

ЛАБОРАТОРНАЯ РАБОТА
ТЕМА: СОЗДАНИЕ ПРЕДСТАВЛЕНИЙ
ЦЕЛЬ РАБОТЫ: ИЗУЧИТЬ СПОСОБЫ СОЗДАНИЯ И
ИСПОЛЬЗОВАНИЯ ПРЕДСТАВЛЕНИЙ, ТРИГГЕРОВ,
ХРАНИМЫХ ПРОЦЕДУР.

ОСНОВНЫЕ СВЕДЕНИЯ

Представление

Механизм представлений является мощным средством СУБД, позволяющим скрыть реальную структуру БД от некоторых пользователей. Реально представление является хранимым в БД запросом, отличаясь от запроса лишь тем, что при изменении данных в таблице они автоматически изменяются и в представлении, что обеспечивает актуальное состояние данных. Представление дает возможность пользователю работать только с теми данными, которые ему нужны, кроме того, механизм представлений позволяет скрыть служебные, конфиденциальные данные.

Создание представления базы данных в системе SQL-сервер может осуществляться следующими способами:

Первый способ: С помощью SQL запроса.

Представление создается командой CREATE VIEW, после которой указывается его имя, а далее следует запрос, формирующий тело представления:

Синтаксис:

```
CREATE VIEW имя_представления  
AS SELECT ...
```

Горизонтальные представления

Горизонтальное представление позволяет ограничить доступ пользователей определенными строками из одной или нескольких таблиц. Например, создать представление, позволяющее руководителю отделения компании под номером 3 иметь доступ только к данным сотрудников своего отделения:

```
CREATE VIEW STAFF3  
AS SELECT *  
FROM STAFF  
WHERE STAFF.Branch_no= 3;
```

Преимущество представления по сравнению с запросами к БД заключается в том, что оно будет модифицировано автоматически всякий раз, когда таблица, лежащая в его основе, изменяется. Например, если в отделение номер 3 будет принят новый сотрудник, то он автоматически отобразится в представлении.

Вертикальные представления

Вертикальные представления позволяют дать доступ к информации в таблице, исключив некоторые поля. Например, для того, чтобы скрыть

данные о зарплате сотрудников надо отобразить в таблицу все поля, исключая поле Salary.

```
CREATE VIEW SALARY_OFF  
AS SELECT Staff_no, FName, LName, DOB, City, Street, House, Flat, Stel_no,  
Position, Branch_no  
FROM STAFF;
```

В рассмотренном примере поля представлений имеют имена, полученные непосредственно из имен полей основной таблицы. Однако иногда возникает необходимость назвать столбцы новыми именами. Это, например, может потребоваться в случае, если столбцы являются вычисляемыми и поэтому не имеющими имен.

Имена, которые необходимо присвоить полям, записываются в круглых скобках после имени таблиц. Они могут не указываться, если совпадают с именами полей запрашиваемой таблицы.

В SQL существует понятие групповых представлений, то есть таких, которые содержат предложение GROUP BY. Представления могут быть основаны сразу на нескольких базовых таблицах.

Упражнение: С помощью SQL запроса создать представление, содержащее данные об агентах, отвечающих за продажу объектов. Представление должно включать номер отделения (Branch_no), номер работника (Staff_no) и сведения о количестве объектов, за которые он отвечает:

```
CREATE VIEW STAFF_PROP (Branch_no, Staff_no, Properties)  
AS SELECT STAFF.Branch_no, STAFF.Staff_no, COUNT(*)  
FROM STAFF INNER JOIN PROPERTY ON STAFF.Staff_no=  
PROPERTY.Staff_no  
GROUP BY STAFF.Branch_no, STAFF.Staff_no;
```

Одной из причин использования представлений является стремление к упрощению многотабличных запросов. После определения представления с соединением нескольких таблиц можно использовать простейшие однотабличные запросы к этому представлению. Однако при создании запросов к представлениям, созданным на основе нескольких таблиц, следует учитывать следующие ограничения:

- если столбец в представлении создается с использованием обобщающей функции, то этот столбец может указываться только в предложениях SELECT и ORDER BY тех запросов, которые обеспечивают доступ к данному представлению. Этот столбец не может использоваться в предложении WHERE, а также не может быть аргументом в обобщающей функции;
- сгруппированное представление не должно соединяться с таблицами базы данных или другими представлениями.

Представление может быть обновляемым только в следующих случаях:

- в нем не используется ключевое слово DISTINCT;







- каждый элемент в списке предложения SELECT представляет собой имя столбца, а не выражение или обобщающую функцию;
- представление должно быть построено на базе одной таблицы;
- запрос, определяющий представление не должен содержать предложений GROUP BY и HAVING.

Второй способ: С помощью конструктора.

Для создания представления надо:

1. Выбрать группу *Представления* в списке объектов базы данных, после чего воспользоваться командой *Создать представление*.

После выполнения этих действий загрузится конструктор представлений. Диалоговое окно дизайнера представлений состоит из следующих частей:

- Панель диаграмм – используется для добавления новых таблиц в представление, описание связей между ними, определения полей, которые будут участвовать в представлении. Для открытия/закрытия данной панели используется кнопка "Показать область схемы" ;
 - Панель-список – на этой панели отображается перечень полей, выбранных в Панели диаграмм. Можно так же добавить новые поля, определить наличие различных критериев и т.д. Для открытия/закрытия данной панели используется кнопка "Показать область условий" ;
 - SQL-панель – данная панель используется для ввода SQL-команд, с помощью которой создается представление. Для открытия/закрытия данной панели используется кнопка "Показать область SQL кода" ;
 - Панель результатов – работу произведенных настроек удобно проверить, используя данную панель, по нажатию кнопки Run , отображаются результаты настроенного представления. Для открытия/закрытия данной панели используется кнопка "Показать область результатов" ;
2. Для создания нового представления надо добавить в него необходимые таблицы. Для этого используется кнопка "Добавление таблицы" . При выполнении этого действия на экран будет выведено диалоговое окно с перечнем имеющихся в базе данных таблиц. Используя кнопку "Добавление таблицы" можно добавить выбранные таблицы в представление.

Кроме того представления могут строиться не только на основании таблиц, но и с использованием других представлений. Для этого в диалоговом окне существует закладка *Представления*, которая позволяет добавлять существующие представления базы данных в создаваемое представление.

3. После добавления таблиц, перечень их полей будет отображен в диаграмме представления. Если были ранее установлены связи между полями данной таблицы с использованием первичных и внешних

ключей, то будет добавлено соответствующее графическое отображение.

4. На панели диаграмм данного диалогового окна, слева от имени поля таблиц, имеется флажок, при установке которого данное поле будет выведено на экран в результате выполнения представления. При выборе имени этого поля, оно автоматически появляется в списке «Панель-список», и в области оператора SELECT на панели *SQL кода*. Проверить правильность создания представления можно, используя кнопку «Выполнить код SQL», в результате чего должны отобразиться данные из созданного представления на панели *результатов*.
5. После сохранения созданного представления его имя появиться в списке объектов *Представления* базы данных. Для просмотра информации из этого представления надо выполнить команду *Выбрать первые 1000 строк*, предварительно выбрав предварительно его в списке объектов *Представления*.

Удаление представления из базы данных осуществляется командой *DROP VIEW имя_представления*. При удалении представления пользователь должен являться его владельцем.

Упражнение: Создайте представление STAFF_PROP, содержащее данные о работниках, отвечающих за продажу объектов.

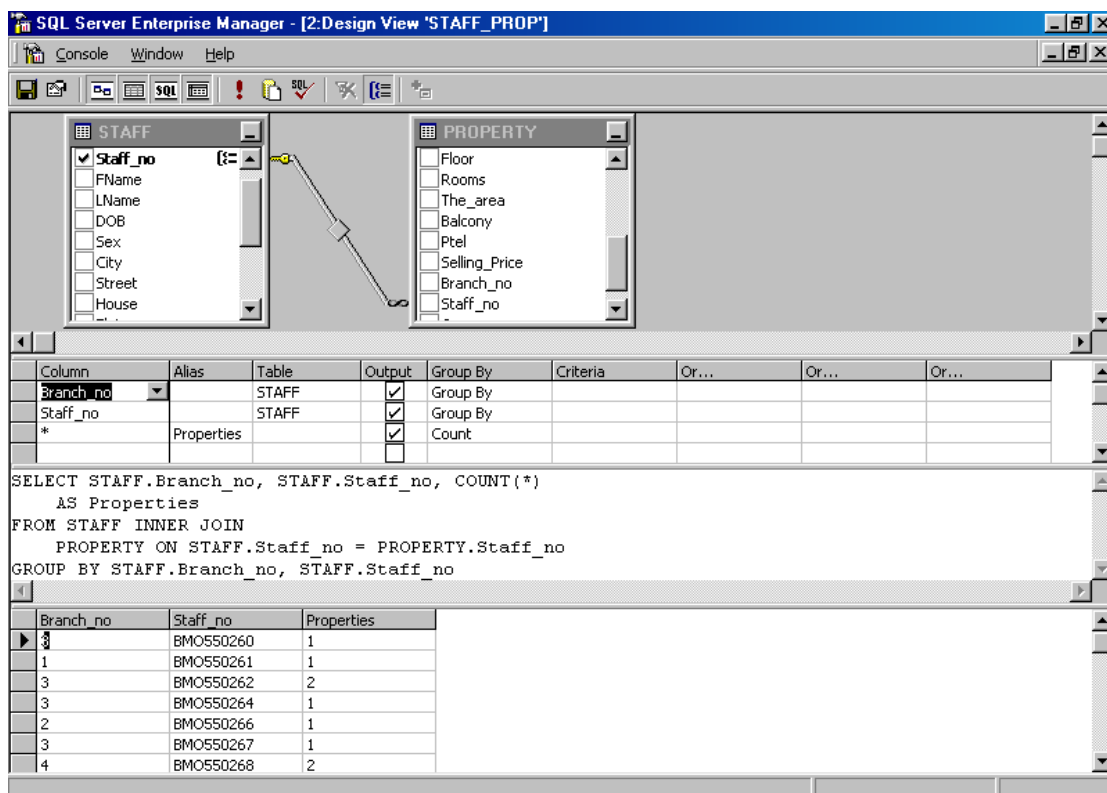


Рисунок 1