

УВАЖАЕМЫЕ СТУДЕНТЫ! Изучите повторите теоретический материал по созданию и наполнению таблиц информацией в среде MS SQL SERVER, выполните задание лабораторной работы, при выполнении готовьте отчет к лабораторной работе – сохраняйте скриншоты пошагового выполнения работы.

Работа рассчитана на 4 часа.

Результаты работы – отчет, предоставить преподавателю на e-mail: xvsviv@rambler.ru **в трехдневный срок с момента получения задания.**

При возникновении вопросов по приведенному материалу обращаться по следующим номерам телефонов: 072-138-93-11.

ВНИМАНИЕ!!! При отправке работы, не забывайте указывать ФИО студента, наименование дисциплины, дата проведения занятия (по расписанию).

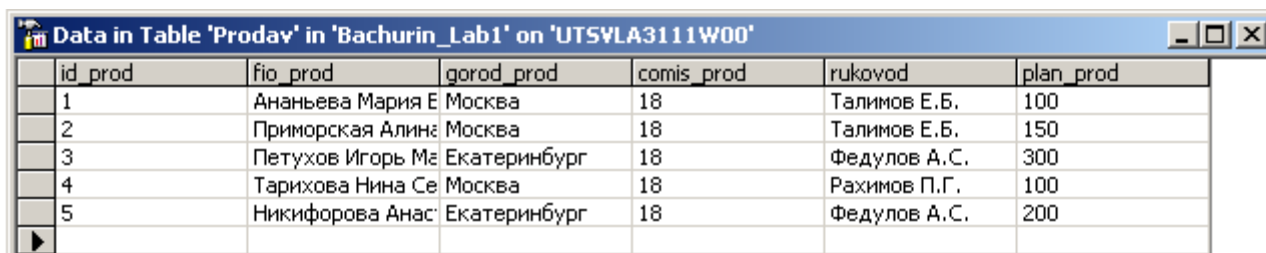
Лабораторная работа на тему: «Создание запросов с использованием представлений.»

Цель работы: практическое освоение методов создания и использования в MS SQL SERVER представлений, хранимых процедур и триггеров.

Ход работы

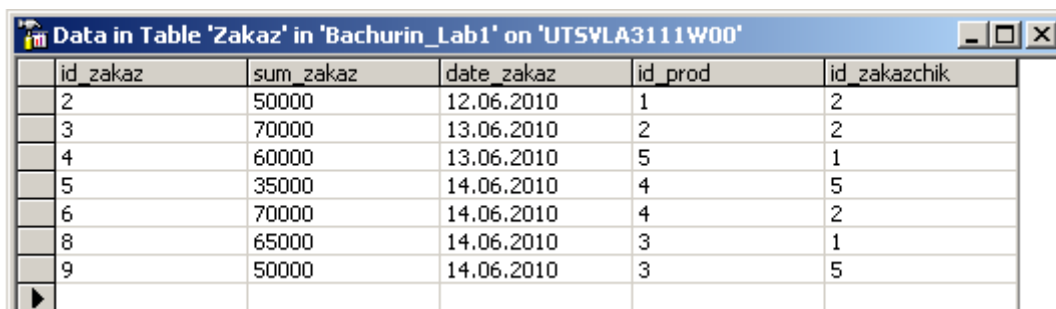
Создайте базу данных Bachurin_Lab1. Ниже приведены заполненные таблицы базы данных.

1. Таблица «Продавцы»:



id_prod	fio_prod	gorod_prod	comis_prod	rukovod	plan_prod
1	Ананьева Мария Е	Москва	18	Талимов Е.Б.	100
2	Приморская Алина	Москва	18	Талимов Е.Б.	150
3	Петухов Игорь Ма	Екатеринбург	18	Федулов А.С.	300
4	Тарихова Нина Се	Москва	18	Рахимов П.Г.	100
5	Никифорова Анас	Екатеринбург	18	Федулов А.С.	200

2. Таблица «Заказы»:



id_zakaz	sum_zakaz	date_zakaz	id_prod	id_zakazchik
2	50000	12.06.2010	1	2
3	70000	13.06.2010	2	2
4	60000	13.06.2010	5	1
5	35000	14.06.2010	4	5
6	70000	14.06.2010	4	2
8	65000	14.06.2010	3	1
9	50000	14.06.2010	3	5

3. Таблица «Заказчики»:

id_zakazchik	fio_zakazchik	gorod_zakazchik	rating_zakazchik	credit
1	Петров Олег Дми	Магадан	83	150000
2	Антипов Валерий	Екатеринбург	45	75000
3	Берегов Михаил Е	Москва	96	500000
4	Чернов Антон Пет	Тула	62	100000
5	Голубев Иосиф Д	Тбилиси	78	120000
6	Горский Борис Ни	Москва	85	90000
7	Романов Глеб Ива	Екатеринбург	91	270000

4. Таблица «Товар»:

id_tovar	naim_tovar	tsena	kolvo
1	Монитор Samsung	4500	20
2	Монитор Samsung	5300	15
3	Монитор Samsung	6420	20
4	Клавиатура Geniu	200	30
5	Монитор LG 19'	4550	22
6	Монитор LG 21'	6950	10
7	Клавиатура LG Ст	190	28
8	Принтер лазерный	4000	8
9	Принтер лазерный	3850	11

С помощью SQL Server Query Analyzer выполнить следующие задания:

1. Создайте представление, которое бы показывало всех заказчиков имеющих рейтинг выше 80.

```
create view hiratings
as select fio_zakazchik, rating_zakazchik
from zakazchik
where rating_zakazchik>80
```

2. Создайте представление, которое бы показывало количество продавцов в каждом городе.

```
create view prod_count
as select gorod_prod as 'Город', count(fio_prod) as 'Количество'
from prodav
group by gorod_prod
```

3. Создайте представление, которое бы показывало каждого продавца с многочисленными заказчиками.

```
create view prod_for_many_zak
as
select a.fio_prod as 'Продавец', count(c.fio_zakazchik) as 'Количество заказчиков'
from prodav a, zakaz b, zakazchik c
where a.id_prod=b.id_prod and b.id_zakazchik=c.id_zakazchik
group by a.fio_prod
having count(c.fio_zakazchik)>1
```

4. Создайте представление, которое показывало фамилии продавцов, не выполняющих план продаж.

```
create view prod_plan
as select a.fio_prod as 'Продавец'
   from prodav a, zakaz b, zakazchik c
   where a.id_prod=b.id_prod and b.id_zakazchik=c.id_zakazchik
   group by a.fio_prod
   having count(c.fio_zakazchik)<sum(a.plan_prod)
```

5. Создайте представление, которое бы показывало всех заказчиков из указанного города с рейтингом больше 100.

```
create view zakazchiki
as select fio_zakazchik
   from zakazchik
   where rating_zakazchik>80 and gorod_zakazchik='Москва'
```

6. Создайте представление, которое бы показывало фамилии продавцов и их общую сумму комиссионных.

```
create view prodavcy
as select a.fio_prod, sum(a.comis_prod)*count(c.fio_zakazchik) as 'Сумма
комиссионных'
   from prodav a, zakaz b, zakazchik c
   where a.id_prod=b.id_prod and b.id_zakazchik=c.id_zakazchik
   group by a.fio_prod
```

7. Создайте представление, которое бы показывало сумму заказов по каждому городу заказчика.

```
create view gorod_zakazchika
as select c.fio_zakazchik, sum(b.sum_zakaz)*count(a.fio_prod) as 'Сумма
заказов'
   from prodav a, zakaz b, zakazchik c
   where a.id_prod=b.id_prod and b.id_zakazchik=c.id_zakazchik
   group by c.fio_zakazchik
```

8. Создайте представление таблицы Продавцов с именем Commissions (Комиссионные). Это представление должно включать только поля Код_продавца, ФИО_продавца и Комиссионные_продавца. С помощью этого представления, можно будет вводить или изменять комиссионные, но только для значений между 0,1 и 0,2.

```
create view table_prodav
as select id_prod, fio_prod, comis_prod
   from prodav
   where comis_prod between 0.1 and 0.2
```

9. Напишите хранимую процедуру, которая увеличивает комиссионные у тех продавцов, сумма продаж которых превышает среднюю сумму продаж всей компании.

```
create procedure comis_plus
@value integer
as
update prodav
set comis_prod=comis_prod+@value
where avg(zakaz.sum_zakaz)<(select sum(a.sum_zakaz)
                             from zakaz a, prodav b
                             where a.id_prod=b.id_prod)
```

Запуск процедуры:

```
execute comis_plus "10"
```

10. Напишите хранимую процедуру с параметрами, показывающую фамилии и сумму продаж продавцов, работающих под руководством указанного начальника.

```
create procedure boss
@boss_name varchar(30)
as
select a.fio_prod, sum(b.sum_zakaz)
from prodav a, zakaz b
where a.id_prod=b.id_prod and a.rukovod=@boss_name
group by a.fio_prod
```

Запуск процедуры:

```
execute boss "Абрамович"
```

11. Напишите хранимую процедуру, которая при выполнении заказа изменяла количество товара в таблице «Товар».

```
create procedure tovar_kolvo
@kolvo integer
as
update tovar
set kolvo=kolvo-@kolvo
```

Запуск процедуры:

```
execute tovar_kolvo "15"
```

12. Напишите триггер, который при добавлении или обновлении таблицы «Заказчики» проверял условие: значение рейтинга должно быть положительным и не превосходить 500.

```
create trigger checking
on zakazchik
for insert, update
as
if rating_zakazchik between 0 and 500
```

```
insert into zakazchik
values (inserted.id_zakazchik, inserted.fio_zakazchik,
        inserted.gorod_zakazchik, inserted.rating_zakazchik, inserted.credit)
else print 'Рейтинг должен быть в пределах от 0 до 500'
```

13. Напишите триггер, который при добавлении или обновлении заказа в кредит сравнивал бы итоговую сумму заказа с максимально допустимым кредитом для данного заказчика.

```
create trigger credit
on zakazchik
for insert, update
as
if credit < (select sum_zakaz
             from zakaz a, zakazchik b
             where a.id_zakazchik=b.id_zakazchik)
print 'Сумма заказа больше максимально допустимой для данного заказчика'
```

14. Написать триггер, который запрещает удалять запись из таблицы «Заказы», если с ней связаны записи из других таблиц.

```
create trigger stop
on zakaz
for delete
as
if not exists(select *
              from zakaz a, zakazcik b, prodav c
              where a.id_zakazchik=b.id_zakazchik or a.id_prod=c.id_prod)
delete from zakaz
where id_zakaz=deleted.id_zakaz
else print 'Удаление невозможно'
```

Вывод: в ходе работы были созданы несколько представлений, хранимых процедур и триггеров, что способствовало практическому закреплению навыков работы в MS SQL SERVER.