

# **Представления (VIEWS) в SQL**



**Выполнил:  
Иванов НС 8к2492**



# Представления или Views

Представляют собой виртуальные таблицы, однако в отличие от обычных стандартных таблиц в БД представления содержат запросы, которые динамически извлекают используемые данные.


# Преимущества представлений

- Они упрощают комплексные SQL-операции.
- Они защищают данные, так как представления могут дать доступ к части таблицы, а не ко всей таблице.
- Позволяют возвращать отформатированные значения из таблиц в нужной и удобной форме.

# Типы представлений

Кроме основных определяемых пользователем представлений есть также следующие:

- Индексированные представления
- Секционированные представления
- Системные представления




# Модифицируемое представление

В SQL есть такое понятие как модифицируемое представление — при изменении данных в самом представлении, эти данные изменятся и в таблицах, которые эти данные хранят.

# Критерии, опред. является ли представление модифицируемым:

- Оно должно выводиться в одну и только в одну базовую таблицу.
- Оно должно содержать первичный ключ этой таблицы ( это технически не предписывается стандартом ANSI, но было бы неплохо придерживаться этого).
- Оно не должно иметь никаких полей, которые бы являлись агрегатными функциями.
- Оно не должно содержать DISTINCT в своем определении.
- Оно не должно использовать GROUP BY или HAVING в своем определении.
- Оно не должно использовать подзапросы ( это - ANSI\_ограничение которое не предписано для некоторых реализаций )
- Оно может быть использовано в другом представлении, но это представление должно также быть модифицируемыми.
- Оно не должно использовать константы, строки, или выражения значений ( например: `comm * 100` ) среди выбранных полей вывода.
- Для INSERT, оно должно содержать любые пол основной таблицы которые имеют ограничение NOT NULL, если другое ограничение по умолчанию, не определено.

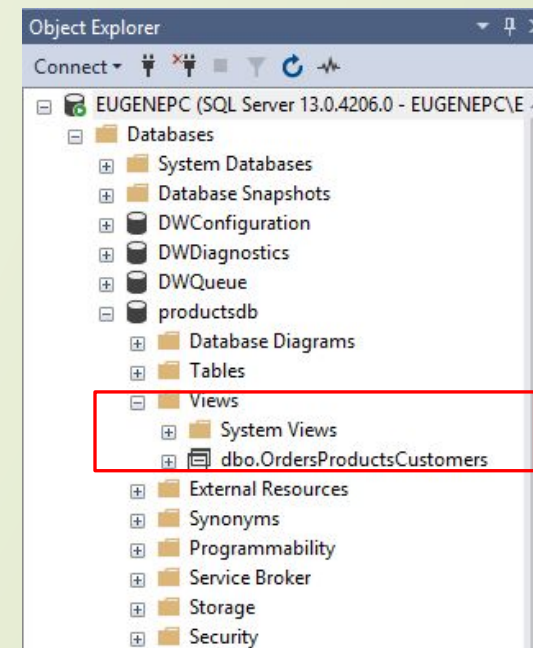


Для **создания представления**  
используется команда **CREATE VIEW**,  
которая имеет следующую форму:

CREATE VIEW название\_представления [(столбец\_1, столбец\_2, ....)]  
AS выражение\_SELECT

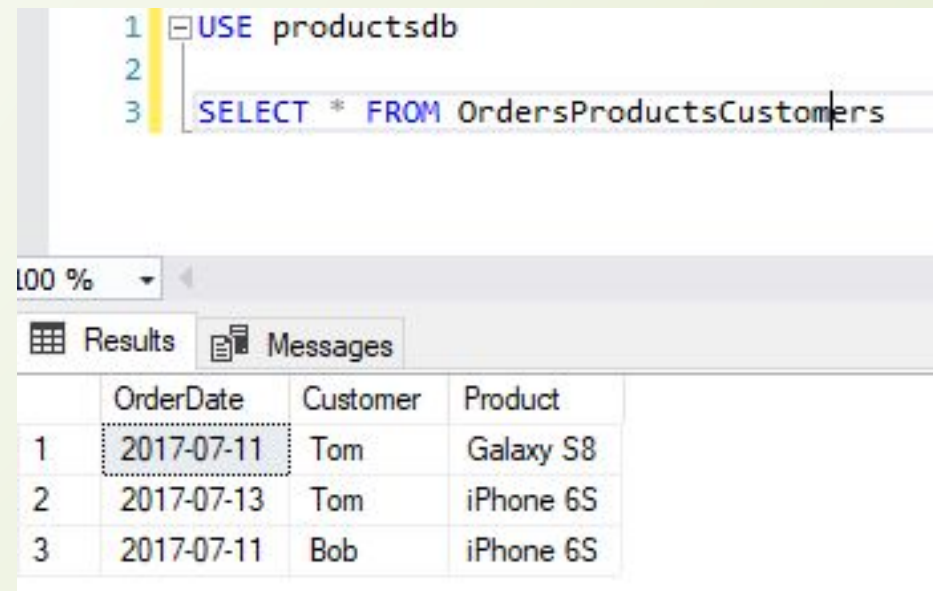
Теперь добавим в базу данных, в которой содержатся данные таблицы, следующее представление:

```
CREATE VIEW OrdersProductsCustomers AS  
SELECT Orders.CreatedAt AS OrderDate,  
       Customers.FirstName AS Customer,  
       Products.ProductName As Product  
FROM Orders INNER JOIN Products ON Orders.ProductId = Products.Id  
INNER JOIN Customers ON Orders.CustomerId = Customers.Id
```





SELECT \* FROM OrdersProductsCustomers



The screenshot shows a SQL Server Enterprise Manager interface. The top pane displays a query with three lines: 1. USE productsdb, 2. (blank), and 3. SELECT \* FROM OrdersProductsCustomers. The bottom pane shows the 'Results' tab with a table containing three rows of data. The first row is selected, and the zoom level is set to 100%.

	OrderDate	Customer	Product
1	2017-07-11	Tom	Galaxy S8
2	2017-07-13	Tom	iPhone 6S
3	2017-07-11	Bob	iPhone 6S

# Изменение представления

Для изменения представления используется команда **ALTER VIEW**.

Эта команда имеет практически тот же самый синтаксис, то и CREATE VIEW:

```
ALTER VIEW название_представления [(столбец_1, столбец_2, ....)]  
AS выражение_SELECT
```



Например, изменим выше созданное представление OrdersProductsCustomers:

```
ALTER VIEW OrdersProductsCustomers
AS SELECT Orders.CreatedAt AS OrderDate,
       Customers.FirstName AS Customer,
       Products.ProductName AS Product,
       Products.Manufacturer AS Manufacturer
FROM Orders INNER JOIN Products ON Orders.ProductId = Products.Id
INNER JOIN Customers ON Orders.CustomerId = Customers.Id
```

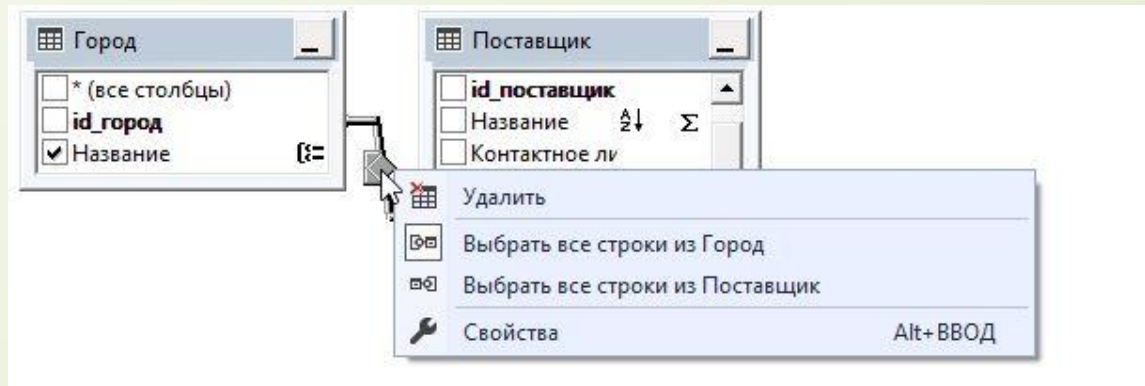
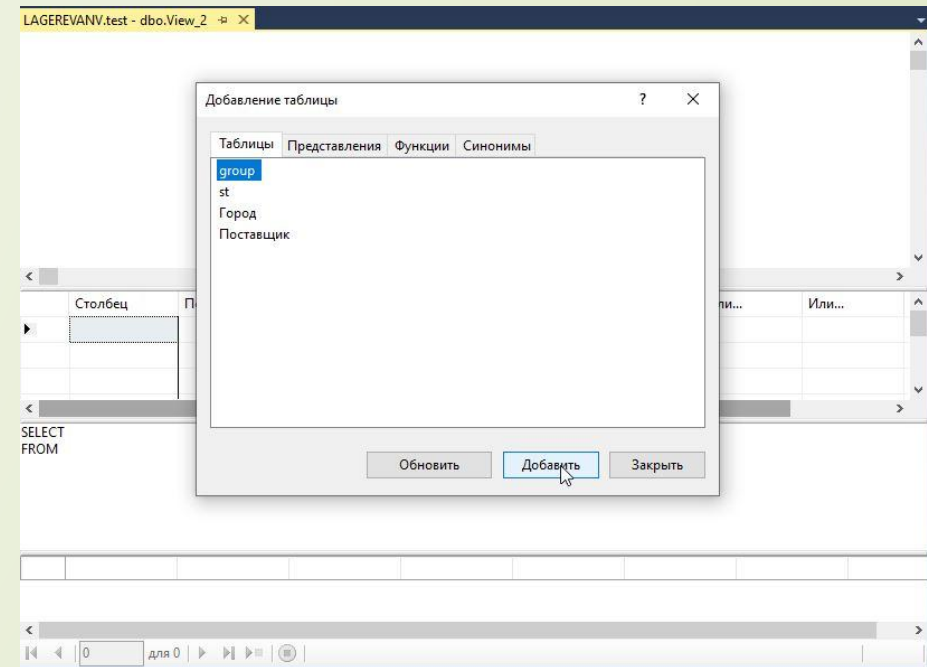
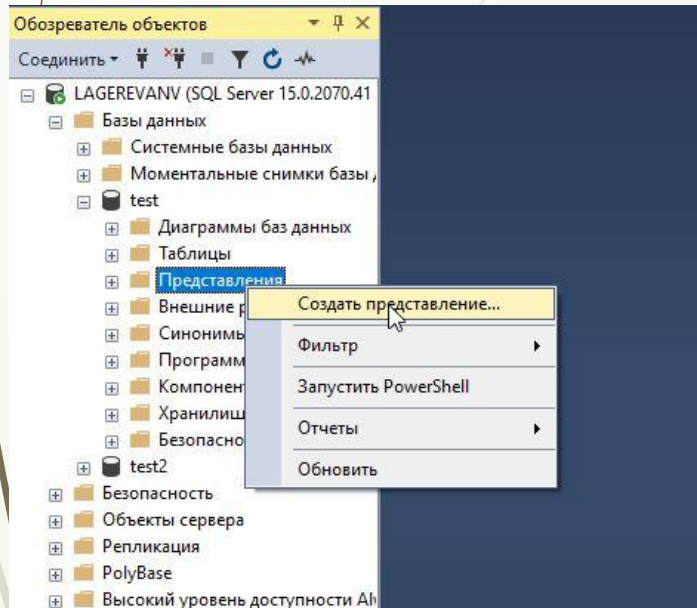
# Удаление представления

Для удаления представления вызывается команда  
**DROP VIEW:**

```
DROP VIEW OrdersProductsCustomers
```

Также стоит отметить, что при удалении таблиц также следует удалить и представления, которые используют эти таблицы.

# Представления в Management Studio



LAGEREVANV.test - dbo.View\_1 SQLQuery1.sql - LA...lager\_pnacc77 (51))

Город

- ☐ \* (все столбцы)
- ☐ id\_город
- ☒ Название

Поставщик

- ☐ id\_поставщик
- ☐ Название
- ☐ Контактное ли
- ☐ Телефон
- ☐ id\_город

Столбец	Псевдо...	Таблица	Выход	Тип сортиро...	Порядок сор...	Group By	Фильтр	Или...
Название	Кол	Поставщик	<input checked="" type="checkbox"/>	По возраста...	1	Count		
Название	Город	Город	<input checked="" type="checkbox"/>			Group By		

```
SELECT TOP (100) PERCENT COUNT(dbo.Поставщик.Название) AS Кол, dbo.Город.Название AS Город
FROM      dbo.Город LEFT OUTER JOIN
           dbo.Поставщик ON dbo.Город.id_город = dbo.Поставщик.id_город
GROUP BY  dbo.Город.Название
ORDER BY  Кол
```

Кол	Город
1	Москва
1	Смоленск
2	Брянск
3	Уфа

1 для 4 Ячейка доступна только для чтения.

- Выполнить код SQL Ctrl+R
- Проверить синтаксис SQL
- Вырезать Ctrl+X
- Копировать Ctrl+C
- Вставить Ctrl+V
- Удалить Del
- Выделить все Ctrl+A
- Панель
- Очистить результаты
- Свойства Alt+ВВОД



**СПАСИБО ЗА ВНИМАНИЕ!**  
**БЛАГО ОН КАК НАЙКИ – I JUST DO IT**