

Tarefa Orientada 14

Subconsultas

Objectivos:

- Subconsultas não correlacionadas
- Operadores *ALL*, *SOME* e *ANY*
- Subconsultas correlacionadas
- Operador *EXISTS*
- Subconsultas incluídas na cláusula *FROM* de uma consulta
- Subconsultas incluídas na cláusula *SELECT* de uma consulta

De modo simplista, uma subconsulta representa uma instrução *SELECT* aninhada dentro de outra consulta. Uma subconsulta pode devolver um valor único, um conjunto de registos que contém uma única coluna, ou ainda um conjunto de registos que contém várias colunas.

Uma subconsulta que devolve um valor único pode ser definida num qualquer local onde seja permitida uma expressão. Uma subconsulta que devolve um conjunto de registos que contém uma única coluna pode ser formulada num local onde se pretenda definir uma lista de valores, tal como, por exemplo, na definição dos valores admissíveis para o operador *IN*. Uma subconsulta que devolve um conjunto de registos que contém várias colunas pode ser definida na cláusula *FROM* de uma consulta.

A sintaxe para uma subconsulta é semelhante à sintaxe utilizada para definir uma consulta normal.

```
(SELECT ALL, DISTINCT lista_de_colunas  
FROM lista_de_tabelas  
[WHERE condições_de_pesquisa]  
[GROUP BY lista_de_colunas]  
[HAVING condições_de_pesquisa])
```

Existem vários locais para utilizar uma subconsulta: na cláusula *WHERE* de uma consulta, como uma condição de pesquisa; na cláusula *HAVING* de uma consulta, como uma condição de pesquisa; na cláusula *FROM* de uma consulta, como especificação de uma tabela e na cláusula *SELECT* de uma consulta, como especificação de uma coluna. Todavia, a utilização mais comum das subconsultas recai nas cláusulas *WHERE* e *HAVING* de uma consulta.

Uma subconsulta pode ser aninhada dentro de outra subconsulta. Contudo, as subconsultas que são aninhadas em mais do que dois ou três níveis de profundidade, podem tornar-se difíceis de analisar e podem deteriorar a performance de execução.

Subconsultas não correlacionadas

Uma subconsulta não correlacionada não depende da consulta exterior. Consequentemente, uma subconsulta não correlacionada é executada em primeiro lugar e, conceptualmente, apenas uma vez.

Deste modo, a consulta exterior depende do resultado devolvido pela subconsulta não correlacionada. São os valores devolvidos pela subconsulta que vão permitir a resolução da consulta exterior.

Resumindo, uma subconsulta não correlacionada é executada apenas uma vez e não depende da consulta exterior. O sentido de execução de uma consulta que contenha uma subconsulta não correlacionada é efectuado de dentro para fora.

EXEMPLOS

1 Formule, analise e execute as dez consultas a seguir apresentadas.

1.1 Consulta que devolve todas as facturas cujo total é superior ao montante médio das facturas registadas na tabela *Facturas*.

```
SELECT NúmeroFactura, DataFactura, TotalFactura
FROM Facturas
WHERE TotalFactura > (SELECT AVG(TotalFactura)
                     FROM Facturas)
ORDER BY TotalFactura
```

Note que a subconsulta é utilizada para calcular o montante médio das facturas registadas na tabela *Facturas*. Depois, a condição de pesquisa, especificada na cláusula *WHERE* da consulta, compara, para cada registo (factura) da tabela *Facturas*, o valor do campo *TotalFactura* com o valor devolvido pela subconsulta, de modo a devolver apenas as facturas cujo total é superior ao montante médio das facturas registadas na tabela *Facturas*.

A seguir, apresenta-se o resultado obtido.

	NúmeroFactura	DataFactura	TotalFactura
1	P-0608	2006-04-11 00:00:00	20551,18
2	P-0259	2006-04-16 00:00:00	26881,40

Note que o resultado devolvido pela subconsulta é o valor 4380,9305.

No exemplo anterior, a subconsulta, através da utilização de uma função de agregação, devolveu um valor agregado que foi usado como condição de pesquisa na cláusula *WHERE* da consulta exterior. Neste tipo de situação, tem que ser utilizada uma subconsulta. No entanto, em muitos casos, as subconsultas podem ser substituídas por junções internas ou externas de tabelas, assim como as operações de junção também podem ser substituídas por subconsultas. Por exemplo, a consulta 1.2 pode ser expressa na forma da consulta 1.3, obtendo-se o mesmo resultado.

1.2 Consulta que utiliza uma subconsulta para devolver o número, a data e o montante total das facturas dos fornecedores de Lisboa.

```

SELECT NúmeroFactura, DataFactura, TotalFactura
FROM Facturas
WHERE Fornecedor IN (SELECT IDFornecedor
                     FROM Fornecedores
                     WHERE Localidade = 'Lisboa')
ORDER BY DataFactura

```

Este exemplo utiliza uma subconsulta para devolver os identificadores dos fornecedores cuja localidade é Lisboa. A seguir, apresenta-se o resultado da subconsulta.

	IDFornecedor
1	1
2	2
3	3
4	4
5	6
6	34
7	82
8	121

Depois, o resultado da subconsulta é utilizado, como uma lista de valores para o operador *IN*, na cláusula *WHERE* da consulta exterior, de modo a serem incluídos no resultado da consulta apenas as facturas desses fornecedores.

A seguir, apresenta-se o resultado final obtido.

	NúmeroFactura	DataFactura	TotalFactura
1	QP58872	2006-02-25 00:00:00	116,54
2	Q545443	2006-03-14 00:00:00	1083,58
3	C73-24	2006-04-17 00:00:00	600,00
4	97/553B	2006-04-26 00:00:00	313,55
5	97/553	2006-04-27 00:00:00	904,14
6	97/522	2006-04-30 00:00:00	1962,13

A consulta 1.2 pode ser expressa através de uma junção interna do seguinte modo.

1.3 Consulta que utiliza uma junção interna para devolver o número, a data e o montante total das facturas dos fornecedores de Lisboa.

```
SELECT NúmeroFactura, DataFactura, TotalFactura
FROM Facturas INNER JOIN Fornecedores
ON Facturas.Fornecedor = Fornecedores.IDFornecedor
WHERE Localidade = 'Lisboa'
```

Esta consulta utiliza uma junção interna entre as tabelas *Facturas* e *Fornecedores*, de modo a devolver as facturas dos fornecedores cuja localidade é Lisboa. O resultado desta consulta é igual ao resultado da consulta 1.2.

Note que o resultado de uma operação de junção pode incluir colunas de ambas as tabelas. Contudo, o resultado de uma consulta que inclua uma subconsulta apenas pode conter colunas da tabela fonte da consulta exterior.

Portanto, se pretender obter também o nome dos fornecedores, terá que optar pela consulta 1.3, e não pela consulta 1.2.

Operadores *ALL*, *SOME* e *ANY*

Como verificado anteriormente, pode utilizar um operador relacional numa condição de pesquisa especificada, por exemplo, na cláusula *WHERE* de uma consulta, de modo a comparar uma expressão com o resultado devolvido por uma subconsulta.

Se não for incluída uma das palavras-chave *ALL*, *SOME* ou *ANY* na condição de pesquisa, então a subconsulta tem que devolver um único valor. Todavia, se alguma dessas palavras-chave for incluída, a subconsulta pode devolver uma lista de valores.

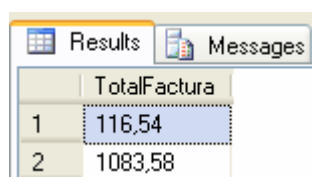
Pode utilizar a palavra-chave *ALL* para testar se uma condição de pesquisa é verdadeira para todos os valores devolvidos por uma subconsulta. Isto é equivalente a formular uma série de condições ligadas pelo operador *AND*.

Se a subconsulta não devolver nenhum registo, então uma comparação que utilize a palavra-chave *ALL* é sempre verdadeira. Por outro lado, se todos os registos devolvidos por uma subconsulta contiverem valores nulos, uma comparação que utilize a palavra-chave *ALL* é sempre falsa.

1.4 Consulta que devolve as facturas cujo total é superior ao valor da factura de mais dispendiosa do fornecedor com o identificador 34.

```
SELECT Nome, NúmeroFactura, TotalFactura
FROM Facturas INNER JOIN Fornecedores
    ON Facturas.Fornecedor = Fornecedores.IDFornecedor
WHERE TotalFactura > ALL (SELECT TotalFactura
    FROM Facturas
    WHERE Fornecedor = 34)
ORDER BY Nome
```

Note que a subconsulta devolve o total das facturas do fornecedor com o identificador 34. A seguir, apresenta-se o resultado da subconsulta.



	TotalFactura
1	116,54
2	1083,58

Depois, a consulta externa devolve as facturas cujo total é superior a todos os valores devolvidos pela subconsulta. Por outras palavras, a consulta externa devolve as facturas cujo total é superior ao maior valor devolvido pela subconsulta.

A seguir, apresenta-se o resultado final obtido.

	Nome	NúmeroFactura	TotalFactura
1	Cartografia do Minho	P-0608	20551,18
2	Cartografia do Minho	P-0259	26881,40
3	Mc Graw Hill	97/522	1962,13
4	Patinter	989319-497	2312,20
5	Patinter	989319-487	1927,54
6	Patinter	989319-477	2184,11
7	Patinter	989319-467	2318,03
8	Patinter	989319-457	3813,33
9	Patinter	989319-447	3689,99
10	Patinter	989319-437	2765,36
11	Patinter	989319-427	2115,81

A palavra-chave *ALL* é, normalmente, usada com os operadores relacionais $<$, $>$, \leq e \geq . Se utilizar o operador relacional $>$, a expressão deve ser superior ao valor máximo devolvido pela subconsulta. Contrariamente, se utilizar o operador $<$, a expressão deve ser inferior ao valor mínimo devolvido pela subconsulta. Se utilizar o operador $=$, então a expressão deve ser igual a todos os valores devolvidos pela subconsulta. Se utilizar o operador $<>$, então a expressão não deve ser igual a qualquer dos valores devolvidos pela subconsulta. Note que uma condição que utilize o operador $<>$ pode ser refeita usando o "operador" *NOT IN*.

Assim, podemos reescrever a condição de pesquisa da consulta exterior do seguinte modo:

```
WHERE TotalFactura > (SELECT MAX(TotalFactura)
FROM FActuras
WHERE Fornecedor = 34)
```

Pode utilizar, de forma indistinta, a palavra-chave *SOME* ou a palavra-chave *ANY*, para testar se uma condição de pesquisa é verdadeira para um ou mais valores devolvidos por uma subconsulta. Isto é equivalente a formular uma série de condições ligadas pelo operador *OR*. Note que a palavra-chave *SOME* é a definida na norma ANSI.

Se não forem devolvidos nenhuns registos ou se todos os registos devolvidos pela subconsulta contiverem valores nulos, uma comparação que utilize as palavras-chave *SOME* ou *ANY* é sempre falsa.

Se utilizar o operador relacional $>$, a expressão deve ser superior a pelo menos um dos valores devolvidos pela subconsulta. Isto significa que a expressão deve ser superior ao valor mínimo devolvido pela subconsulta. Se utilizar o operador $<$, a expressão deve ser inferior a pelo menos um dos valores devolvidos pela subconsulta. Isto significa que a expressão deve ser inferior ao valor máximo devolvido pela subconsulta. Se utilizar o operador $=$, então a expressão deve ser igual a um dos os valores devolvidos pela subconsulta.

Subconsultas correlacionadas

Uma subconsulta correlacionada depende da consulta exterior. Consequentemente, Uma subconsulta correlacionada é executada uma vez para cada registo processado pela consulta exterior. A consulta exterior fornece os valores necessários à subconsulta.

O sentido de execução de uma consulta que contenha uma subconsulta correlacionada é efectuado de fora para dentro.

Note que, se utilizar os operadores lógicos e/ou relacionais na condição de pesquisa da consulta exterior, o resultado devolvido pela subconsulta correlacionada deverá devolver apenas um único registo, de modo a que a comparação possa ser efectuada.

Resumindo, uma subconsulta correlacionada depende da consulta exterior e, por isso, é executada uma vez para cada registo processado pela consulta exterior. O sentido de execução de uma consulta que contenha uma subconsulta correlacionada é efectuado de fora para dentro.

1.5 Consulta que utiliza uma subconsulta correlacionada e que devolve as facturas que apresentam um total superior ao montante médio das facturas de cada fornecedor. A consulta deve ainda devolver o nome do fornecedor, bem como o número e o total das facturas e deve ser ordenada pelo nome dos fornecedores e pelo total das facturas.

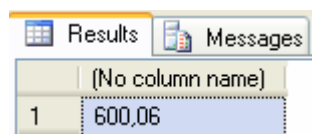
```
SELECT Fornecedor, NúmeroFactura, TotalFactura
FROM Facturas AS FacturasExterior
WHERE TotalFactura > (SELECT AVG(TotalFactura)
                      FROM Facturas AS FacturasInterior
                      WHERE FacturasInterior.Fornecedor =
                          FacturasExterior.Fornecedor)
ORDER BY Fornecedor, TotalFactura
```

Note que a condição de pesquisa da subconsulta se refere ao valor do campo *Fornecedor* da tabela *Facturas* da consulta exterior. Deste modo, apenas as facturas relativas ao vendedor incluído no registo da consulta exterior que

está a ser processado serão envolvidas no cálculo da média. Por outras palavras, para cada registo processado da consulta exterior, o valor do campo *Fornecedor* vai ser utilizado, na subconsulta, no local onde é feita a referência à coluna *Fornecedor* da consulta exterior. Por exemplo, se o valor do campo *Fornecedor* do registo a ser processado pela consulta exterior for 34, a subconsulta correlacionada será executada do seguinte modo:

```
SELECT AVG(TotalFactura)
FROM Facturas AS FacturasInterior
WHERE FacturasInterior.Fornecedor = 34
```

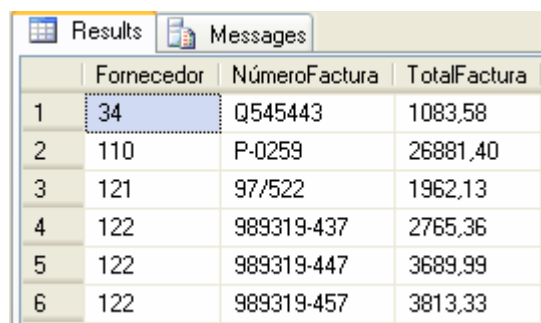
O valor devolvido pela subconsulta será o seguinte:



(No column name)
1 600,06

O valor devolvido pela subconsulta vai determinar se a factura relativa ao registo da consulta exterior que está a ser processado vai ser, ou não, incluída no resultado final da consulta. Neste caso, apenas as facturas do fornecedor com o identificador 34 que tenham um total superior a 600,06 euros irão ser incluídas no resultado da consulta. Note que tem que ser efectuado um processamento análogo para todas as facturas armazenadas na tabela *Facturas*.

A seguir, apresenta-se o resultado final obtido.



	Fornecedor	NúmeroFactura	TotalFactura
1	34	Q545443	1083,58
2	110	P-0259	26881,40
3	121	97/522	1962,13
4	122	989319-437	2765,36
5	122	989319-447	3689,99
6	122	989319-457	3813,33

Resumindo, de modo simplista, para cada factura da consulta exterior vai ser calculada a o montante médio das facturas do fornecedor dessa factura. Depois, caso o valor total da factura a ser processada seja superior ao montante médio previamente calculado, a factura é incluída no resultado final da consulta. Caso contrário, a factura é descartada do resultado final da consulta.

Note que numa subconsulta correlacionada para efectuar a referência a uma coluna da consulta exterior se utiliza a sintaxe *NomeTabelaExterior.NomeDaColuna*. No entanto, se a subconsulta correlacionada utilizar a mesma tabela que é usada pela consulta exterior, então podem ser atribuídos *alias* aos nomes das respectivas tabelas, de modo a evitar ambiguidade na distinção das colunas.

Operador *EXISTS*

Pode utilizar o operador *EXISTS* para verificar se são, ou não, devolvidos, pela subconsulta, um ou mais registos. Quando se utiliza o operador *EXISTS* com uma subconsulta, na realidade a subconsulta não devolve nenhum registo. Em vez disso, a subconsulta devolve uma indicação sobre se algum dos registos verifica a condição especificada. Assim, uma consulta que utilize este operador, pode ser executada mais rapidamente.

Dado que não é devolvido nenhum registo pela subconsulta, não é relevante saber quais as colunas que vão ser especificadas na cláusula *SELECT* da subconsulta. Normalmente, utiliza-se o ***.

Embora seja possível utilizar o operador *EXISTS* com subconsultas correlacionadas e não correlacionadas, ele é comumente usado com subconsultas correlacionadas, pois é, normalmente, melhor utilizar uma junção do que usar uma subconsulta não correlacionada com o operador *EXISTS*.

1.6 Consulta que devolve os fornecedores que não têm facturas.

```
SELECT IDFornecedor, Nome, Localidade
FROM Fornecedores
WHERE NOT EXISTS (SELECT *
                  FROM Facturas
                  WHERE Fornecedor = Fornecedores.IDFornecedor)
```

Neste exemplo, a subconsulta correlacionada selecciona todas as facturas que têm o mesmo fornecedor que está a ser processado pela consulta exterior. Depois da subconsulta ser executada, a condição de pesquisa da cláusula *WHERE* da consulta exterior utiliza a combinação de operadores *NOT EXISTS* para verificar se foi encontrada alguma factura para o fornecedor corrente. Se não, o registo relativo a esse fornecedor é incluído no resultado da consulta. Caso contrário, não é incluído.

Subconsultas incluídas na cláusula *FROM* de uma consulta

Note que, apesar de nos exemplos anteriores sobre subconsultas não correlacionadas e correlacionadas se ter utilizado sempre a cláusula *WHERE* da consulta exterior para incluir a subconsulta, também se aplicam os mesmos princípios para utilizar a subconsulta na cláusula *HAVING* da consulta exterior.

Embora as subconsultas sejam utilizadas comumente nas cláusulas *WHERE* ou *HAVING* de uma consulta, também é possível usá-las nas cláusulas *FROM* e *SELECT*.

Uma subconsulta que é incluída na cláusula *FROM* de uma consulta exterior pode devolver um qualquer conjunto de registos e de colunas. Ao resultado de subconsulta dá-se, comumente, o nome de tabela derivada. Quando se cria uma tabela derivada, deve-se associar-lhe um *alias*, de modo a que seja possível utilizar a tabela derivada na consulta exterior.

Quando se inclui uma subconsulta na cláusula *FROM* de uma consulta exterior, deve associar *alias*es a todos os valores calculados do resultado da subconsulta.

As tabelas derivadas são úteis quando necessitamos de sumariar ainda mais o resultado de uma consulta sumário.

1.7 Consulta que utiliza uma tabela derivada para devolver os cinco fornecedores que apresentam os maiores montantes médios das facturas.

```

SELECT Facturas.Fornecedor, MAX(DataFactura) AS [Última Factura],
      AVG(TotalFactura) AS [Montante médio das facturas]
FROM Facturas JOIN
      (SELECT TOP 5 Fornecedor,
      AVG(TotalFactura) AS [Montante médio das facturas]
      FROM Facturas
      GROUP BY Fornecedor
      ORDER BY [Montante médio das facturas] DESC) AS [Melhor Fornecedor]
      ON Facturas.Fornecedor = [Melhor Fornecedor].Fornecedor
GROUP BY Facturas.Fornecedor
ORDER BY [Última Factura] DESC

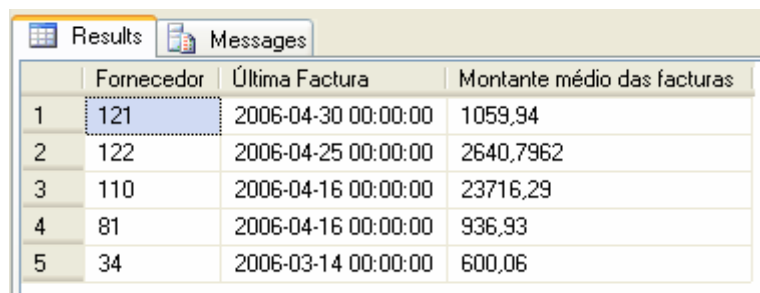
```

A subconsulta deste exemplo cria uma tabela derivada que contém o identificador e o montante médio das facturas dos fornecedores com os cinco maiores montantes médios de facturação. Para tal, agrupa as facturas por fornecedor, ordena os registos (fornecedores) por ordem decrescente do montante médio das suas facturas e devolve os cinco fornecedores com maiores montantes médios de facturação. O resultado (tabela derivada) devolvido pela subconsulta é o seguinte.

	Fornecedor	Montante médio das facturas
1	110	23716,29
2	122	2640,7962
3	121	1059,94
4	81	936,93
5	34	600,06

Depois, é efectuada a operação de junção interna entre a tabela derivada e a tabela *Facturas*. Os registos resultantes desta junção são depois agrupados por fornecedor. Finalmente, determina-se a data mais recente e calcula-se, novamente, o montante médio das facturas dos fornecedores agrupados anteriormente. O resultado final é ordenado por ordem decrescente das datas das facturas atrás determinadas.

A seguir, apresenta-se o resultado final obtido.



	Fornecedor	Última Factura	Montante médio das facturas
1	121	2006-04-30 00:00:00	1059,94
2	122	2006-04-25 00:00:00	2640,7962
3	110	2006-04-16 00:00:00	23716,29
4	81	2006-04-16 00:00:00	936,93
5	34	2006-03-14 00:00:00	600,06

Relativamente à consulta anterior, deve ter em atenção quatro considerações. Primeira, foi atribuído um *alias* à tabela derivada, de modo a que possa ser referenciada na consulta exterior. Segunda, também foi associado um *alias* ao resultado da função de agregação *AVG* usada na subconsulta, pois uma tabela derivada não pode conter colunas sem nome. Terceira, foi incluída uma cláusula *ORDER BY* na subconsulta, uma vez que foi usada a palavra-chave *TOP*. Quarta, foi necessário incluir a função de agregação *AVG* também na cláusula *SELECT* da consulta exterior, pois foi utilizada uma cláusula *GROUP BY* na consulta exterior. Deste modo, apenas podem ser incluídas na cláusula *SELECT* funções de agregação, nomes de colunas que são utilizadas na cláusula *GROUP BY*, ou colunas com valores fixos.

Note que, quando usada na cláusula *FROM*, uma subconsulta é semelhante a uma vista, pois devolve os registos e colunas de uma ou mais tabelas. Assim, as tabelas derivadas podem ser úteis para testar possíveis soluções para criar vistas. Depois de a tabela derivada funcionar correctamente, pode definir a vista baseando-se na subconsulta utilizada para criar a tabela derivada.

Subconsultas incluídas na cláusula *SELECT* de uma consulta

Quando se inclui uma subconsulta numa cláusula *SELECT* de uma consulta exterior, a subconsulta deve devolver um valor unitário.

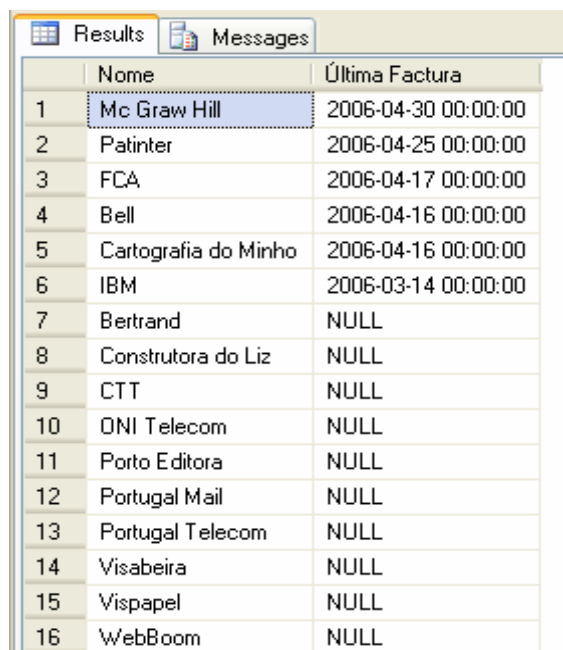
Na maioria dos casos, as subconsultas incluídas na cláusula *SELECT* de uma consulta exterior são subconsultas correlacionadas.

1.8 Consulta que utiliza uma subconsulta correlacionada na sua cláusula *SELECT* para devolver a factura mais recente de cada fornecedor.

```
SELECT DISTINCT Nome, (SELECT MAX(DataFactura)
                        FROM Facturas
                        WHERE Facturas.Fornecedor =
                           Fornecedores.IDFornecedor) AS [Última Factura]
FROM Fornecedores
ORDER BY [Última Factura] DESC
```

A subconsulta deste exemplo calcula a data mais recente das facturas de cada fornecedor. Para tal, refere-se ao campo *IDFornecedor* da tabela *Fornecedores* usada na consulta exterior.

A seguir, apresenta-se o resultado final obtido.



	Nome	Última Factura
1	Mc Graw Hill	2006-04-30 00:00:00
2	Patinter	2006-04-25 00:00:00
3	FCA	2006-04-17 00:00:00
4	Bell	2006-04-16 00:00:00
5	Cartografia do Minho	2006-04-16 00:00:00
6	IBM	2006-03-14 00:00:00
7	Bertrand	NULL
8	Construtora do Liz	NULL
9	CTT	NULL
10	ONI Telecom	NULL
11	Porto Editora	NULL
12	Portugal Mail	NULL
13	Portugal Telecom	NULL
14	Visabeira	NULL
15	Visapel	NULL
16	WebBoom	NULL

Uma consulta que inclui uma subconsulta na sua cláusula *SELECT* pode, na maioria dos casos, ser reescrita usando uma operação de junção, em vez de utilizar a subconsulta.

1.9 Consulta, semelhante à consulta anterior, que utiliza uma operação de junção para devolver a factura mais recente de cada fornecedor.

```
SELECT Nome, MAX(DataFactura) AS [Última Factura]
FROM Fornecedores LEFT OUTER JOIN Facturas
    ON Fornecedores.IDFornecedor = Facturas.Fornecedor
GROUP BY Nome
ORDER BY [Última Factura] DESC
```

Esta consulta devolve o mesmo resultado da consulta anterior. Dado que a operação de junção é normalmente mais rápida e mais fácil de ler, as subconsultas são raramente utilizadas na cláusula *SELECT* de uma consulta.

1.10 Consulta que devolve, para cada localidade, o fornecedor que apresenta o maior montante dispendido.

```
SELECT Sumário1.Localidade, Sumário1.Nome,
    TopFactLocalidade.SomaFacturas
FROM
    (SELECT Fo_Sub.Localidade, Fo_Sub.Nome,
        SUM(Fa_Sub.TotalFactura) AS SomaFacturas
    FROM Facturas AS Fa_Sub INNER JOIN Fornecedores AS Fo_Sub
        ON Fa_Sub.Fornecedor = Fo_Sub.IDFornecedor
    GROUP BY Fo_Sub.Localidade, Fo_Sub.Nome) AS Sumário1
INNER JOIN
    (SELECT Sumário2.Localidade,
        MAX(Sumário2.SomaFacturas) AS SomaFacturas
    FROM
        (SELECT Fo_Sub.Localidade, Fo_Sub.Nome,
            SUM(Fa_Sub.TotalFactura) AS SomaFacturas
        FROM Facturas AS Fa_Sub JOIN Fornecedores AS Fo_Sub
            ON Fa_Sub.Fornecedor = Fo_Sub.IDFornecedor
        GROUP BY Fo_Sub.Localidade, Fo_Sub.Nome) AS Sumário2
    GROUP BY Sumário2.Localidade) AS TopFactLocalidade
    ON Sumário1.Localidade = TopFactLocalidade.Localidade AND
        Sumário1.SomaFacturas = TopFactLocalidade.SomaFacturas
ORDER BY Sumário1.Localidade
```

A consulta deste exemplo utiliza três subconsultas: *Sumário1*, *Sumário2* e *TopFactLocalidade*. As subconsultas *Sumário1* e *TopFactLocalidade* são alvo de uma operação de junção interna na cláusula *WHERE* da consulta exterior.

A subconsulta *Sumário2* está aninhada na cláusula *FROM* da subconsulta *TopFactLocalidade*.

As subconsultas *Sumário1* e *Sumário2* são idênticas. Relacionam dados das tabelas *Fornecedores* e *Facturas* e produzem um resultado que inclui a soma do valor total das facturas para cada fornecedor agrupado pela localidade e pelo nome.

A subconsulta *TopFactLocalidade* produz um resultado que inclui a localidade e a soma de maior valor do total das facturas dos fornecedores dessa localidade.

As colunas especificadas na cláusula *SELECT* da consulta externa são derivadas a partir do resultado da junção entre as subconsultas *Sumário1* e *TopFactLocalidade*, e o resultado final é ordenado pela localidade.

A seguir, apresenta-se o código das subconsultas *Sumário1* e *Sumário2*.

```
SELECT Fo_Sub.Localidade, Fo_Sub.Nome,
       SUM(Fa_Sub.TotalFactura) AS SomaFacturas
FROM Facturas AS Fa_Sub INNER JOIN Fornecedores AS Fo_Sub
     ON Fa_Sub.Fornecedor = Fo_Sub.IDFornecedor
GROUP BY Fo_Sub.Localidade, Fo_Sub.Nome
```

A seguir apresenta-se o resultado da execução das subconsultas *Sumário1* e *Sumário2*.

	Localidade	Nome	SomaFacturas
1	Setubal	Bell	936,93
2	Braga	Cartografia do Minho	47432,58
3	Lisboa	FCA	600,00
4	Lisboa	IBM	1200,12
5	Lisboa	Mc Graw Hill	3179,82
6	Viseu	Patinter	21126,37

A seguir apresenta-se o código da subconsulta *TopFactLocalidade*.

```

SELECT Sumário2.Localidade,
       MAX(Sumário2.SomaFacturas) AS SomaFacturas
FROM (SELECT Fo_Sub.Localidade, Fo_Sub.Nome,
            SUM(Fa_Sub.TotalFactura) AS SomaFacturas
      FROM Facturas AS Fa_Sub JOIN Fornecedores AS Fo_Sub
      ON Fa_Sub.Fornecedor = Fo_Sub.IDFornecedor
      GROUP BY Fo_Sub.Localidade, Fo_Sub.Nome) AS Sumário2
GROUP BY Sumário2.Localidade

```

A seguir apresenta-se o resultado da execução da subconsulta *TopFactLocalidade*.

	Localidade	SomaFacturas
1	Braga	47432,58
2	Lisboa	3179,82
3	Setubal	936,93
4	Viseu	21126,37

O resultado final da execução da consulta 1.10 é o seguinte:

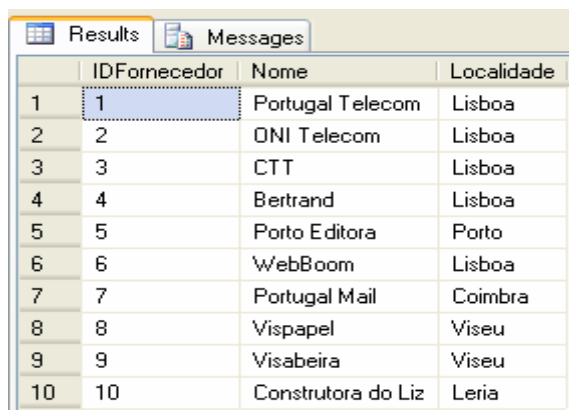
	Localidade	Nome	SomaFacturas
1	Braga	Cartografia do Minho	47432,58
2	Lisboa	Mc Graw Hill	3179,82
3	Setubal	Bell	936,93
4	Viseu	Patinter	21126,37

QUESTÕES

2 Crie as seguintes quatro consultas.

2.1 Recorrendo à utilização de subconsultas, crie uma consulta que devolva o identificador, o nome e a localidade dos fornecedores que não têm facturas registadas na base de dados.

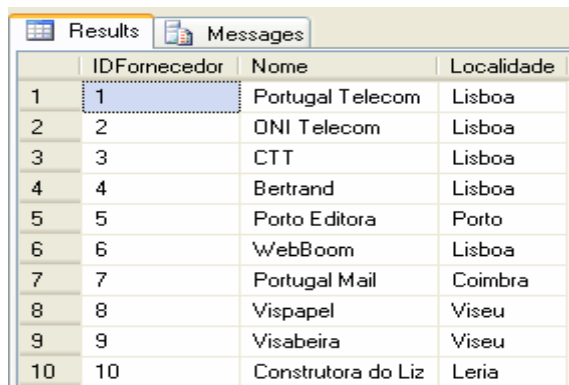
Resultado



	IDFornecedor	Nome	Localidade
1	1	Portugal Telecom	Lisboa
2	2	ONI Telecom	Lisboa
3	3	CTT	Lisboa
4	4	Bertrand	Lisboa
5	5	Porto Editora	Porto
6	6	WebBoom	Lisboa
7	7	Portugal Mail	Coimbra
8	8	Vispapel	Viseu
9	9	Visabeira	Viseu
10	10	Construtora do Liz	Leria

2.2 Crie uma consulta que devolva o identificador, o nome e a localidade dos fornecedores que não têm facturas registadas na base de dados. Contudo, para esta nova consulta utilize uma junção, em vez de uma subconsulta. Para construir a consulta, não utilize o operador *EXISTS*.

Naturalmente, o resultado obtido é igual ao resultado da execução da consulta 2.1.



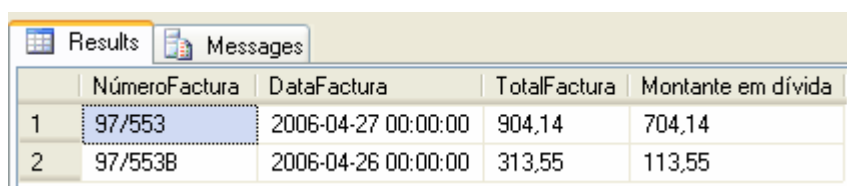
	IDFornecedor	Nome	Localidade
1	1	Portugal Telecom	Lisboa
2	2	ONI Telecom	Lisboa
3	3	CTT	Lisboa
4	4	Bertrand	Lisboa
5	5	Porto Editora	Porto
6	6	WebBoom	Lisboa
7	7	Portugal Mail	Coimbra
8	8	Vispapel	Viseu
9	9	Visabeira	Viseu
10	10	Construtora do Liz	Leria

Note que uma consulta que utiliza uma subconsulta na cláusula *WHERE* para “criar” a lista de valores do operador *IN*, pode ser implementada através de uma junção interna. Do mesmo modo, uma consulta que utiliza uma subconsulta na cláusula *WHERE* para “criar” a lista de valores do operador *NOT IN*, pode ser implementada através de uma junção externa.

Portanto, pode optar por qualquer das abordagens para formular a consulta. Normalmente, há que ter em consideração a preferência do programador e, principalmente a performance da execução da consulta. Como existem vários factores que podem contribuir para essa performance, aconselha-se, em casos particulares, a construção da consulta nas duas vertentes e a comparação do tempo de execução de ambas.

2.3 Crie uma consulta que devolva o número, a data, o total e o montante em dívida das facturas por pagar cujo montante em dívida é inferior ao valor médio das facturas que ainda não foram pagas. O resultado deve ser ordenado, por forma decrescente do total das facturas.

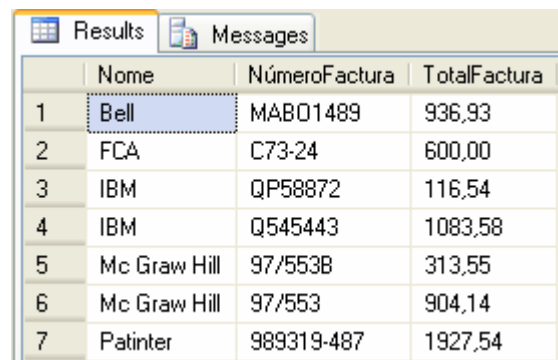
Resultado



	NúmeroFactura	DataFactura	TotalFactura	Montante em dívida
1	97/553	2006-04-27 00:00:00	904,14	704,14
2	97/553B	2006-04-26 00:00:00	313,55	113,55

- 2.4 Crie uma consulta que devolva o nome dos fornecedores cujas facturas apresentam um total inferior ao montante de pelo menos uma das facturas do fornecedor com o nome *Mc Graw Hill*. A consulta deve ainda devolver o número e o total das facturas e deve ser ordenada pelo nome dos fornecedores.

Resultado



	Nome	NúmeroFactura	TotalFactura
1	Bell	MAB01489	936,93
2	FCA	C73-24	600,00
3	IBM	QP58872	116,54
4	IBM	Q545443	1083,58
5	Mc Graw Hill	97/553B	313,55
6	Mc Graw Hill	97/553	904,14
7	Patinter	989319-487	1927,54