**Лабораторная работа № 5**

**Создание представлений**

**Задание**

1.В соответствие с техническим заданием создать необходимые представления.

2.Предусмотреть для каждой группы пользователей не менее 5 представлений. Некоторые представления могут содержать запросы из предыдущей лабораторной работы.

3.Каждое представление проанализировать, и создать в таблицах недостающие индексы.

4.Подготовить и сдать преподавателю отчёт по лабораторной работе в печатном виде.

5.Отчёт должен содержать:

* титульный лист;
* описание разработанных представлений;
* для каждого представления привести:
  + назначение представления,
  + группы пользователей, которые будут его выполнять в процессе работы,
  + инструкцию создания представления на языке SQL,
  + скриншот результата выполнения представления,
  + список полей, для которых определены индексы.

6.Продемонстрировать преподавателю базу данных с работающими представления. При необходимости введите в базу дополнительное количество тестовых данных, так чтобы результаты выполнения представлений были содержательными (непустыми).

**Учебный пример разработки базы данных**

**Создание представлений**

Представление(view) это виртуальная таблица, содержимое которой (столбцы и строки) определяется запросом. Фактически это таблица-шаблон, данные в ней непосредственно не хранятся, а вынимаются из основных таблиц.

Представление можно использовать в следующих целях:

-для представления информации из базы данных в удобном для пользователя виде,

-в качестве механизма безопасности, позволяющего пользователям обращаться к данным через представления, при этом можно не предоставлять им разрешений на непосредственный доступ к базовым таблицам.

Рассмотрим пример.

В нашей учебной базе данных «Кафе» мы создавали запрос «Меню нашего кафе», который выдавал список предлагаемых блюд.

На языке SQL запрос выглядел так:

-- Меню нашего кафе

SELECT Вид, Блюдо AS [Название блюда], Цена

FROM Блюда

ORDER BY Вид, Цена DESC;

Создадим на основе этого запроса представление с именем «Меню». Уберем только сортировку данных, так как для меню может потребоваться разная сортировка.

CREATE VIEW Меню

AS

SELECT Вид, Блюдо AS [Название блюда], Цена

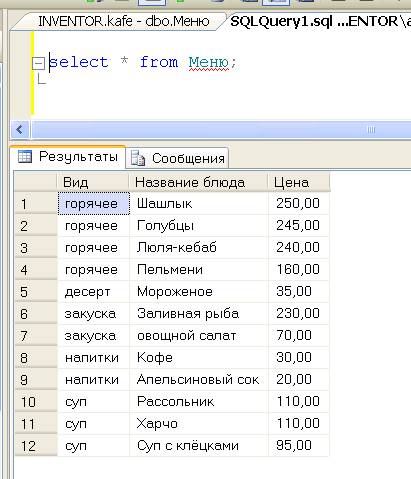
FROM Блюда;

Теперь представление «Меню» создано и храниться в нашей базе данных. Мы можем использовать его в любых запросах как обычную таблицу. Например,

SELECT \* FROM Меню

ORDER BY Вид, Цена DESC;

Получим:



В среде MS SQL Server Management Studio мы можем создавать представления не только с помощью инструкции CREATE VIEW , но также в визуальном режиме конструирования. Для этого в Обозревателе объектов находим нашу базу данных и в узле «Представления» выполняем из контекстного меню команду «Создать представление…».

Попробуйте самостоятельно разработать наиболее удобный для работы каждой группы пользователей набор представлений. Используйте как визуальный режим создания, так и инструкции CREATE VIEW .

Представление 1.

CREATE VIEW [Счет со скидкой]

AS

SELECT ID\_обращения AS Обращение,K.Фамилия, K.Имя, K.Отчество,O.[Заявленный дефект],

O.Гарантия,O.Стоимость,O.Скидка, Итоговая = Стоимость\*(100-Скидка)\*0.01

FROM Обращения as O

LEFT JOIN Клиенты as K ON K.Клиент\_ID=O.ID\_клиента

Представление 2.

**Назначение:** Итоговая стоимость ремонта с учетом скидки.

**Группы пользователей:** Администраторы.

CREATE VIEW [Итоговая стоимость] AS

SELECT ID\_клиента AS Клиент ,ID\_обращения AS Обращение,Стоимость = Стоимость\*(100-Скидка)\*0.01

FROM Обращения

Представление 3.

CREATE VIEW [Завершение ремонта] AS

SELECT ID\_обращения as Обращение,ID\_клиента AS Клиент, Номер\_техники as Техника,

[Заявленный дефект] as Дефект,DATEADD(day,[Время ремонта],[Дата принятия]) as Конец\_ремонта

FROM Обращения

Представление 4.

CREATE VIEW [Гарантийные ремонты] AS

SELECT ID\_клиента,Номер\_техники,Стоимость,[Дата принятия],[Время ремонта]

FROM Обращения

WHERE (Гарантия ='Есть' OR Гарантия ='Улучшенная')

Представление 5.

CREATE VIEW [Накладная поставок] AS

SELECT Поставки.[Номер поставки],Поставки.[Стоимость поставки],

Поставки.[Ед.товара],Поставщики.[Откуда поставляет] , Сотрудники.Фамилия

FROM Поставки, Поставщики, Сотрудники

WHERE Поставки.ID\_поставщика = Поставщики.ID\_поставщика AND

Сотрудники.ID\_сотрудника=Поставки.ID\_сотрудника

Представление 6.

CREATE VIEW [Расположение деталей] AS

SELECT Z.[Название запчасти],Z.Стоимость,M.номер\_стелажа,M.номер\_полки

FROM Запчасти as Z

JOIN [Места хранения] as M ON M.id\_места=Z.id\_места

Представление 7.

CREATE VIEW [Поставки запчастей] AS

SELECT Z.[Название запчасти],Z.Стоимость,P2.Название as Поставщик,P2.[Откуда поставляет] as Доставлена

FROM Запчасти as Z

JOIN Поставки as P1 ON P1.[Номер поставки]=Z.[Номер поставки]

JOIN Поставщики as P2 ON P1.ID\_поставщика=P2.ID\_поставщика

Представление 8.

CREATE VIEW [Дефекты техник] AS

SELECT T1.[Серийный номер],T1.Модель,P1.[Заявленный дефект],P2.[Название запчасти],P1.Стоимость,P1.Гарантия

FROM Используются as E

JOIN Обращения as P1 ON P1.ID\_обращения=E.id\_обращения

JOIN Запчасти as P2 ON P2.ID\_запчасти=E.id\_детали

JOIN Техника as T1 ON T1.[Серийный номер]=P1.Номер\_техники

Представление 9.

CREATE VIEW [Средняя стоимость ремонта телефона] AS

SELECT Техника.Компания, SUM(Обращения.Стоимость\*(100-Обращения.Скидка)/100)/COUNT(\*) as [Средняя стоимость ремонта]

FROM Обращения

JOIN Техника ON Техника.[Серийный номер]=Обращения.Номер\_техники

GROUP BY Техника.Компания

HAVING SUM(Обращения.Стоимость\*(100-Обращения.Скидка)/100)>1000

Представление 10.

CREATE VIEW [Общая сумма поставок] AS

SELECT Поставщики.Название, SUM(Поставки.[Стоимость поставки]) as [Общая стоимость поставки]

FROM Поставки

JOIN Поставщики ON Поставщики.ID\_Поставщика=Поставки.ID\_поставщика

GROUP BY Поставщики.Название

HAVING SUM(Поставки.[Стоимость поставки])>1000

Представление 11.

CREATE VIEW [Отремонтированная техника] AS

SELECT K.Фамилия,K.Имя,K.Отчество,K.Телефон,K.email,Техника.Модель

FROM Техника

JOIN Клиенты as K ON K.Клиент\_ID=Техника.ID\_клиент

WHERE Техника.Состояние='готов'

Представление 12.

CREATE VIEW [Техника без гарантии] AS

SELECT ID\_клиента,Номер\_техники,Стоимость,[Дата принятия],[Время ремонта]

FROM Обращения

WHERE (Гарантия ='Отсутствует' )

Представление 13.

CREATE VIEW [Поставка запчастей] AS

SELECT Z.[Название запчасти],Z.Стоимость,P2.Название as Поставщик,P2.[Откуда поставляет] as Доставлена

FROM Запчасти as Z

JOIN Поставки as P1 ON P1.[Номер поставки]=Z.[Номер поставки]

JOIN Поставщики as P2 ON P1.ID\_поставщика=P2.ID\_поставщика

where P2.[Откуда поставляет]=(SELECT [Откуда поставляет]

FROM Поставщики

WHERE ID\_поставщика=2)

Представление 14.

Представление 15.