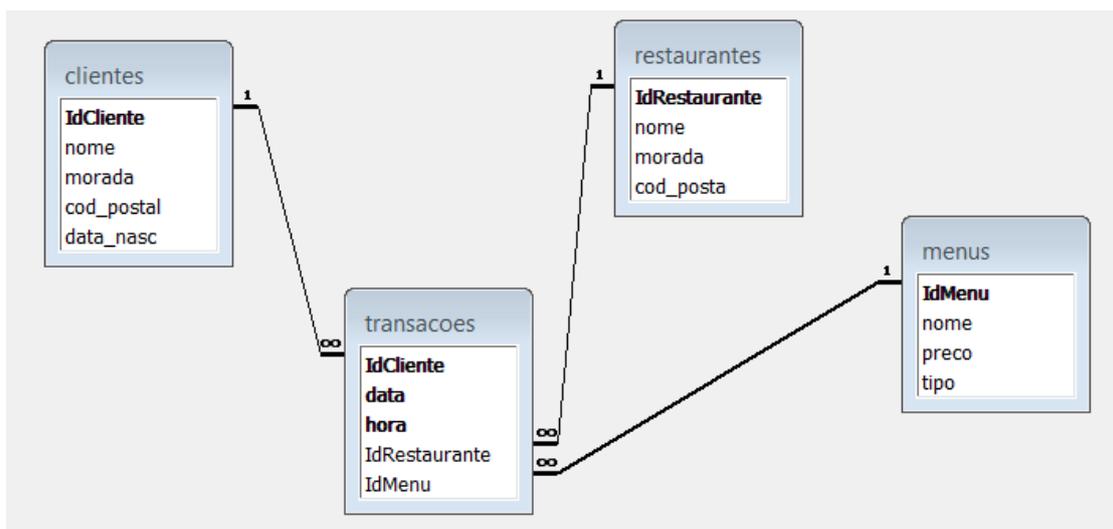
Crie ainda, uma chave substituta (“surrogate key”) na tabela Transações. Qual a nova representação gráfica da base de dados?

Resposta:

Chave principal: atributo que identifica de forma única as linhas (tuplos) da tabela

Chave estrangeira: atributo que é chave principal numa outra tabela, permitindo a relação entre duas tabelas

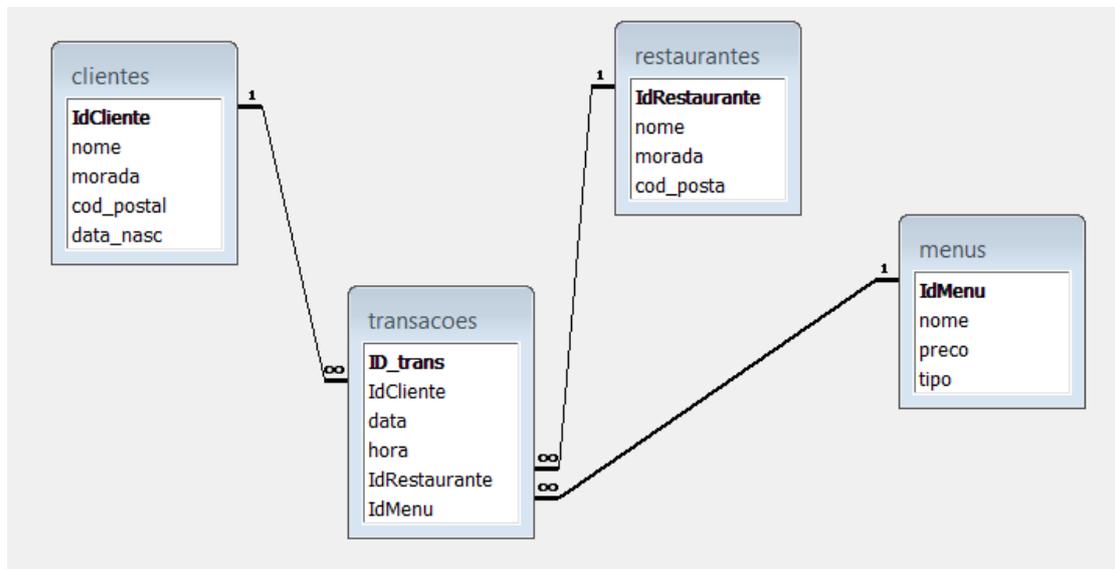
Com transações (IdCliente, data, hora -> IdRestaurante, IdMenu) teremos:



Para a versão com uma chave substituta (“surrogate key”) teremos:

transações (ID_trans -> IdCliente, data, hora, IdRestaurante, IdMenu)

onde “ID_trans” substituí os 3 atributos (IdCliente, data, hora).



Critérios de correção:

- definição de chaves: 1 décima
- representação com atributos concatenados : 2 décimas
- representação com *surrogate key*: 2 décimas
- nota: a chave “id_trans” substituí os 3 atributos (IdCliente, data, hora)
- penalização 50% a 100% se não cumprir as regras de representação

1.2) Exprima em SQL as consultas com a seguinte indentação. Evite, ainda, as cláusulas WITH, TOP e LIMIT:

```
SELECT.....
FROM .....
WHERE.....
AND.....(SELECT.....
          FROM.....
          WHERE.....
GROUP BY .....
HAVING .....
```

1.2.a) Quais os nomes dos clientes que compraram menus com tipo “Infantil” e “Dieta”.

- utilize uma junção
- utilize a união

```
// utilizando a junção
SELECT transacoes.IdCliente
FROM menus, transacoes
WHERE menus.IdMenu=transacoes.IdMenu
AND (menus.tipo Like "Infantil" OR menus.tipo Like "Dieta")
```

```
// utilizando a união
SELECT transacoes.idCliente
FROM menus, transacoes
WHERE menus.IdMenu=transacoes.IdMenu
AND menus.tipo Like "Infantil"
UNION
SELECT transacoes.idCliente
FROM menus, transacoes
WHERE menus.IdMenu=transacoes.IdMenu
AND menus.tipo Like "Dieta"
```

Critérios de correção:

- obrigatório usar junções de tabelas e cláusula OR
- obrigatório usar UNION
- penalização 50% a 100% se não cumprir as regras de indentação

1.2.b) Quais os nomes dos clientes que compraram menus do tipo infantil?

- utilize sub-consultas com cláusula IN
- utilize sub-consultas com cláusula EXISTS

```
// utilizando cláusula IN
SELECT DISTINCT nome
FROM clientes
WHERE IdCliente IN (SELECT IdCliente
                    FROM transacoes
                    WHERE IdMenu IN ( SELECT IdMenu
                                    FROM Menus
                                    WHERE tipo="Infantil" ))
```

```
// utilizando cláusula EXISTS
SELECT DISTINCT nome
FROM clientes AS C
WHERE EXISTS (SELECT *
             FROM transacoes T
             WHERE T.IdCliente=C.IdCliente
             AND EXISTS ( SELECT *
                        FROM Menus M
                        WHERE M.IdMenu=T.IdMenu
                        AND tipo="Infantil" ))
```

Crítérios de correção:

- obrigatório usar várias cláusulas IN
- obrigatório usar várias cláusulas EXISTS
- penalização 50% a 100% se não cumprir as regras de indentação

1.2.c) Qual o número médio de transações dos clientes que fizeram compras na última semana?

```
// solução parcial: número de transações dos clientes que fizeram compras
// na última semana
SELECT DISTINCT IdCliente, COUNT(*) AS Conta
FROM transacoes
WHERE data Between #10/27/2015# And #11/3/2015#
GROUP BY IdCliente
```

```
// resposta final
SELECT AVG(Conta) AS Media
FROM (SELECT DISTINCT IdCliente, COUNT(*) AS Conta
      FROM transacoes
      WHERE data Between #10/27/2015# And #11/3/2015#
      GROUP BY IdCliente)
```

Critérios de correção:

- obrigatório contar as transações dos clientes, antes de fazer a média
- penalização 50% a 100% se não cumprir as regras de indentação