

Усъвършенствани техники за заявки



Подзаявки



Основни акценти

- Въведение в подзаявките
- Вложени подзаявки
- Корелационни подзаявки
- Използване на ключовите думи EXISTS и NOT EXISTS

Въведение в подзаявките

Подзаявката е оператор SELECT вложен в оператор SELECT, INSERT, UPDATE или DELETE или в друга подзаявка.

- Не могат да се използват подзаявки, обръщащи се към колони, съдържащи данни тип text и image
- Ограничения в нивата на влагане на подзаявките – не повече от 32 нива; зависи от наличната памет

Къде да използваме подзаявки?

- Подзаявките се ограждат в скоби () и се използват:
 - На мястото на израз в заявка. На мястото на израза подзаявката връща
 - единична стойност
 - или списък от стойности.
 - На мястото на таблица - подзаявката връща многоколонен набор от данни

Подзаявки на мястото на израз

Подзаявката замества израз в:

- списъка за избор
- или в клаузата **WHERE** с оператор за сравнение (=, >, <, >=, <=, <>)
- Списъкът за избор на подзаявката може да съдържа само един израз или име на колона.
- Вложената подзаявка може да върне:
 - единична стойност
 - или списък от стойности

Пример на подзаявка, която връща единична стойност

Извличане на данни за стоките, които са поръчани в максимално количество (код на стоката, поръчано количество и номер на поръчката).

```
SELECT ProductId, Quantity, OrderId
FROM [Order Details]
WHERE Quantity =
    (SELECT MAX(Quantity)
     FROM [Order Details])
```

Използване на локална променлива

деклариране на локална променлива

```
DECLARE @Q Numeric(18,2)
```

-- даване стойност на локалната променлива

```
SET @Q = (SELECT MAX(Quantity)  
          FROM [Order Details])
```

-- Извличане на данни за поръчки с макс. к-во

```
SELECT ProductId, Quantity, OrderId  
FROM [Order Details]  
WHERE Quantity = @Q
```

Особености при използване на подзаявка, връщаща единична стойност

Стойността, върната от подзаявката, замества израза в списъка за избор или в клаузата WHERE

- За да върне подзаявката единична стойност в нейния списък за избор се задава:
 - име на колона;
 - агрегатна функция;
 - израз.
- Ако вложената заявка върне повече от една стойност, когато външната заявка очаква единична стойност, SQL Server извежда съобщение за грешка

Подзаявка, връщаща единична стойност

- Извличане данни за клиентите и техните поръчки, които са направени през последния ден, регистриран в таблицата "Поръчки".

```
SELECT C.CustomerID, CompanyName  
, OrderID, CONVERT(varchar(10),OrderDate,104)  
FROM Orders O JOIN Customers C  
ON O.CustomerId = C.CustomerId  
WHERE  
OrderDate = (SELECT max(OrderDate)  
FROM Orders)
```

Друг вариант с разделяне в две заявки

```
DECLARE @ORDDATE DATE
SET @ORDDATE =
    (SELECT max(OrderDate)
     FROM Orders)
SELECT C.CustomerID, CompanyName
, OrderID, CONVERT(varchar(10),OrderDate,104)
FROM Orders O JOIN Customers C
ON O.CustomerId = C.CustomerId
WHERE
    OrderDate = @ORDDATE
```

Използване на подзаявки в списъка за избор

За всяка поръчка да се изведат: номер на поръчката, код на стоката, стойността на поръчаната стока, процент на стойността на поръчаната стока от общата стойност на всички поръчки.

```
SELECT OrderID, ProductId,  
       , Quantity*UnitPrice AS Product_sold  
       , ((Quantity*UnitPrice)/  
(SELECT SUM(Quantity*UnitPrice)  
FROM [Order Details])*100) AS  
                               Percent_of_all_orders  
FROM [Order Details]
```

Използване на подзаявки в списъка за избор

```
SELECT OrderID, P.ProductId, ProductName
, Quantity*OD.UnitPrice AS Product_sold
, ((Quantity*OD.UnitPrice)/
  (SELECT SUM(Quantity*UnitPrice)
FROM [Order Details])*100) AS
  Percent_of_all_orders

FROM [Order Details] OD JOIN Products P
ON OD.ProductID = P.ProductID
```

Пример на подзаявка, връщаща списък от стойности

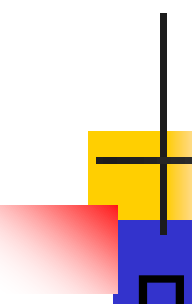
Извличане имената на клиентите, които са правили
поръчки след определена дата (1.09.1996)

```
SELECT CompanyName
FROM Customers
WHERE CustomerID IN
    (SELECT DISTINCT CustomerID
     FROM Orders
     WHERE OrderDate > '09/01/1996')
ORDER BY 1
```

Особености при използване на подзаявки, връщащи списък от стойности

Подзаявката се задава в клаузата **WHERE** на мястото на израза, връщащ списък от стойности

- За да се направи сравнение с елементите на списъка от стойности, които връща подзаявката, се използва ключовата дума **IN** .
- Всяка вложена подзаявка се изчислява само веднъж.



Заявката с вложена подзаявка може да се замени със заявка със съединяване

Пример:

```
SELECT DISTINCT C.CompanyName
FROM Customers C
INNER JOIN Orders O ON
C.CustomerID = O.CustomerID
WHERE
O.OrderDate > '09/01/1996'
```

Използване на подзаявка като извлечена таблица (derived table)

На мястото на таблица в клаузата FROM може да се зададе подзаявка, която извлича таблица.

- В подзаявката чрез ключовата дума AS се задава псевдоним на извлечената таблица.
- Чрез външната заявка се селектират данни от извлечената таблица, като колоните се задават с уточнени имена, използващи псевдонима на извлечената таблица.



Пример:

Да се извлекат данни за клиентите, номерата на техните поръчки и общата им стойност (код и име на клиента, номер на поръчката и обща стойност на поръчката).

Пример на използване на подзаявка като извлечена таблица

```
USE Northwind
```

```
GO
```

```
SELECT C.CustomerID, CompanyName  
,T.Orderid, Subtotal
```

```
FROM Customers C JOIN Orders O
```

```
ON C.CustomerID=O.CustomerID
```

```
JOIN
```

```
(SELECT OrderID, Sum(UnitPrice*Quantity) AS  
Subtotal
```

```
FROM [Order Details] GROUP BY OrderID) AS T
```

```
ON T.Orderid=O.Orderid
```

Корелационни подзаявки

Използват се когато вътрешната заявка трябва да получи информация от външната заявка преди да може да бъде изпълнена

- Трябва да се използват синонимни имена за разграничаване на таблиците
- Корелационните подзаявки обикновено могат да се заменят със заявки със съединяване. Това позволява на оптимизатора на заявки да определи как да корелира данните по по-ефективен начин.



Пример на корелационна подзаявка

Да се изведат номерата на поръчките и кодовете и имената на клиентите, закупили стоката с код 33 в количество по-голямо от 20.

Решение:

```
SELECT OrderID, O.CustomerID,  
        CompanyName  
FROM Orders  O JOIN Customers C  
ON O.CustomerId = C.CustomerId  
WHERE  
(SELECT Quantity  
FROM [Order Details] OD  
WHERE O.OrderID=OD.OrderID  
      AND OD.ProductId=33) >20
```

Заявка със съединяване, еквивалентна
на корелационната подзаявка

```
SELECT O.OrderID, O.CustomerID  
    , CompanyName  
FROM Customers C JOIN Orders O  
ON C.CustomerId=O.CustomerId  
JOIN [Order Details] OD  
ON O.OrderID=OD.OrderID  
WHERE OD.Quantity >20  
        AND OD.ProductId=33
```

Особености на корелационните подзаявки

- В корелационната подзаявка вътрешната подзаявка се изпълнява многократно - за всеки ред, който външната заявка селектира.
- Резултатите от вътрешната заявка се сравняват с критерия, зададен в клаузата WHERE на външната заявка.
- Корелационната подзаявка връща единична стойност или списък от стойности за всеки ред, специфициран в клаузата FROM на външната заявка

Стъпки при изпълнение на корелационните подзаявки

1. Външната заявка предава стойност на колоната на вътрешната заявка
 2. Вътрешната заявка използва тази стойност за да удовлетвори (изпълни) вътрешната заявка
 3. Вътрешната заявка връща стойност обратно на външната заявка
 4. Външната заявка се изпълнява върху върнатата стойност
 5. Процесът се повтаря за всяка следваща стойност на колоната от външната заявка
- Обратно към стъпка 1



Пример:

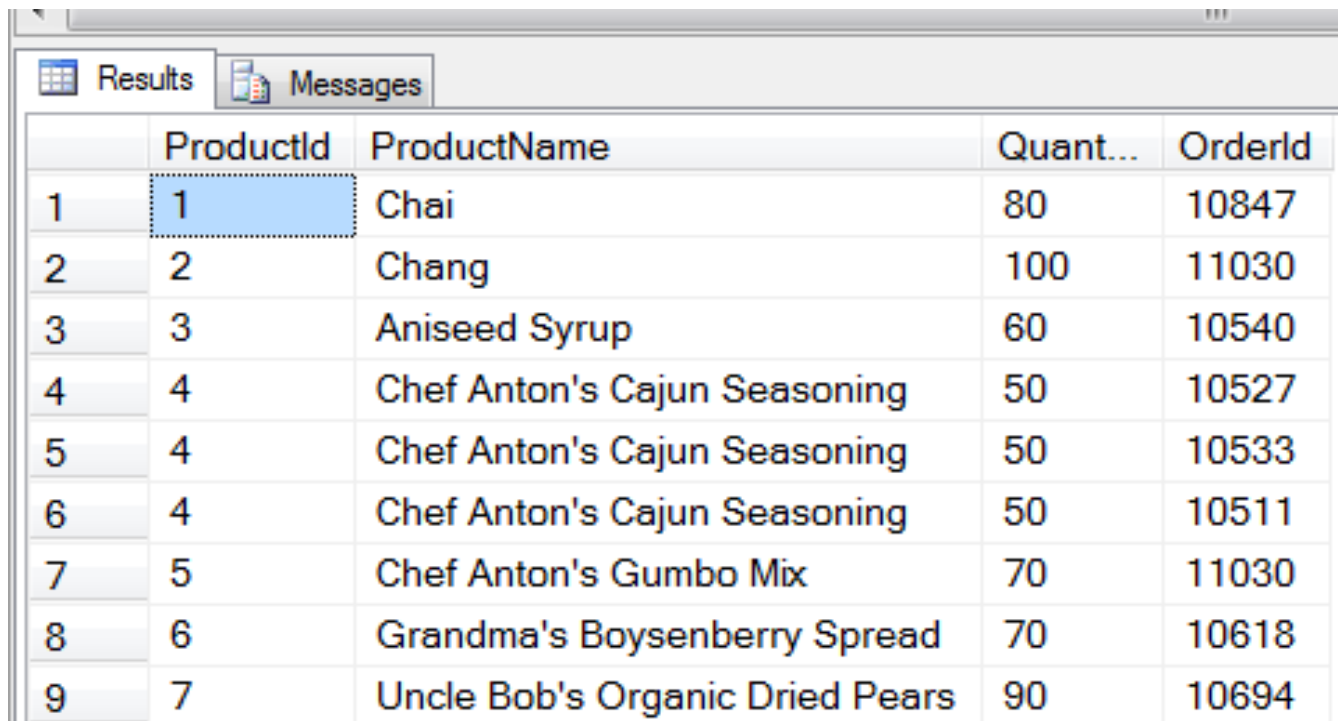
- Да се изведат данни за максимално поръчаното количество от всяка стока и с коя поръчка е поръчано.
- Решението е заявка с корелационна подзаявка, използваща копие на таблицата от външната заявка.

Пример на корелационна подзаявка,
използваща копие на една и съща таблица

```
SELECT P.ProductId, ProductName, Quantity
, OrderId
FROM Products P
      JOIN [Order Details] ORD1
      ON  P.ProductID = ORD1.ProductID
WHERE Quantity =
      (SELECT MAX(Quantity)
      FROM [Order Details] ORD2
      WHERE ORD1.ProductId=ORD2.ProductId)
```

Резултат от заявката

Извеждат се данни за максимално поръчаното количество от всяка стока и с коя поръчка е поръчано:



	ProductId	ProductName	Quant...	OrderId
1	1	Chai	80	10847
2	2	Chang	100	11030
3	3	Aniseed Syrup	60	10540
4	4	Chef Anton's Cajun Seasoning	50	10527
5	4	Chef Anton's Cajun Seasoning	50	10533
6	4	Chef Anton's Cajun Seasoning	50	10511
7	5	Chef Anton's Gumbo Mix	70	11030
8	6	Grandma's Boysenberry Spread	70	10618
9	7	Uncle Bob's Organic Dried Pears	90	10694

Използване на ключовите думи **EXISTS** и **NOT EXISTS**

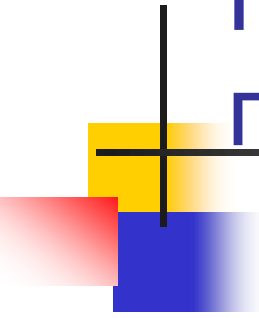
Използват се с корелационни подзаявки за да се ограничи резултатният набор от външната заявка до редовете, които удовлетворяват подзаявката.

- Синтаксис:

- WHERE EXISTS (подзаявка)
- WHERE NOT EXISTS (подзаявка)

Използване на ключовите думи **EXISTS** и **NOT EXISTS**

- Изразът **EXISTS** (подзаявка) определя дали съществуват данни в списъка от стойности, връщан от подзаявката като връща стойност TRUE или FALSE, в зависимост от това дали са върнати редове от подзаявката.
- **WHERE NOT EXISTS** (подзаявка) връща TRUE, ако не са върнати данни и FALSE в противен случай.



Пример: Кои клиентите са правили
поръчки през 1996 година

```
SELECT DISTINCT CompanyName
FROM Customers CS
WHERE EXISTS
    (SELECT * FROM Orders O WHERE
      CS.CustomerId = O.CustomerId
      AND year(OrderDate) = 1996)
```

Процес на обработка на подзаявка, използваща EXISTS или NOT EXISTS

- Външната заявка предава стойност на вътрешната заявка.
- Вътрешната заявка не произвежда никакви данни; тя връща TRUE или FALSE в зависимост от условието, дадено в нея.
- Външната заявка извежда съответния ред с данни, в зависимост от това дали резултатът от тестването е TRUE или FALSE.

Представяне на корелационната подзаявка като заявка със съединяване

```
SELECT DISTINCT CompanyName  
FROM Orders  
INNER JOIN Customers CS  
ON Orders.CustomerId = S.CustomerId  
WHERE year(OrderDate) = 1996
```


За и против подзаявките

Защо да използваме подзаявки

- За да разделим комплексните заявки в серия от логически стъпки
- За да отговорим на запитване, което се осъществява върху резултатите от друга заявка
- Заявките, съдържащи подзаявки често могат да се представят чрез съединяване на таблици
- Защо да използваме съединяване вместо подзаявки
 - SQL Server изпълнява съединяването по-бързо отколкото подзаявките

Пример, изискващ използване на подзаявка:

Да се изведат данни за стоките, които не са поръчвани през дадена година (напр. 1996)

Предложение:

```
SELECT DISTINCT P.ProductID, ProductName
FROM Products P
LEFT JOIN [Order Details] OD
    ON P.ProductID = OD.ProductID
LEFT JOIN ORDERS O
    ON OD.OrderID = O.OrderID
WHERE YEAR(OrderDate) != 1996
      OR OrderDate IS NULL
```

Защо този *SELECT* не е решение?

В резултата след съединяването на съответните редове от трите таблици се извеждат кода и името на продуктите за следните три случая:

- За продуктите изобщо няма поръчки (колоната за дата е с неопределена стойност);
- Продуктите не са поръчвани през 1996, но са поръчвани през други години;
- Продуктите са поръчвани през различни години в т.ч. и през 1996 (данните за поръчките за 1996 г. не се извеждат, тъй като е зададено условието годината да не е равна на 1996, но се извеждат данните за другите години).

Решение:

```
SELECT ProductID, ProductName
FROM Products
WHERE ProductID NOT IN
    (SELECT ProductID
     FROM [Order Details] OD JOIN Orders O
     ON OD.OrderID = O.OrderID
     WHERE YEAR(OrderDate) = 1996)
```

Друго решение (от резултата на първия SELECT се изважда резултата от втория SELECT (продукти, поръчвани през 1996):

```
SELECT DISTINCT P.ProductID, ProductName
FROM Products P LEFT JOIN [Order Details] OD
ON P.ProductID = OD.ProductID
LEFT JOIN ORDERS O
ON OD.OrderID = O.OrderID
WHERE YEAR(OrderDate) != 1996
      OR YEAR(OrderDate) IS NULL
```

EXCEPT

```
SELECT DISTINCT P.ProductID, ProductName
FROM Products P JOIN [Order Details] OD
ON P.ProductID = OD.ProductID
JOIN ORDERS O
ON OD.OrderID = O.OrderID
WHERE YEAR(OrderDate) = 1996
ORDER BY 1
```

Пример, изискващ използване на подзаявка:

Да се изведат данни за клиентите, които не са
правили поръчки през дадена година (напр.
1996)

Предложение

(от студентите по време на лекцията на 20.04.16)

```
USE Northwind
GO
SELECT CompanyName
FROM Customers
WHERE CustomerID IN
(SELECT CustomerID FROM Orders
WHERE YEAR(OrderDate) != 1996)
ORDER BY 1
```

Не е решение, защото условието във вложената подзаявка само игнорира редовете за поръчките, за които годината е 1996.

В резултата от вложената заявка ще се изведат

- кодовете на клиентите които не са имали поръчки през 1996;
- кодовете на клиентите, които са имали поръчки през различни години, в т.ч. и през 1996.

Данни за клиентите, които са правили поръчки
през дадена година (напр. 1996)

USE Northwind

GO

SELECT CompanyName

FROM Customers

WHERE CustomerID IN

(SELECT CustomerID FROM Orders
WHERE YEAR(OrderDate) = 1996)

Решение

Данни за клиентите, които не са правили поръчки
през дадена година (напр. 1996)

```
USE Northwind
```

```
GO
```

```
SELECT CompanyName
```

```
FROM Customers
```

```
WHERE CustomerID NOT IN
```

```
(SELECT CustomerID FROM Orders  
WHERE YEAR(OrderDate) = 1996)
```