

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

Съставяне на подзаявки

SQL предоставя възможност за влагане на заявки една в друга. Например:

1) Заявка, извеждаща данните за продуктите с минимална доставна цена.

```
SELECT * FROM Products
```

```
WHERE price = ( SELECT MIN(price) FROM Products );
```

2) Заявка, извеждаща данните за продуктите с доставна цена, по-голяма от средната доставна цена на всички продукти.

```
SELECT * FROM Products
```

```
WHERE price > ( SELECT AVG(price) FROM Products );
```

В тези примери трябва подзаявката да връща една стойност – един ред и една колона, в противен случай се получава съобщение за грешка. За да се осигури връщане на точно една стойност, се използват обобщаващи функции или условие за стойността на колоната (колони) на първичния ключ или на колона (колони) с ограничение за уникалност. Когато подзаявката не върне резултат, конструкцията няма да върне грешка, но и основната заявка няма да върне резултат.

Разширение на стандартния SQL позволява извеждане на резултата от подзаявката в списъка с полета на главната заявка. Например:

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

3) Заявка, извеждаща име на продукт и брой на продажбите на съответния продукт.

```
SELECT ProductName, ( SELECT COUNT(*)
```

```
FROM SaleDetails sd
```

```
WHERE sd.ProductID = p.ProductID )
```

```
AS CountOfProductsSales
```

```
FROM Products p;
```

Тази заявка е еквивалентна на следната:

```
SELECT ProductName,
```

```
COUNT(sd.ProductID) AS CountOfProductsSales
```

```
FROM Products p
```

```
LEFT JOIN SaleDetails sd ON sd.ProductID = p.ProductID
```

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

GROUP BY ProductName;

За подзаявки, генериращи произволен брой редове, се използва специалния оператор **I**
N

.
Например
:

4) Заявка, извеждаща данните за продажбите на служителите с дадена фамилия.

```
SELECT *
```

```
FROM Sales
```

```
WHERE EmployeeID IN ( SELECT EmployeeID FROM Employees
```

```
WHERE LastName = "Име" );
```

Тази заявка е еквивалентна на следната:

```
SELECT s.*
```

```
FROM Sales s
```

```
INNER JOIN Employees e ON e.EmployeeID = s.EmployeeID
```

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

```
WHERE LastName = "Име";
```

Едно съединение може да се напише като подзаявка, но обратното не винаги е вярно. Съединението в повечето случаи се изпълнява по-бързо, отколкото еквивалентната подзаявка, но целта на заявката е по-ясно определена, когато даден въпрос бъде решен чрез подзаявка.

5) Заявка, извеждаща данните за клиентите, намиращи се в градове, в които има доставчици.

```
SELECT * FROM Customers
```

```
WHERE City IN ( SELECT DISTINCT City FROM Suppliers );
```

6) Заявка, извеждаща идентификаторите и имената на продуктите, които не са продавани през изминалия месец и продажбата им не е преустановена.

```
SELECT ProductID, ProductName
```

```
FROM Products
```

```
WHERE ProductID NOT IN
```

```
( SELECT DISTINCT ProductID
```

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

FROM SaleDetails sd

INNER JOIN Sales s ON sd.SaleID = s.SaleID

WHERE SaleDate BETWEEN

DATEADD("m", -1, Date()) AND Date())

AND Discontinued = 0;

7) За извеждане на първите три най-високи цени и съответните имена на продукти, ще е необходимо да се използва DISTINCT с TOP. Ако едновременно с това бъде избрано и името на съответния продукт, няма да се получи желанния резултат, защото DISTINCT се прилага върху целия резултатен ред и всяка комбинация ProductName и Price е уникална за таблицата Products. Затова се използва подзаявка, за да се определи кои редове имат една от трите най-високи цени:

SELECT ProductID, ProductName, Price

FROM Products

WHERE price IN (SELECT DISTINCT TOP 3 price

FROM Products

ORDER BY price DESC)

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

ORDER BY price DESC;

В този случай редовете се връщат подредени в низходящ ред по цена, тъй като редът се контролира от външната заявка, към която е добавен ORDER BY.

Използване на оператори за подзаявки – EXISTS, ANY(или SOME), ALL

EXISTS е оператор, който приема подзаявка като аргумент и връща стойност TRUE, ако тази подзаявка връща някакви изходни данни и стойност FALSE – в противен случай. Този оператор не може да върне стойност UNKNOWN. За подзаявките, които използват EXISTS, няма значение коя колона избира EXISTS или дали избира всички колони, защото този оператор просто регистрира дали има някакви изходни данни от подзаявката или не и изобщо не използва генерираните стойности. Например:

1) Заявка, извеждаща идентификаторите и имената на продуктите, които не са продавани и продажбата им не е преустановена.

```
SELECT p.ProductID, p.ProductName
```

```
FROM Products p
```

```
WHERE NOT EXISTS (SELECT * FROM SaleDetails sd
```

```
WHERE sd.ProductID = p.ProductID)
```

```
AND p.Discontinued = 0;
```

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

Тази заявка е еквивалентна на следната:

```
SELECT p.ProductID, p.ProductName
```

```
FROM Products p
```

```
LEFT JOIN SaleDetails sd ON sd.ProductID = p.ProductID
```

```
WHERE sd.ProductID IS NULL AND p.Discontinued = 0;
```

2) Заявка, извеждаща имената на продуктите, които са продавани.

```
SELECT p.ProductName
```

```
FROM Products p
```

```
WHERE EXISTS (SELECT * FROM SaleDetails sd
```

```
WHERE sd.ProductID = p.ProductID);
```

Тази заявка е еквивалентна на следната:

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

```
SELECT DISTINCT p.ProductName
```

```
FROM Products p
```

```
INNER JOIN SaleDetails sd ON sd.ProductID = p.ProductID;
```

ANY (или **SOME**) връща стойност TRUE, ако някоя стойност, избрана от подзаявката, удовлетворява условието на външната заявка. Например:

3) Заявка, извеждаща данните за клиентите, намиращи се в градове, в които има доставчици.

```
SELECT * FROM Customers
```

```
WHERE City = ANY (SELECT DISTINCT City FROM Suppliers);
```

Тази заявка е еквивалентна на следната:

```
SELECT * FROM Customers
```

```
WHERE City IN (SELECT DISTINCT City FROM Suppliers);
```

4) Заявка, извеждаща продуктите с доставна цена, по-голяма от доставната цена на някой (кой да е) продукт от дадена по името си категория.

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

```
SELECT p.ProductID, p.ProductName, p.Price
```

```
FROM Products p
```

```
WHERE p.price >
```

```
ANY (SELECT DISTINCT price FROM Products p
```

```
INNER JOIN categories c
```

```
ON p.CategoryID = c.CategoryID
```

```
WHERE c.CategoryName = "name")
```

Тази заявка е еквивалентна на следната:

```
SELECT p.ProductID, p.ProductName, p.Price
```

```
FROM Products p
```

```
WHERE p.price >
```

```
(SELECT MIN(price) FROM Products p
```

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

INNER JOIN categories c

ON p.CategoryID = c.CategoryID

WHERE c.CategoryName = "name");

Когато подзаявката не генерира изходни данни, ANY връща FALSE.

ALL връща стойност TRUE, ако всяка стойност, избрана от подзаявката, удовлетворява условието на външната заявка. Например:

5) Заявка, извеждаща продуктите с доставна цена по-голяма от доставната цена на всеки продукт от дадена по името си категория.

SELECT p.ProductID, p.ProductName, p.Price

FROM Products p

WHERE p.price >

ALL (SELECT DISTINCT price FROM Products p

INNER JOIN categories c

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

```
ON p.CategoryID = c.CategoryID
```

```
WHERE c.CategoryName = "name")
```

Тази заявка не е еквивалентна на следната:

```
SELECT p.ProductID, p.ProductName, p.Price
```

```
FROM Products p
```

```
WHERE p.price >
```

```
(SELECT MAX(price) FROM Products p
```

```
INNER JOIN categories c
```

```
ON p.CategoryID = c.CategoryID
```

```
WHERE c.CategoryName = "name");
```

Когато подзаявката не генерира изходни данни, ALL връща TRUE. Следователно за пример 5, когато категория със зададеното име не съществува, първата заявка ще изведе данните за всички продукти от таблицата Products, докато втората (с обобщаващата функция MAX) няма да генерира никакви изходни данни. Освен това ако

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

има продукт от зададената категория, на който не е въведена доставна цена (т.е. price има стойност NULL), първата заявка няма да върне резултат, за разлика от втората, която може и да върне някакви изходни данни.

6) Заявка, извеждаща идентификаторите и имената на продуктите, които не са продавани и продажбата им не е преустановена.

```
SELECT p.ProductID, p.ProductName
```

```
FROM Products p
```

```
WHERE p.ProductID < ALL (SELECT DISTINCT ProductID
```

```
FROM SaleDetails)
```

```
AND Discontinued = 0;
```

Взаимосвързани (съпоставени) заявки

Взаимосвързаните (съпоставени) заявки използват подзаявки, от които се прави обръщение към таблицата от FROM на външната (основната, главната) заявка. По този начин подзаявката се изпълнява многократно, по един път за всеки ред от таблицата в основната заявка, при което резултатите от всяко изпълнение на подзаявката трябва да бъдат съпоставени със съответния ред от външната заявка. Взаимосвързаните заявки се използват за сравняване на определени редове от една таблица с условие от съответстващата таблица. Например:

1) Заявка, извеждаща всички редове от SaleDetails, за които продажната цена на

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

съответния продукт е по-голяма или равна на средната аритметична стойност на продажните цени на този продукт с поне 50%.

```
SELECT * FROM SaleDetails sd
```

```
WHERE price >= (SELECT 1.5*AVG(price)
```

```
FROM SaleDetails sd1
```

```
WHERE sd.ProductID = sd1.ProductID);
```

2) Заявка, извеждаща данните за клиентите, пазарували на дадена дата (21.08.2003г.).

```
SELECT * FROM Customers c
```

```
WHERE #21/08/2003# IN
```

```
(SELECT DISTINCT DATEVALUE(SaleDate)
```

```
FROM Sales s
```

```
WHERE c.CustomerID = s.CustomerID);
```

Тази заявка е еквивалентна на следната:

Съставяне на подзаявки

Написано от
Понеделник, 06 Февруари 2012 14:51 -

```
SELECT DISTINCT c.*
```

```
FROM Customers c
```

```
INNER JOIN Sales s ON c.CustomerID = s.CustomerID
```

```
WHERE DATEDIFF("d", s.SaleDate, #2003/08/21#) = 0;
```

3) Заявка, която връща продажбите на продуктите, чиито приходи са под 75% от средните приходи на всички продажби на съответния продукт.

```
SELECT sd.SaleID, p.ProductID, p.ProductName,
```

```
sd.Price*sd.Quantity*(1-sd.Discount)
```

```
AS ExtendedPrice
```

```
FROM Products p
```

```
INNER JOIN SaleDetails sd ON p.ProductID = sd.ProductID
```

```
WHERE sd.Price*sd.Quantity*(1-sd.Discount)
```