

ПРЕДСТАВЛЕНИЯ VIEW

**Представления или
просмотры (view)
представляют собой
временные (виртуальные)
таблицы и являются объектами
БД, информация в которых не
хранится постоянно как в
базовых таблицах, а
формируется динамически при
обращении к ним.**

В каких случаях применяют представления

1. Повторное использование инструкций SQL.
2. Упрощение сложных запросов.
3. Вывод фрагментов таблицы вместо всей таблицы.
4. Защита данных. Пользователям можно предоставить доступ к определенному подмножеству таблиц, а не ко всем таблицам.
5. Изменение форматирования и способа отображения данных.

Представление – это фактически тот же запрос, который выполняется каждый раз при участии в какой-либо команде.

Результат выполнения этого запроса в каждый момент времени становится содержанием представления.

У пользователя создается впечатление, что он работает с настоящей, реально существующей таблицей.

В отличие от других объектов базы данных представление не занимает дисковой памяти за исключением памяти, необходимой для хранения самого представления.

Синтаксис

CREATE | ALTER VIEW <ИМЯ

просмотра>

[(имена столбцов)]

AS

SELECT [запрос]

[WITH CHECK OPTION]

***Имена столбцов* в представлении соответствуют именам столбцов в исходных таблицах.**

Явное указание имени столбца требуется для вычисляемых столбцов или при объединении нескольких таблиц имеющих столбцы с одинаковыми именами.

Имена столбцов перечисляются через запятую, в соответствии с порядком их следования в представлении.

Параметр **WITH CHECK OPTION** предписывает серверу выполнять проверку изменений производимых через представления на соответствие критериям, определенным в операторе **SELECT**.

Оно гарантирует, что сделанные изменения будут представлены в таблице.

**Если пользователь пытается
выполнить изменения,
приводящие к исключению
строки из представления, при
заданном аргументе WITH
CHECK OPTION сервер выдаст
сообщение об ошибке и
изменения будут отклонены.**

Удаление представления

**DROP VIEW <ИМЯ
просмотра>;**

Пример 1.

Представление клиентов из Москвы.

CREATE VIEW VV1

(Фамилия, Телефон)

AS

SELECT surname, phone

FROM client

WHERE address **LIKE** '%Москва%';

В **примере**
представление **просто**
ограничивает **доступ**
пользователя **к** **данным**
таблицы Client, **позволяя**
видеть **только** **часть**
значений.

Представление можно использовать в команде так же, как и любую другую таблицу. К представлению можно строить запрос, модифицировать его(если оно отвечает определенным требованиям), соединять с другими таблицами.

Содержание представления не фиксировано и обновляется каждый раз, когда на него ссылаются в команде.

Обычно в представлениях используют имена, полученные непосредственно из имен полей основной таблицы.

В случае использования итоговых функций или вычисляемых столбцов, необходимо столбцам дать новые имена.

Пример 2.

Вывести информацию о продуктах с ценой, уменьшенной на 5 процентов для тех продуктов, которых на складе больше 100.

```
CREATE VIEW VV2
```

```
(name, kol_vo, sale_price)
```

```
AS
```

```
SELECT name_prod, qty, price -  
price*0.05
```

```
FROM product
```

```
WHERE qty > 100;
```

Пример 3.

Вывести сумму заказов отдельно для каждого клиента

```
CREATE VIEW VV3
```

```
(id_client, summa)
```

```
AS
```

```
SELECT id_client,
```

```
sum(price*product.qty)
```

```
FROM order
```

```
GROUP BY id_client;
```

Обновление данных в представлениях

**Представления
могут быть
модифицируемыми и
немодифицируемыми
(в режиме «только
для чтения»)**

Обновление данных в представлениях

Представления в режиме «только для чтения» создают сложные запросы, которые можно выполнить и повторить снова, сохраняя полученную информацию.

Немодифицируемое представление с данными из разных таблиц

Пример 4.

Представить клиентов с указанием номера телефона и количества заказанного товара.

```
CREATE VIEW VV4
```

```
AS
```

```
SELECT Client.surname,
```

```
Client.phone, Order.qty
```

```
FROM Client INNER JOIN Order
```

```
ON Client.id=Order.id;
```

Немодифицируемое представление с данными из разных таблиц

Пример 4.

Выбрать клиентов с количеством товара в заказе, превышающем 100.

```
SELECT Client.surname, Order.qty  
FROM VV4  
WHERE Order.qty>100;
```

Немодифицируемое представление для форматирования данных

Пример 5.

Представить данные о клиентах в следующем виде: Иванов Олег, тел. 89062356521.

```
CREATE VIEW VV5
```

```
AS
```

```
SELECT CONCAT(surname, name,'  
тел.',phone) AS Dannyae
```

```
FROM Client;
```

Немодифицируемое представление для форматирования данных

Пример 6.

Представить данные о клиентах в следующем виде: Иванов Олег, тел. 89062356521, используя готовое представление.

```
SELECT *  
FROM VV5;
```

Обновление данных в представлениях

**Модифицируемые
используются для защиты
конфиденциальных или
не относящихся к
потребностям данного
пользователя частей
таблицы.**

Критерии модифицируемого представления

1. Основывается только на одной базовой таблице
2. Содержит первичный ключ этой таблицы
3. Не содержит **DISTINCT** в своем определении
4. Не применяет в своем определении подзапросы
5. НЕ использует **GROUP BY** или **HAVING**

Критерии модифицируемого представления

4. В просмотр должен быть включен каждый столбец таблицы, имеющий атрибут **NOT NULL**
5. Оператор **SELECT** просмотра не использует агрегирующие(итоговые) функции, соединения таблиц, хранимые процедуры и функции, определенные пользователем
6. Основывается на одиночном запросе, поэтому объединение **UNION** не разрешено.

Обновление данных в представлениях

Если просмотр удовлетворяет этим условиям, к нему могут применяться операторы **INSERT, UPDATE, DELETE.**