



доц. д-р Цветанка Георгиева-Трифенова

ИЗГЛЕДИ



СЪДЪРЖАНИЕ

- ✗ Създаване, променяне и изтриване на изгледи
- ✗ Променяне на данни посредством изгледи

ИЗГЛЕДИ

- ✗ **Изгледът** (*view*) е таблица, чието съдържание се взема или извлича от други таблици.
 - + не съдържат постоянно данните, съхранявани в базата от данни;
 - + може да се направи обръщение към изглед от SELECT, INSERT, UPDATE или DELETE;
 - + представлява заявка, която се извиква за изпълнение винаги, когато в някоя конструкция се извършва препращане към този изгледа.

СЪЗДАВАНЕ НА ИЗГЛЕД

```
CREATE VIEW view_name [ (column_list) ]  
AS  
    SELECT statement  
[WITH CHECK OPTION]
```

✗ Пример

```
CREATE VIEW All_Employees_And_Customers  
AS  
    SELECT FirstName+' '+LastName AS Name,  
           CompanyName  
    FROM Employees e  
    LEFT JOIN Sales s  
        ON e.EmployeeID = s.EmployeeID  
    FULL JOIN Customers c  
        ON s.CustomerID = c.CustomerID
```

СЪЗДАВАНЕ НА ИЗГЛЕД (2)

```
CREATE VIEW view_name [(column_list)]  
AS  
    SELECT statement  
[WITH CHECK OPTION]
```

× Пример

	Name	CompanyName
1	Стоян Георгиев	Алтметал ЕООД
2	Стоян Георгиев	МИПСТОП 90 ЕООД
3	Георги Хростов	NULL
4	Ваня Хростова	Деница Стар ЕООД
5	Стеля Миланова	Деница Стар ЕООД
6	Атанас Лазаров	NULL
7	Катя Цветанова	NULL
8	NULL	Мики-92 ЕООД

d_Customers

tName AS Name,

ployeeID

stomerID

ИЗПОЛЗВАНЕ НА ИЗГЛЕД

- ✗ Изглед може да се използва в:
 - + заявки за извличане на данни;
 - + съединения с други таблици и/или изгледи;
 - + конструкции за актуализиране на данни;
 - + конструкции за вмъкване на данни;
 - + конструкции за изтриване на данни.

✗ Пример

```
SELECT * FROM All_Employees_And_Customers  
WHERE Name LIKE 'Ст%'  
ORDER BY Name
```

	Name	CompanyName
1	Стела Миланова	Деница Стар ЕООД
2	Стоян Георгиев	Алтметал ЕООД
3	Стоян Георгиев	МИПСТОР 90 ЕООД

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР

```
CREATE VIEW TotalForDay
AS
SELECT
    CONVERT(datetime, CONVERT(char(10), SaleDate, 112))
        AS DateOfSale,
    COUNT(DISTINCT CustomerID) AS CustomersCount,
    COUNT(DISTINCT EmployeeID) AS EmployeesCount,
    COUNT(SaleID) AS SalesCount,
    AVG(TotalForSale) AS AverageTotal,
    SUM(TotalForSale) AS SumTotal
FROM Sales
GROUP BY CAST(CONVERT(char(10), SaleDate, 112) AS
    datetime)
```

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (2)

```
CREATE VIEW TotalForDay
AS
SELECT
    CONVERT(datetime, CONVERT(char(10), SaleDate, 112))
        AS DateOfSale,
    COUNT(DISTINCT CustomerID) AS CustomersCount,
    COUNT(DISTINCT EmployeeID) AS EmployeesCount,
    COUNT(SaleID) AS SalesCount,
    AVG(TotalForSale) AS AverageTotal,
    SUM(TotalForSale) AS SumTotal
```

	DateOfSale	CustomersCount	EmployeesCount	SalesCount	AverageTotal	SumTotal
1	2010-01-30 00:00:00.000	2	1	2	25.375	50.75
2	2010-01-31 00:00:00.000	1	2	2	5.95	11.90

ИЗПОЛЗВАНЕ НА ИЗГЛЕД – ПРИМЕРИ

```
SELECT * FROM TotalForDay
```

```
SELECT * FROM TotalForDay  
WHERE SumTotal=( SELECT MAX(SumTotal)
```

	DateOfSale	CustomersCount	EmployeesCount	SalesCount	AverageTotal	SumTotal
1	2010-01-30 00:00:00.000	2	1	2	25.375	50.75

```
SELECT * FROM TotalForDay  
WHERE SalesCount>=2 AND SumTotal>=1000
```

	DateOfSale	CustomersCount	EmployeesCount	SalesCount	AverageTotal	SumTotal

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (3)

```
CREATE VIEW MaxTotalForCustomer
AS
    SELECT  c.CustomerID, c.CompanyName,
            s.TotalForSale, s.SaleDate
    FROM    Customers c
    INNER JOIN Sales s
        ON  c.CustomerID=s.CustomerID
    WHERE   TotalForSale =
        ( SELECT MAX(TotalForSale)
          FROM Sales s1
          WHERE s.CustomerID = s1.CustomerID )
```

✖ Пример за заявка към изгледа:

```
SELECT * FROM MaxTotalForCustomer
WHERE CompanyName LIKE 'L%'
```

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (3)

```
CREATE VIEW MaxTotalForCustomer  
AS
```

	CustomerID	CompanyName	TotalForSale	SaleDate
1	1	Алтметал ЕООД	8.75	2010-01-30 09:08:00.000
2	3	МИПСТОР 90 ЕООД	42.00	2010-01-30 10:08:00.000
3	2	Деница Стар ЕООД	1.80	2010-01-31 12:08:00.000
4	2	Деница Стар ЕООД	10.10	2010-01-31 13:08:00.000

	CustomerID	CompanyName	TotalForSale	SaleDate
1	3	МИПСТОР 90 ЕООД	42.00	2010-01-30 10:08:00.000
2	2	Деница Стар ЕООД	10.10	2010-01-31 13:08:00.000
3	1	Алтметал ЕООД	8.75	2010-01-30 09:08:00.000

CustomerID)

✗ Пример за заявка към изгледа:

```
SELECT * FROM MaxTotalForCustomer  
WHERE CompanyName LIKE 'A%'
```


СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (4)

```
CREATE VIEW CityPeople
AS
  SELECT CustomerID AS ID, CompanyName, City,
         'клиент' AS Name
  FROM Customers
 UNION
  SELECT SupplierID, CompanyName, City,
         'доставчик'
  FROM Suppliers
```

✗ Пример за заявка към изгледа:

```
SELECT * FROM CityPeople
WHERE City BETWEEN 'А' AND 'Д'
ORDER BY City
```

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (4)

```
CREATE VIEW CityPeople  
AS
```

	ID	CompanyName	City	Name
1	1	Алтметал ЕООД	Велико Търново	клиент
2	2	Деница Стар ЕООД	Велико Търново	клиент
3	3	МИПСТОР 90 ЕООД	Велико Търново	клиент
4	4	Мики-92 ЕООД	Велико Търново	клиент
5	1	БЕТИ - 2002	Велико Търново	доставчик
6	2	АЛЕМИТ ООД	Велико Търново	доставчик
7	3	АППИ ООД	Пясковец	доставчик

CompanyName, City,

City,

✖ Пример за заявка към изгледа:

```
SELECT * FROM CityPeople  
WHERE City BETWEEN 'А' AND 'Д'  
ORDER BY City
```

СЪЗДАВАНЕ НА ИЗГЛЕД – ПРИМЕР (5)

```
CREATE VIEW TopProductPrice
AS
    SELECT TOP 3 WITH TIES
           ProductID, ProductName, Price
    FROM Products
    ORDER BY Price DESC
```

	ProductID	ProductName	Price
1	2	портокали	3.50
2	4	картофи	3.00
3	7	макарони	2.60

✖ Пример за заявка към изгледа:

```
SELECT * FROM TopProductPrice
```

```
SELECT ProductName, SUM(Quantity) AS SumOfQuantity
FROM TopProductPrice t
INNER JOIN SaleDetails sd
    ON t.ProductID=sd.ProductID
GROUP BY ProductName
```

	ProductName	SumOfQuantity
1	картофи	2.000
2	макарони	2.000
3	портокали	5.000

ПРОМЕНЯНЕ И ИЗТРИВАНЕ НА ИЗГЛЕДИ

✗ Променяне на изгледи

```
ALTER VIEW view_name [(column_list)]  
AS  
    SELECT statement  
[WITH CHECK OPTION]
```

✗ Изтриване на изгледи

```
DROP VIEW view_name
```

ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ

- ✗ Модификациите са ограничени до една таблица;
 - + конструкциите INSERT и UPDATE са позволени за изгледи на множество таблици, ако промяната засяга само една таблица;
 - + конструкциите DELETE не са разрешени за изгледи на множество таблици.
- ✗ Някои изгледи са само за четене;
 - + не могат да се добавят (INSERT), изтриват (DELETE) редовете или променят (UPDATE) колони от изглед, включващ обобщаващи функции, UNION, INTERSECT, EXCEPT, ключовите думи GROUP BY или DISTINCT, както и колоните в изгледа, които са изчислими колони или резултат от вградени функции.

ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ (2)

✗ Опцията WITH CHECK OPTION

- + конструкции за добавяне (INSERT) и променяне (UPDATE) на данни чрез изгледи се проверяват дали засегнатите редове отговарят на критериите на изгледа.

✗ Пример

```
CREATE VIEW Customers_VT  
AS  
    SELECT * FROM Customers  
    WHERE City = 'Велико Търново'  
WITH CHECK OPTION
```


ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ (3)

Изпълнението на конструкция от вида:

```
UPDATE Customers_VT  
    SET City = 'София'  
WHERE CustomerID = 2
```

предизвиква извеждане на:

Msg 550, Level 16, State 1, Line 1

The attempted insert or update failed because the target view either specifies WITH CHECK OPTION or spans a view that specifies WITH CHECK OPTION and one or more rows resulting from the operation did not qualify under the CHECK OPTION constraint.

ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ (4)

- ✗ Опцията WITH CHECK OPTION се прилага по отношение на всички изгледи, които съдържат изглед с тази опция.
- ✗ Пример

```
CREATE VIEW ListCustomers_VT  
AS  
    SELECT CustomerID, CompanyName, City  
    FROM Customers_VT
```

ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ (5)

- ✗ Опцията WITH CHECK OPTION се прилага и по отношение на другите изгледи, които изгледът съдържа.
- ✗ Пример

```
CREATE VIEW Product_List
AS
    SELECT ProductID, ProductName, Price,
           Stock, ReorderLevel, Discontinued
    FROM Products
    WHERE Discontinued = 0
```

```
CREATE VIEW ProductsForReorder
AS
    SELECT *
    FROM Product_List
    WHERE Stock <= ReorderLevel
WITH CHECK OPTION
```


ПРОМЕНЯНЕ НА ДАННИ ПОСРЕДСТВОМ ИЗГЛЕДИ (6)

✗ Актуализацията:

```
UPDATE ProductsForReorder  
    SET Discontinued = 1  
WHERE ProductID = 87
```

не нарушава непременно условието в изгледа *ProductsForReorder*, но нарушава условието в изгледа *Product_List*, на който се базира.

✗ Изтриване на ред чрез изглед води до изтриването на ред от съответната таблица:

```
DELETE FROM ProductsForReorder  
WHERE ProductID = 12
```

ЗАДАЧИ

✖ **Задача 1.** Да се създаде изглед, който да показва:

- + **1.1.** имената на всички продукти и броя на техните продажби за текущия ден;
- + **1.2.** данните за продуктите, чиято продажба не е преустановена и наличното количество (`Stock`) е критично (т.е. по-малко или равно на `ReorderLevel`); изгледът да не допуска промяна на данните, нарушаваща условието му;
- + **1.3.** данните за продуктите, чиято доставна цена е по-голяма от доставните цени на всички продукти от дадена категория (`CategoryID = 1`). Да се напише заявка към изгледа, показваща 3^{те} най-ниски цени;
- + **1.4.** датите, на които не е продаван даден продукт (`ProductID = 1`). Да се напише заявка към изгледа, показваща само датите от изминалите 6 месеца, подредени в намаляващ ред.



Цветанка Георгиева-Трифорова, 2017

Някои права запазени.

Презентацията е достъпна под лиценз Creative Commons,

Признание-Некомерсиално-Без производни,

<https://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>