

УДК

УДК

Утверждено учебным управлением МЭИ

Рецензент: к.т.н., доцент Зубов В.С.

Подготовлено на кафедре прикладной математики

Лабораторный практикум по курсу “Базы данных”. Федин В.А., Бурцев А.П.,
Мотина Н.А., Рогова Н.Г. – М.: Изд-во МЭИ, 2002. – с.

ЛАБОРАТОРНАЯ РАБОТА № 1

Цель:

1. Приобрести практический навык в создании таблиц БД и загрузки их данными.
2. Освоить создание индексов.

Справочный материал:

1. Конспект курса.

Используемые технические средства:

1. SQL-сервер, работающий на компьютере Pentium II под управлением операционной системы Windows 2000 Advanced Server.
2. SQL Query Analyser, работающий на компьютере Pentium под управлением операционной системы Windows 2000 Professional.

Упражнения:

Сценарий входа в систему:

I Вход в сеть.

После загрузки системы WINDOWS 2000 перед пользователем высветится окно с названием “Вход в Windows”. В этом окне необходимо ввести логин и пароль работы с сетью (их необходимо получить у преподавателя). В результате этого на экране появится набор пиктограмм различных WINDOWS-приложений.

II Вход в систему MS SQL Server и диалоговой SQL:

Нажать кнопку “Пуск” и в появившемся меню выбрать пункт “Программы”; при этом появится следующее меню, в котором необходимо найти пункт с названием “Microsoft SQL Server” и выбрать его; в результате этого появляется третье меню (см. рис. 1).

В этом меню необходимо найти пункт с названием программы “SQL Query Analyser” и выполнить на нем один щелчок мыши.

III Установка связи пользователя с базой данных:

На экране появится диалоговое окно Connect to SQL Server.

В этом окне пользователю нужно ввести:

1. Имя сервера (M708-SQL - выбрать из списка имен серверов и щелкнуть левой кнопкой мыши).
2. Имя пользователя (например: STUD1 [Tab]);
3. Пароль (например: stud1 [Tab]); (см. рис. 2)

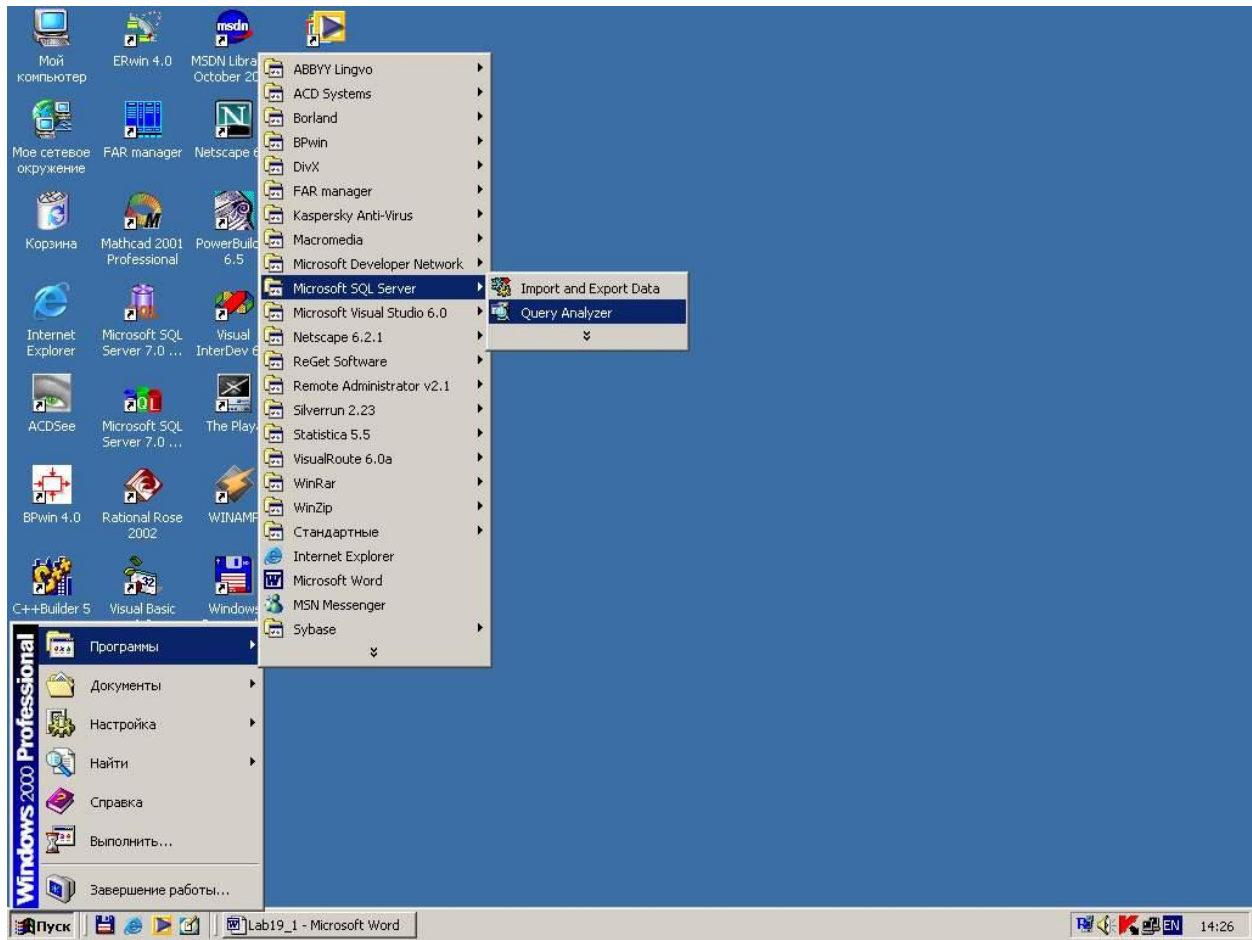


Рис. 1



Рис. 2

Для перехода к вводу следующего имени или пароля используется клавиша Tab. Если все имена и пароль введены правильно, то пользователь получает доступ к основному окну Microsoft SQL Server Query Analyser, состоящее из двух областей (см. рис. 3).

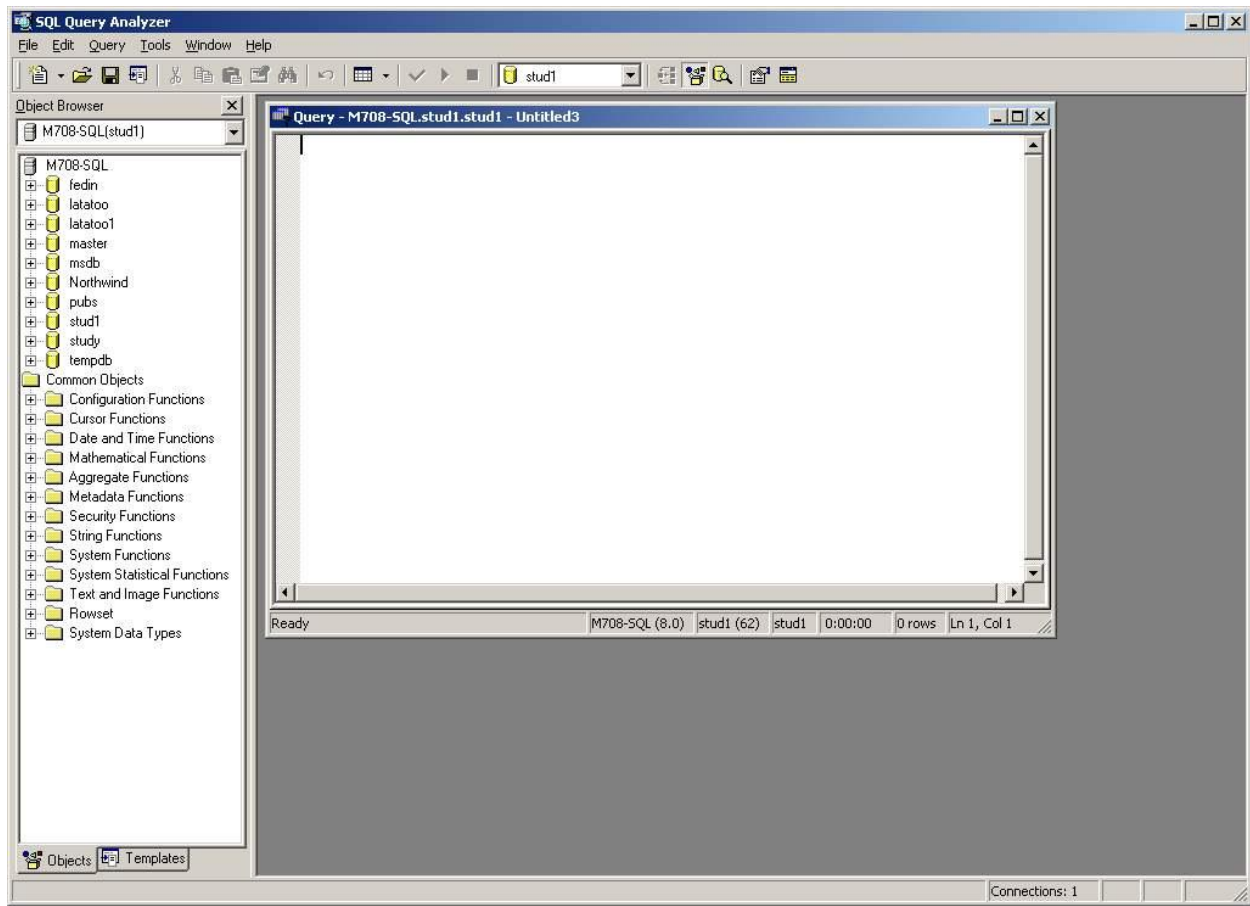


Рис. 3

В левой области отображается структура таблиц. Правая область предназначена для ввода SQL-команд, результаты выполнения запросов отображаются по мере получения информации в специальном окне. В частности, в верхней области для выполнения запросов к БД вводятся SELECT-предложения, а в нижней - выдаются отчеты по этим запросам. По окончании ввода команды в верхней области для ее выполнения нажимается клавиша [F5]. Если команда не помещается в одну строчку, то для переноса нажимается [Enter].

Если SQL-команда вводится неверно, то в нижней области окна появляется сообщение SQL-сервера об ошибке.

На этом сценарий входа в систему заканчивается, и пользователь может начинать свою работу.

1) Создание таблицы с именем poss.

В приложении VI приведена структура информационного объекта "Учащиеся России", состоящая из 12 полей. На основе характеристик каждого из приведенных полей в верхней области окна необходимо ввести следующую команду create table:

```
create table poss
```

```
(nomer integer,
 fio char (40) not null,
 data_rogden char (6),
 pol char (1),
 sem_polog_k char (1),
 spec char (6),
 kat_obuch_k char (2),
 vuz_k integer,
 mp char (2),
 gp char (2),
 mo char (2),
 gok char (2)) [F5]
```

Если пользователь допустит синтаксическую ошибку при наборе этой команды, то в нижней области окна появится сообщение о синтаксической ошибке, а команда create table не выполнится. Пользователю в этом случае необходимо найти и исправить ошибки, а затем вновь запустить эту команду.

Если команда create table выполнится успешно, то в нижней области окна появится сообщение "The command(s) completed successfully". Чтобы убедиться в том, что таблица создана в БД, необходимо в верхней области окна ввести команду sp_help [F5]. При этом в нижней области окна появится список всех таблиц БД.

Аналогично создаются вспомогательная таблица "вузы-города-страны СНГ" и четыре таблицы словаря БД, структуры которых также приведены в приложении VI.

Для создания этих таблиц в верхней области окна необходимо последовательно вводить следующие команды create table:

2) Создание таблицы с именем vuz_gorod:

```
create table vuz_gorod
(cod_vuza integer,
 nazvanie_vuza char (25),
 cod_goroda char (3),
 nazvanie_goroda char (20),
 cod_stran char (2),
 nazvanie_stran char (15)) [F5]
```

3) Создание таблицы с именем vuz:

```
create table vuz
(cod integer,
 uch_zavedenie char (25)) [F5]
```

4) Создание таблицы с именем family:

```
create table family
(cod char (1),
 family_status char (12)) [F5]
```


7) Загрузка таблицы poss.

В таблицу poss загружается 2000 записей; с помощью оператора insert (первая форма) можно загрузить только одну запись. Поэтому для загрузки таблицы poss требуется 2000 операторов insert (первая форма). Если их задавать "вручную", то это займет очень много времени.

Чтобы значительно ускорить процесс загрузки, необходимо на дискете подготовить файлы, каждый из которых содержит набор пакетов. Каждый пакет содержит одну команду insert и признак конца пакета - go (см. конспект курса). Количество пакетов в файле ограничено (порядка 200-250), так как ограничен по памяти буфер ввода команд (операторов, пакетов) SQL Query Analyser.

Таким образом, для загрузки таблицы poss в БД пользователю необходимо:

- а) вставить дискету с подготовленными файлами в дисковод;
- б) с помощью меню "File" последовательно открыть файлы (команда меню Open) на дискете и, тем самым, переписывать их в верхнюю область окна;
- в) запустить на выполнение полученный в верхней области окна набор пакетов, нажав клавишу [F5].

Остальные таблицы БД загружаются аналогично. Чтобы убедиться, что таблицы загружены данными, необходимо в верхней области окна ввести команду: `SELECT * FROM <имя таблицы>` [F5] для каждой таблицы, например: `SELECT * FROM kat_obuch` [F5].

В нижней области окна будет распечатана таблица kat_obuch.

8) Создание индексов для таблицы poss.

Для ускорения поиска в таблице poss необходимо создать индексы по ключевым полям (т.е. по тем полям, по которым чаще всего требуется поиск данных). Это следующие поля:

- а) номер;
- б) пол;
- в) код семейного положения;
- г) специальность;
- д) код категории обучения;
- е) код учебного заведения;
- ж) год приема;
- з) год окончания.

Индексы создаются с помощью команды create index, причем для поля "номер" создается уникальный индекс, а для остальных полей - простые негрупповые индексы.

Таким образом, в верхней области окна необходимо ввести последовательно 8 команд:

- а) `create unique index nomind on poss (nomer)` [F5]
- б) `create index polind on poss (pol)` [F5]

- в) create index semind on poss (sem_polog_k) [F5]
- г) create index specind on poss (spec) [F5]
- д) create index katind on poss (kat_obuch_k) [F5]
- е) create index vuzind on poss (vuz_k) [F5]
- ж) create index gpind on poss (gp) [F5]
- з) create index goind on poss (gok) [F5]

Информацию о созданных индексах можно посмотреть с помощью команды `sp_helpindex <имя таблицы>`.

9) Создание индексов для остальных таблиц БД.

Для вспомогательной таблицы "вузы-города-страны СНГ" индексы создаются для трех ключевых полей:

- а) код учебного заведения;
- б) код города;
- в) код страны.

Для поля "Код учебного заведения" создается уникальный индекс. В верхней области окна необходимо последовательно ввести три команды:

- а) create unique index codvgind on vuz_gorod (cod_vuza) [F5]
- б) create index codgoind on vuz_gorod (cod_goroda) [F5]
- в) create index codvsind on vuz_gorod (cod_stran) [F5]

Для четырех таблиц словаря БД индексы создаются для каждого первого поля их структуры, т.е. поля, содержащего код. Причем, индексы обязательно должны быть уникальными. Поэтому в верхней области окна необходимо последовательно ввести четыре команды:

- а) create unique index codsind on family (cod) [F5]
- б) create unique index codkind on kat_obuch (cod) [F5]
- в) create unique index codgind on gorod (cod) [F5]
- г) create unique index codvind on vuz (cod) [F5]

В том случае, если пользователь обнаружит, что неправильно создал таблицы или индексы, то ему необходимо пересоздать неправильные таблицы или индексы. Для этого ему необходимо сначала уничтожить эти объекты, используя команду `drop`, а затем снова создать эти объекты, используя команду `create`.

Примеры команды `drop` (см. конспект курса):

1. drop table poss
2. drop index poss.nomind

В том случае, если пользователь неправильно загрузит данные в таблицу, то он может воспользоваться командой `truncate table` для быстрого удаления этих данных из таблицы, а затем вновь начать процесс загрузки, например:

```
truncate table poss
```

В качестве заданий к данной лабораторной работе пользователям предлагается создать таблицы БД, соответствующие другим странам СНГ

(Украина, Белоруссия и др.) и имеющие такую же структуру, что и таблица ross. Далее, каждому пользователю необходимо загрузить свою таблицу данными с дискеты, а также создать индексы для своей таблицы.

ЛАБОРАТОРНАЯ РАБОТА № 2

Цель:

1. Научиться использовать предложения оператора SELECT и некоторые функции для выполнения простых запросов к БД в режиме диалогового (интерактивного) SQL.
2. Приобрести практические навыки в использовании операторов INSERT, UPDATE и DELETE.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

1) Выполнение запроса по одной таблице БД, например: выдать названия всех городов России (исключая дубликаты).

В верхней области окна необходимо ввести:

```
select distinct nazvanie_goroda from vuz_gorod where cod_stran='01' [F5]
```

где '01' - это код России.

В результате выполнения этого запроса в нижней области окна появится отчет, представляющий собой список названий городов России (одно название города в одной строчке). Для просмотра отчета по запросу необходимо пользоваться мышкой и полосами прокрутки.

Замечание: Если в команде будет допущена ошибка, то вместо отчета будет выдано окно с сообщением об ошибке (см. рис.5).

2) Выполнение запроса по одной таблице БД с использованием функции BETWEEN, например:

выдать фамилии всех студентов и студентов-заочников России.

В верхней области окна необходимо ввести:

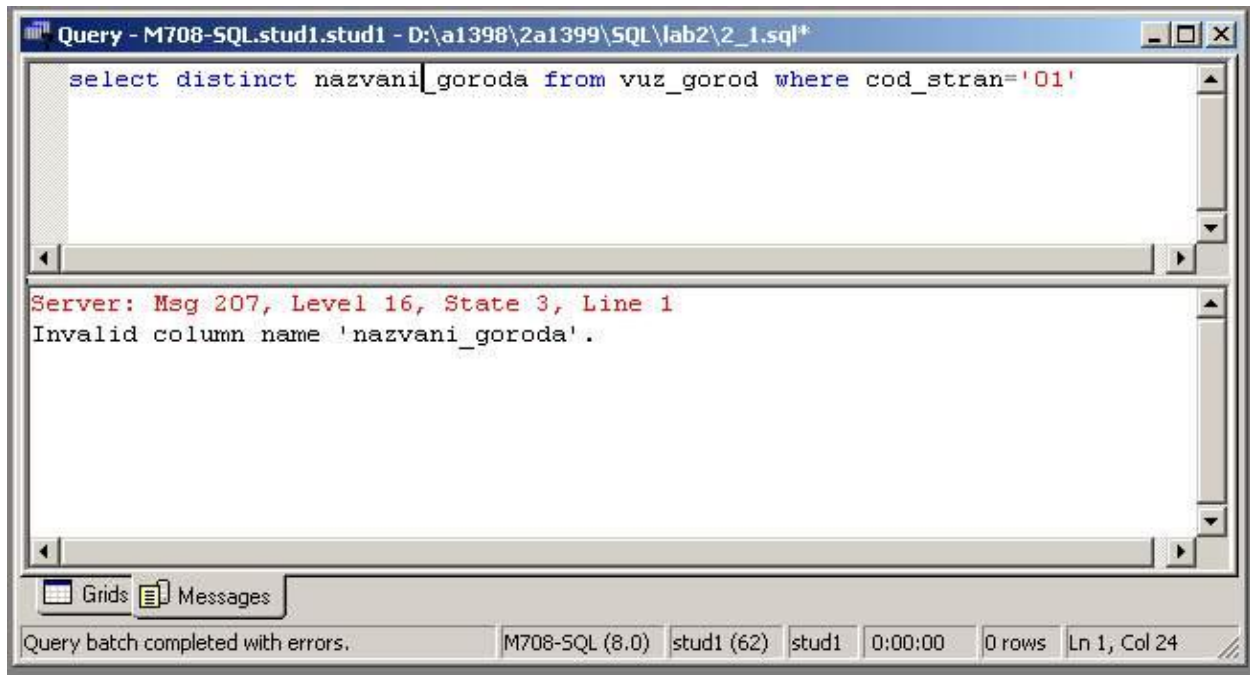


Рис. 5

`select fio from poss where kat_obuch_k between '03' and '04' [F5]`

где '03' и '04' - это коды, соответствующие категориям обучения "студент" и "студент-заочник".

Отчет представлен в нижней части окна (см. рис. 6).

3) Выполнение запроса по одной таблице БД с использованием функции IN, например:

выдать данные по учащимся России, закончившим учебу в 95, 97, 98 годах.

В верхней области окна необходимо ввести:

`select * from poss where gok in ('95', '97', '98') [F5].`

Отчет будет представлен в нижней части окна (см. рис. 7).

4) Выполнение запроса по одной таблице БД с использованием функции LIKE, например:

выдать сведения по учащимся России с фамилией КУТЕПОВ (А).

В верхней области окна необходимо ввести:

`select * from poss where fio like '%КУТЕПОВ%' [F5]`

Отчет будет представлен в нижней части окна (см. рис. 8).

5) Выполнение запроса по одной таблице БД с использованием функции NULL, например:

выдать данные по учащимся России, у которых отсутствует учетный номер.

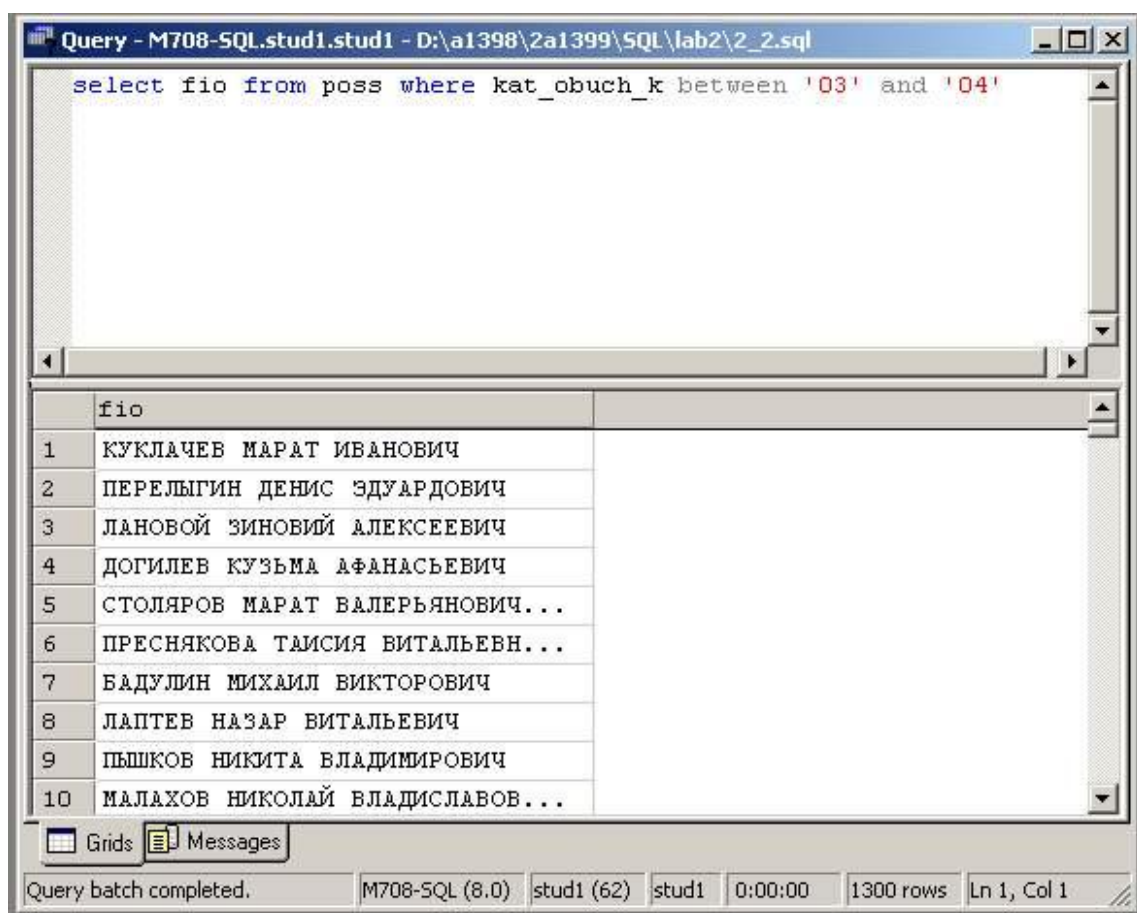


Рис. 6

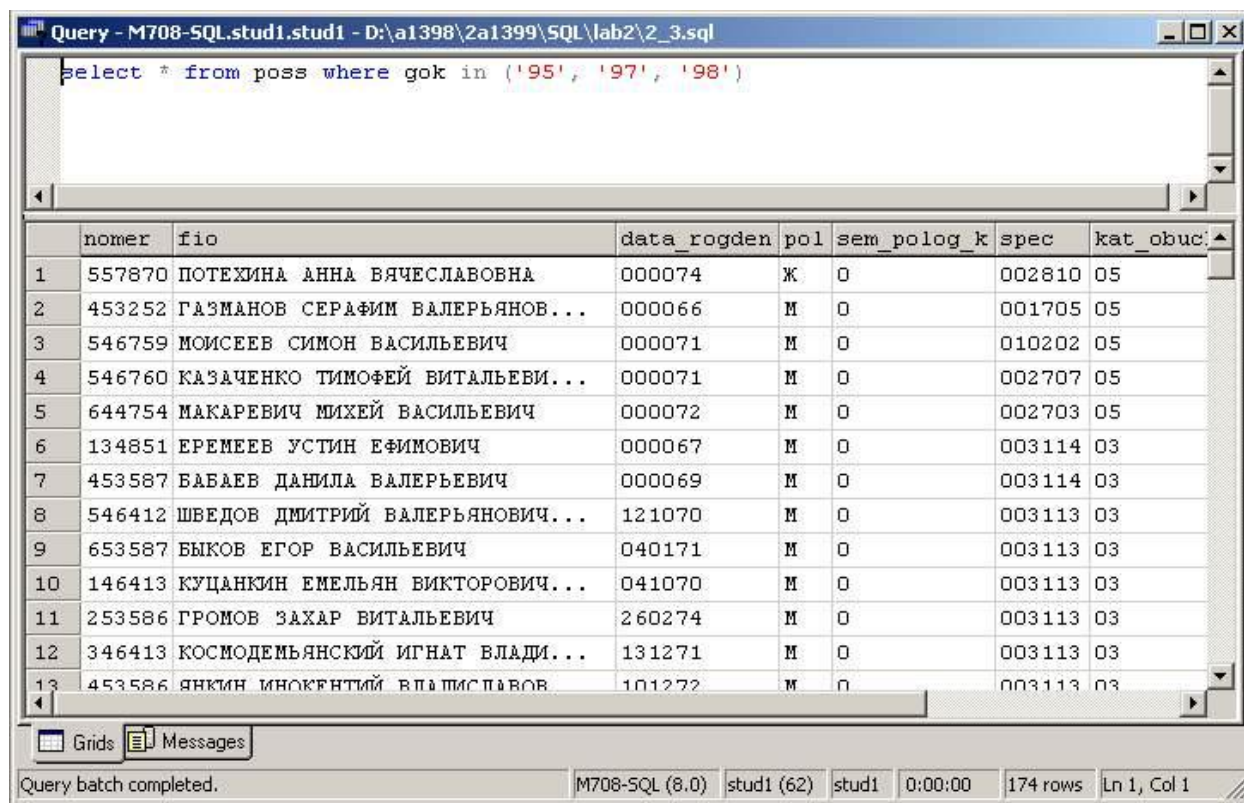


Рис. 7

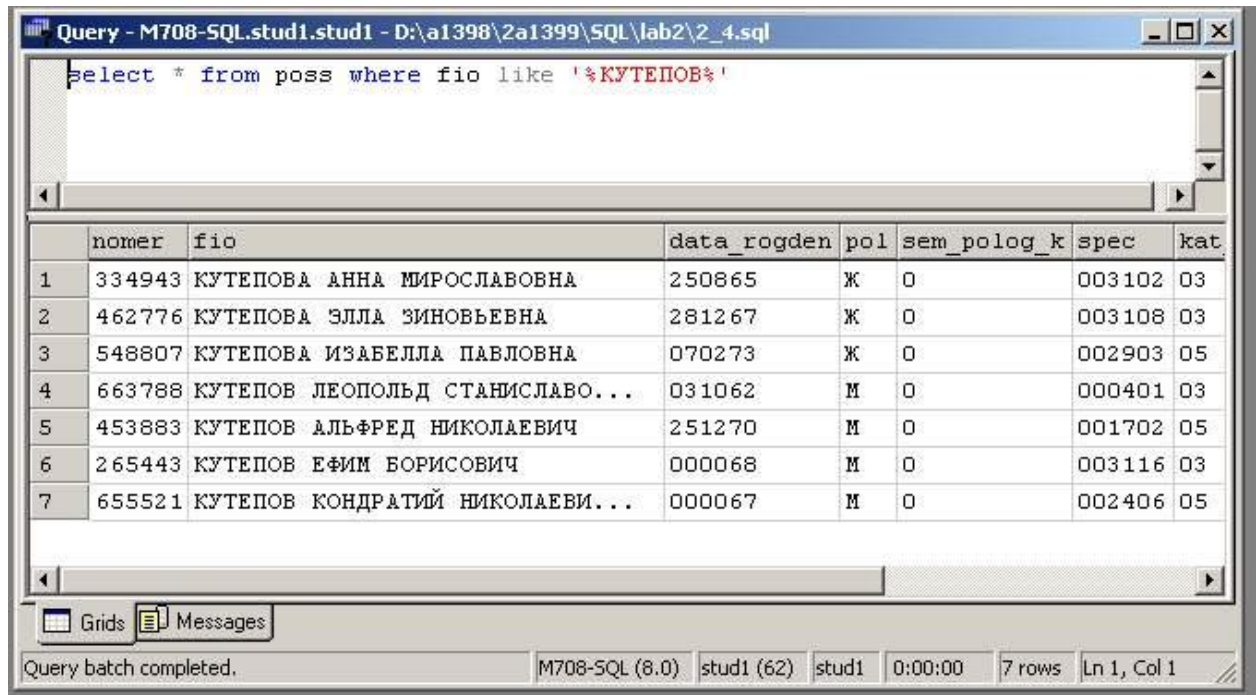


Рис. 8

В верхней области окна необходимо ввести:
 select * from poss where nomer is null [F5]

Отчет будет представлен в нижней части окна (см. рис. 9).

Замечание: Функции BETWEEN, IN, LIKE, NULL могут использоваться с операцией NOT.

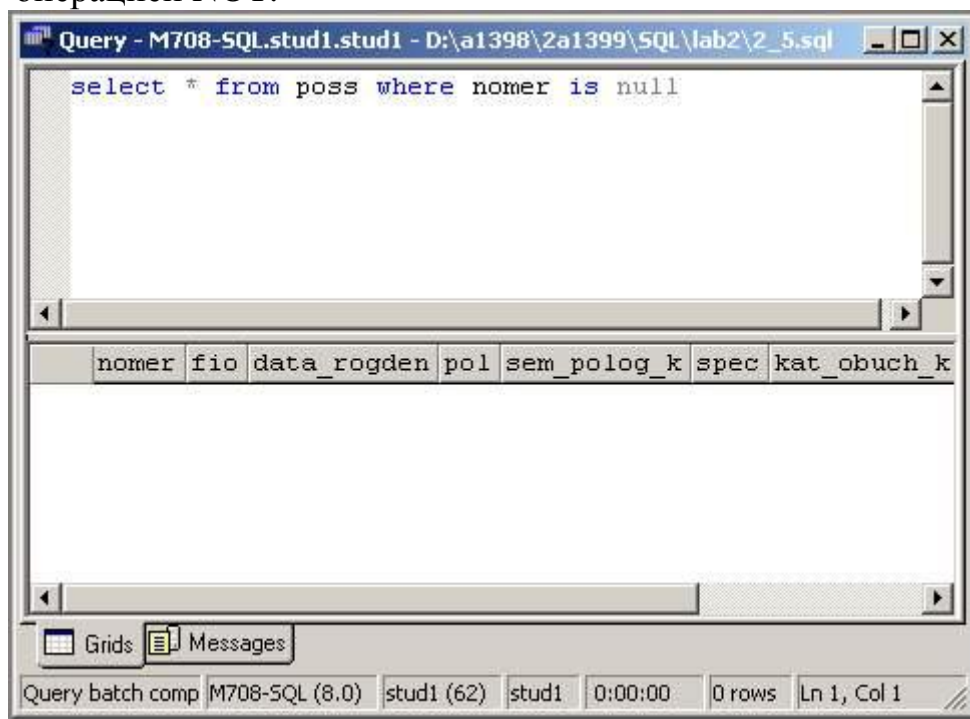


Рис. 9

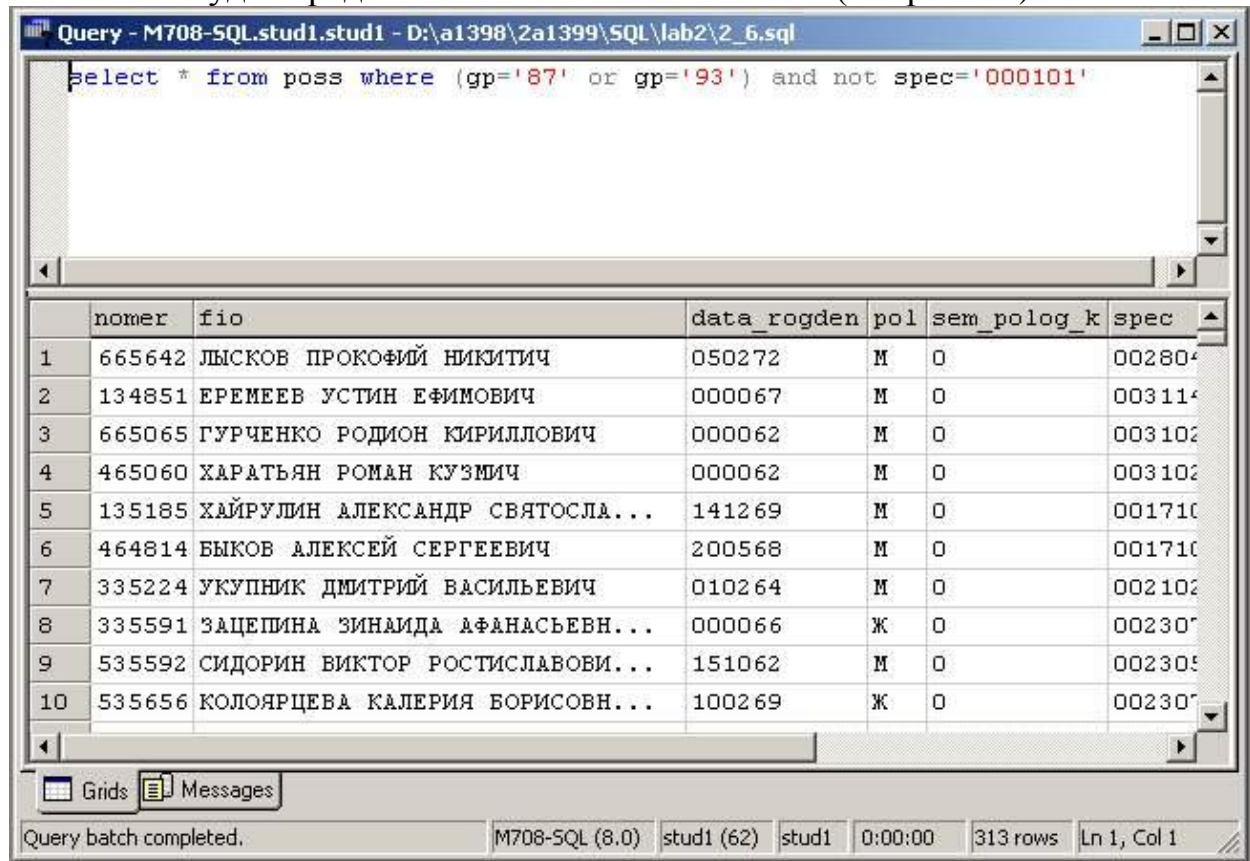
6) Выполнение запроса по одной таблице БД с использованием сложного условия, например:

выдать данные по учащимся России, принятым на учебу в 87 и 93 годах, за исключением специальности 0101 (математика).

В верхней области окна необходимо ввести:

`select * from poss where (gp='87' or gp='93') and not spec='000101' [F5]`

Отчет будет представлен в нижней части окна (см. рис. 10).



The screenshot shows the M708-SQL studio window. The title bar reads "Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab2\2_6.sql". The query editor contains the SQL statement: `select * from poss where (gp='87' or gp='93') and not spec='000101'`. Below the editor, a table of results is displayed with 10 rows and 7 columns. The columns are: `nomer`, `fio`, `data_rogden`, `pol`, `sem_polog_k`, and `spec`. The status bar at the bottom indicates "Query batch completed.", "M708-SQL (8.0)", "stud1 (62)", "stud1", "0:00:00", "313 rows", and "Ln 1, Col 1".

	nomer	fio	data_rogden	pol	sem_polog_k	spec
1	665642	ЛЫСКОВ ПРОКОФИЙ НИКИТИЧ	050272	М	0	002804
2	134851	ЕРЕМЕЕВ УСТИН ЕФИМОВИЧ	000067	М	0	003114
3	665065	ГУРЧЕНКО РОДИОН КИРИЛЛОВИЧ	000062	М	0	003102
4	465060	ХАРАТЬЯН РОМАН КУЗМИЧ	000062	М	0	003102
5	135185	ХАЙРУЛИН АЛЕКСАНДР СВЯТОСЛА...	141269	М	0	001710
6	464814	БЫКОВ АЛЕКСЕЙ СЕРГЕЕВИЧ	200568	М	0	001710
7	335224	УКУПНИК ДМИТРИЙ ВАСИЛЬЕВИЧ	010264	М	0	002102
8	335591	ЗАЦЕПИНА ЗИНАИДА АФАНАСЬЕВН...	000066	Ж	0	002307
9	535592	СИДОРИН ВИКТОР РОСТИСЛАВОВИ...	151062	М	0	002309
10	535656	КОЛЯРЦЕВА КАЛЕРИЯ БОРИСОВН...	100269	Ж	0	002307

Рис. 10

7) Выполнение запроса по одной таблице БД с сортировкой записей, например:

выдать коды и названия всех учебных заведений Белоруссии, отсортировав записи по возрастанию кодов.

В верхней области окна необходимо ввести:

`select cod_vuza, nazvanie_vuza from vuz_gorod where cod_stran='03' order by cod_vuza [F5]`

Отчет будет представлен в нижней части окна (см. рис. 11).

Замечание: Во фразе `order by` вместо имени поля (`cod_vuza`) можно использовать порядковый номер позиции этого поля в списке выборки (для `cod_vuza` этот номер равен 1), т.е. можно указать `order by 1`. Другими словами, запрос будет выглядеть так:
`select cod_vuza, nazvanie_vuza`


```
from vuz_gorod
where cod_stran='03' order by 1
```

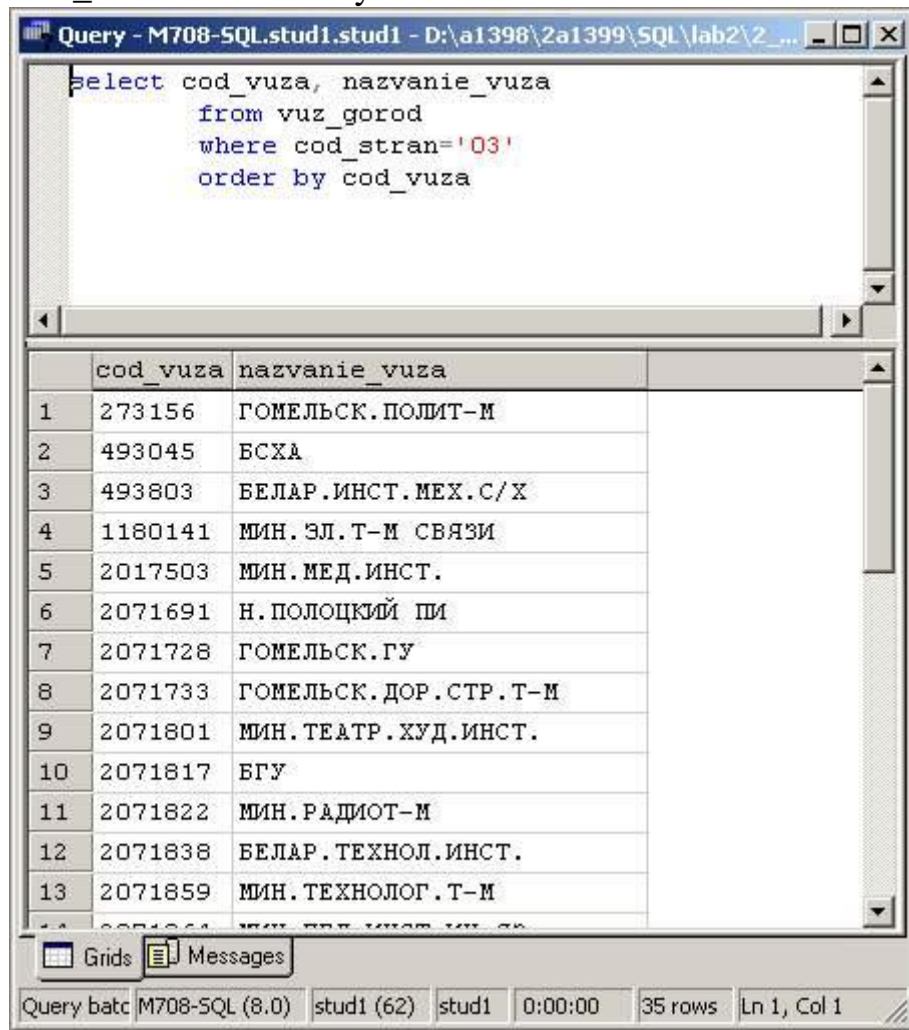


Рис. 11

8) Выполнение запроса и выдача количества записей по одной таблице БД с использованием функции COUNT(*), например:

выдать количество учебных заведений России.

В верхней области окна необходимо ввести:

```
select count(*) from vuz_gorod where cod_stran='01' [F5]
```

Отчет будет представлен в нижней части окна (см. рис. 12).

Замечание: Для использования агрегатных функций AVG и SUM необходимо в структуре таблицы иметь поля числового типа, например: количество учащихся. Агрегатные функции count, min, max могут быть применены к полям любого типа.

9) Выполнение запроса по одной таблице БД с группировкой записей, например:

выдать названия городов и учебных заведений России, сгруппировав записи по городам и учебным заведениям.

В верхней области окна необходимо ввести:

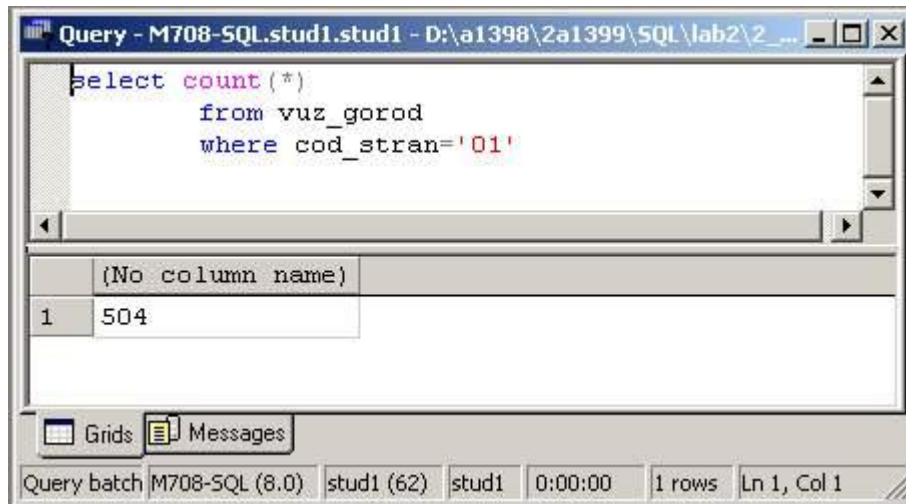


Рис. 12

`select nazvanie_goroda, nazvanie_vuza from vuz_gorod where cod_stran='01'`
`group by nazvanie_goroda, nazvanie_vuza` [F5]

Отчет будет представлен в нижней части окна (см. рис. 13). Помимо отображения результатов выполнения запроса в табличном виде SQL Query Analyser позволяет просматривать графическое отображение процесса выполнения запроса (см. рис. 14). Чтобы его увидеть, необходимо выбрать пункт меню "Query", а затем "Show Execution Plan" или нажать Ctrl+K.

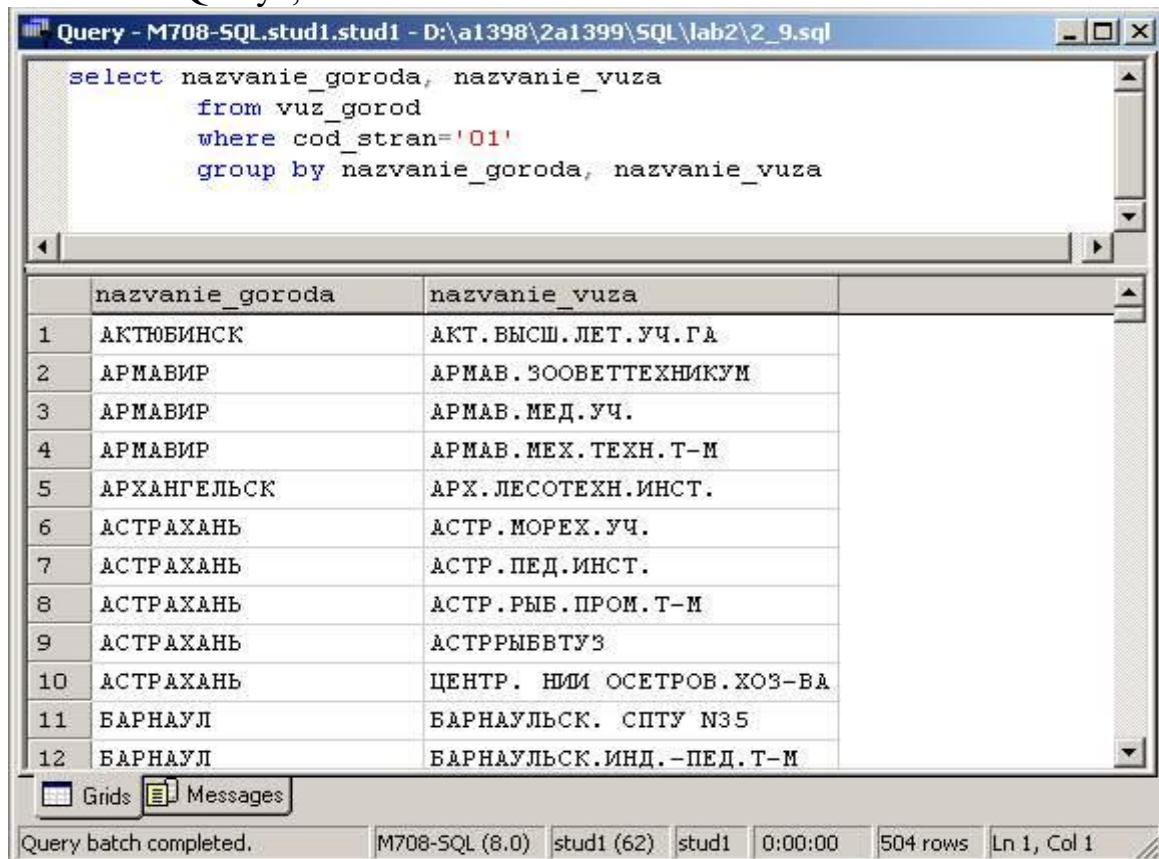


Рис. 13

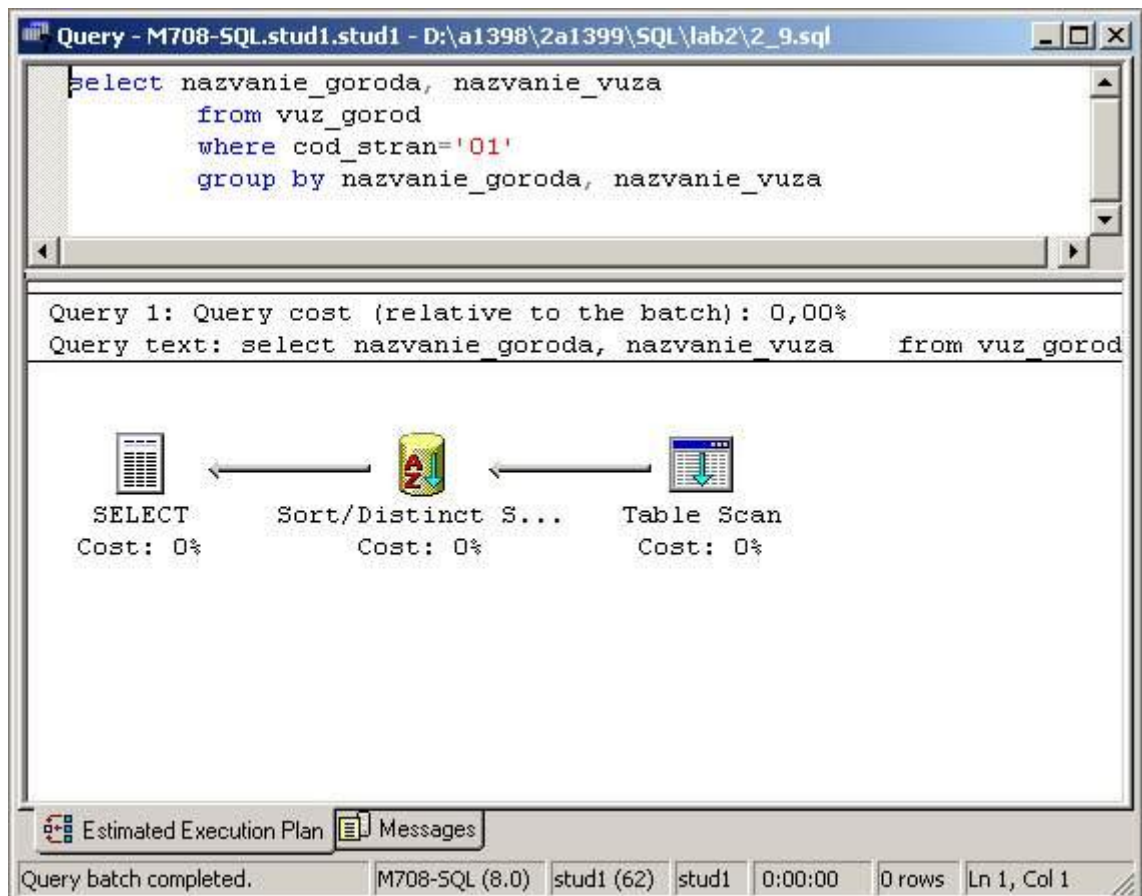


Рис. 14

10) Выполнение запроса по одной таблице БД с использованием фразы **HAVING**, например:

выдать учетные номера учащихся, встречающиеся больше одного раза.

В верхней области окна необходимо ввести:

```
select nomer from poss group by nomer having count(*)>1 [F5]
```

Отчет будет представлен в нижней части окна (см. рис. 15).

Упражнения на использование операторов update, insert и delete.

11) Установка в качестве значения поля fio ваших фамилии, имени и отчества вместо ФИО учащегося с минимальным учетным номером.

Это делается за 3 шага:

- ♦ Определение минимального учетного номера учащегося.

В верхней области окна необходимо ввести:

```
select min (nomer) from poss [F5]
```

Результатом этого запроса в нижней части окна будет число, равное минимальному номеру. Это число необходимо использовать на втором шаге.

- ♦ Изменение значения поля fio в записи с найденным номером.

В верхней области окна необходимо ввести:

```
update poss set fio='<фамилия> <имя> <отчество>' where
nomer=<минимальный номер> [F5]
```

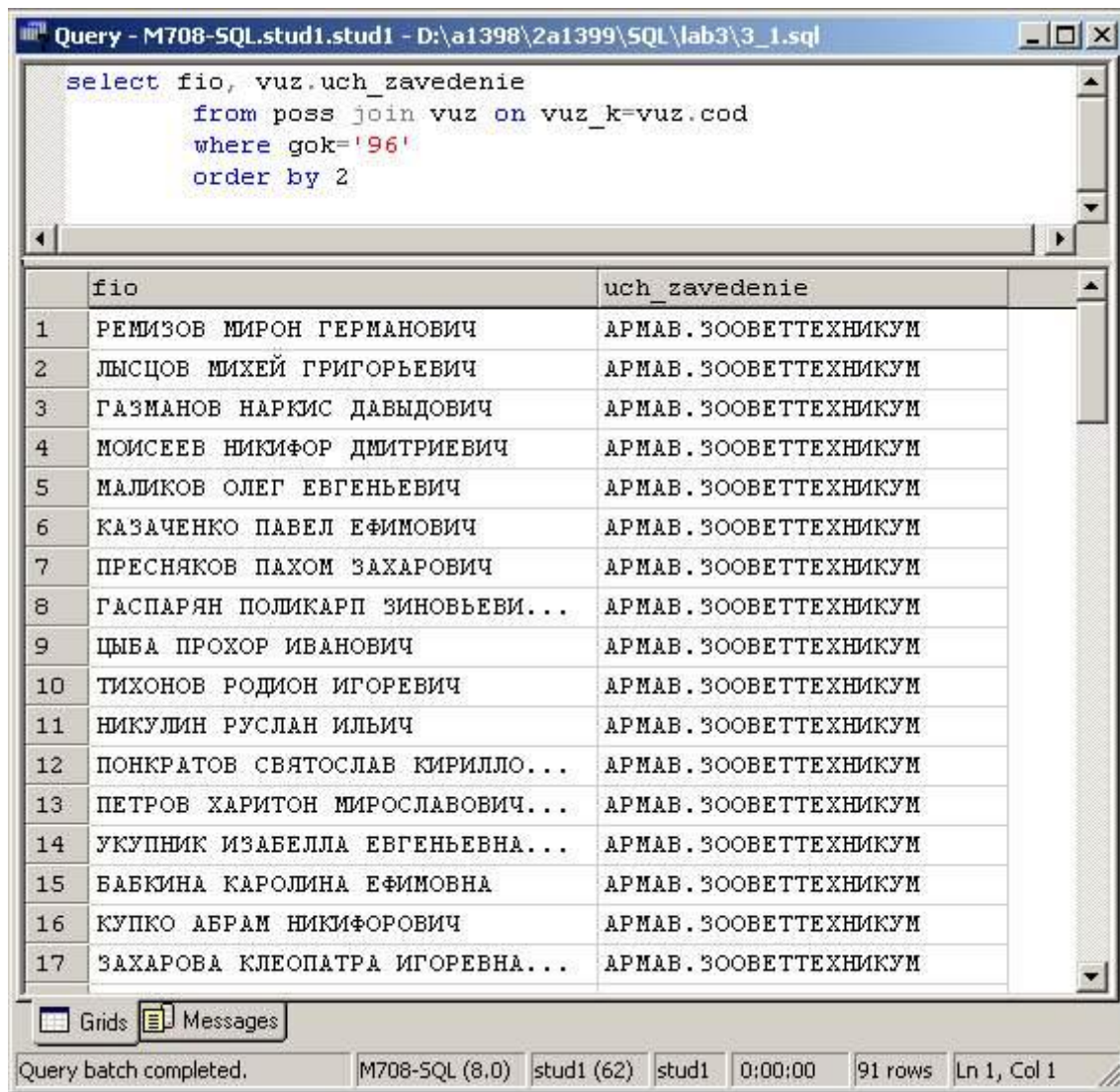



Рис. 15

- ♦ Для того, чтобы убедиться в том, что в поле fio появилось новое значение, нужно выполнить следующий запрос:
 select fio from poss where nomer=<мин. номер> [F5]

12) Удаление записи из таблицы БД по заданным фамилии, имени и отчеству, например:

удалить из БД данные о Журавлеве Аверьяне Алексеевиче. Это делается за три шага:

- ♦ Прежде чем удалить запись с этими данными необходимо сделать запрос, указав в качестве условия выборки ФИО Журавлев Аверьян Алексеевич. В верхней области окна надо ввести:
 select * from poss where fio = 'ЖУРАВЛЕВ АВЕРЬЯН АЛЕКСЕЕВИЧ'
 [F5]

В нижней области окна в результате появится запись, включающая значения по всем полям структуры таблицы poss:

534305 ЖУРАВЛЕВ АВЕРЬЯН АЛЕКСЕЕВИЧ 000073 М 0 001001 05
0129000 08 90 03 95

- ◆ Для удаления этой записи в верхней области окна необходимо ввести:
delete from poss where fio = 'ЖУРАВЛЕВ АВЕРЬЯН АЛЕКСЕЕВИЧ'
[F5]
- ◆ Чтобы убедиться, что запись удалена, нужно повторить запрос первого шага.

13) Добавление новой записи в таблицу БД, например:

добавить только что удаленную запись. Это делается за два шага:

- ◆ В верхней области окна надо ввести:
insert poss values (534305, 'ЖУРАВЛЕВ АВЕРЬЯН АЛЕКСЕЕВИЧ',
'000073', 'М', '0', '001001', '05', 0129000, '08', '90', '03', '95') [F5]
- ◆ Для того, чтобы убедиться в том, что новая запись добавлена, нужно повторить запрос из предыдущего упражнения, т.е. ввести:
select * from poss where fio = 'ЖУРАВЛЕВ АВЕРЬЯН АЛЕКСЕЕВИЧ'
[F5]

Варианты заданий на выполнение лабораторной работы № 2 приведены в приложении I.

ЛАБОРАТОРНАЯ РАБОТА № 3

Цель:

1. Приобрести практические навыки использования операции соединения нескольких таблиц БД при выполнении сложных списковых запросов.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

1) Выполнение запроса по двум таблицам БД с использованием операции соединения, например:

выдать фамилии и названия учебных заведений по выпускникам России 1996 года, отсортировав данные по учебным заведениям.

Для выполнения этого запроса необходимо использовать три условия, указанных в описании операции Join (соединения таблиц) в конспекте курса. Так как в задании по запросу требуется выдать названия учебных заведений, а в таблице poss такого поля нет (есть только код учебного заведения), то,

кроме таблицы poss, необходимо использовать таблицу vuz, где это поле есть.

Поэтому в списке выборки имя второго поля должно быть составным, т.е. с именем таблицы vuz (составное имя поля - это <имя табл.>.<имя поля>), в предложении from должны быть записаны два имени таблиц (poss и vuz), а в предложении where должно быть указано условие соединения этих двух таблиц по полю код учебного заведения.

Таким образом, в верхней области окна необходимо ввести:

```
select fio, vuz.uch_zavedenie
```

```
from poss, vuz
```

```
where gok='96'
```

```
and vuz_k=vuz.cod
```

```
order by 2          [F5]
```

или

```
select fio, vuz.uch_zavedenie
```

```
from poss join vuz on vuz_k=vuz.cod
```

```
where gok='96'
```

```
order by 2          [F5]
```

Отчет будет представлен в нижней области окна (см. рис. 16).



Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab3\3_1.sql

```
select fio, vuz.uch_zavedenie
  from poss join vuz on vuz_k=vuz.cod
 where gok='96'
 order by 2
```

	fio	uch_zavedenie
1	РЕМИЗОВ МИРОН ГЕРМАНОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
2	ЛЫСЦОВ МИХЕЙ ГРИГОРЬЕВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
3	ГАЗМАНОВ НАРКИС ДАВЫДОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
4	МОИСЕЕВ НИКИФОР ДМИТРИЕВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
5	МАЛИКОВ ОЛЕГ ЕВГЕНЬЕВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
6	КАЗАЧЕНКО ПАВЕЛ ЕФИМОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
7	ПРЕСНЯКОВ ПАХОМ ЗАХАРОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
8	ГАСПАРЯН ПОЛИКАРП ЗИНОВЬЕВИ...	АРМАВ.ЗООВЕТТЕХНИКУМ
9	ЦЫБА ПРОХОР ИВАНОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
10	ТИХОНОВ РОДИОН ИГОРЕВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
11	НИКУЛИН РУСЛАН ИЛЬИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
12	ПОНКРАТОВ СВЯТОСЛАВ КИРИЛЛО...	АРМАВ.ЗООВЕТТЕХНИКУМ
13	ПЕТРОВ ХАРИТОН МИРОСЛАВОВИЧ...	АРМАВ.ЗООВЕТТЕХНИКУМ
14	УКУПНИК ИЗАБЕЛЛА ЕВГЕНЬЕВНА...	АРМАВ.ЗООВЕТТЕХНИКУМ
15	БАВКИНА КАРОЛИНА ЕФИМОВНА	АРМАВ.ЗООВЕТТЕХНИКУМ
16	КУПКО АБРАМ НИКИФОРОВИЧ	АРМАВ.ЗООВЕТТЕХНИКУМ
17	ЗАХАРОВА КЛЕОПАТРА ИГОРЕВНА...	АРМАВ.ЗООВЕТТЕХНИКУМ

Query batch completed. M708-SQL (8.0) stud1 (62) stud1 0:00:00 91 rows Ln 1, Col 1

Рис. 16

2) Выполнение запроса по трем таблицам БД с использованием операции соединения, например:

выдать фамилии, даты рождения, названия учебных заведений, пол, специальности, категории обучения, месяцы окончания и годы окончания по учащимся, принятым на учебу в 1991 году; данные отсортировать по учебным заведениям и специальностям.

В этом запросе требуется выдать значения двух полей, не принадлежащих структуре таблицы poss, а именно: названия учебных заведений и названия категорий обучения. Поэтому в списке выборки для этих полей должны быть указаны составные имена (т.е. с именем таблицы), в предложении from должны быть записаны три имени таблиц, а в предложении where - два условия соединения таблиц (таблицы poss с таблицей vuz по полю код учебного заведения; таблицы poss с таблицей kat_obuch по полю код категории обучения).

Таким образом, в верхней области окна необходимо ввести:

```
select fio, data_rogden, vuz.uch_zavedenie,
pol, spec, kat_obuch. kat_obucheniya, mo, gok
from poss, vuz, kat_obuch
where gp='91'
and vuz_k=vuz.cod
and kat_obuch_k=kat_obuch.cod
order by 3,5                                [F5]
```

или

```
select fio, data_rogden, vuz.uch_zavedenie,
pol, spec, kat_obuch. kat_obucheniya, mo, gok
from poss join vuz on vuz_k=vuz.cod
      join kat_obuch on kat_obuch_k=kat_obuch.cod
where gp='91'
order by 3,5
```

Отчет будет представлен в нижней области окна (см. рис. 17).

3) Выполнение запроса по четырем таблицам БД со сложным условием, например:

выдать фамилии, даты рождения, названия учебных заведений, пол, семейные положения, специальности, категории обучения, месяцы приема, годы приема, месяцы окончания и годы окончания по выпускникам России 1993 и 1995 г.г. по специальностям от 0701 до 0705; данные отсортировать по годам окончания, специальностям и учебным заведениям.

В этом запросе требуется выдать значения трех полей, не принадлежащих структуре таблицы poss, а именно: названия учебных заведений, названия семейных положений и названия категорий обучения. Поэтому в списке выборки для этих полей должны быть указаны составные

Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab3\3_2.sql

```

select fio, data_rogden, vuz.uch_zavedenie,
       pol, spec, kat_obuch. kat_obucheniya, mo, gok
from poss join vuz on vuz_k=vuz.cod
       join kat_obuch on kat_obuch_k=kat_obuch.cod
where gp='91'
order by 3,5

```

	fio	data_rogden	uch_zavedenie	pol	spec
1	ПОНКРАТОВ СВЯТОСЛАВ КИРИЛЛО...	000071	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031
2	ПЕТРОВ ХАРИТОН МИРОСЛАВОВИЧ...	000070	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031
3	УКУПНИК ИЗАБЕЛЛА ЕВГЕНЬЕВНА...	000072	АРМАВ.ЗООВЕТТЕХНИКУМ	Ж	0031
4	БАБКИНА КАРОЛИНА ЕФИМОВНА	000070	АРМАВ.ЗООВЕТТЕХНИКУМ	Ж	0031
5	КУПКО АБРАМ НИКИФОРОВИЧ	000070	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031
6	ЗАХАРОВА КЛЕОПАТРА ИГОРЕВНА...	000068	АРМАВ.ЗООВЕТТЕХНИКУМ	Ж	0031
7	МАКОВ АГАФОН ПЛАТОНОВИЧ	000071	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031
8	ДОГИЛЕВА ЛАДА КУЗМИНИЧНА	000070	АРМАВ.ЗООВЕТТЕХНИКУМ	Ж	0031
9	МЕТВЕЕВА ДИАНА ДАВЫДОВНА	241266	АРМАВ.ЗООВЕТТЕХНИКУМ	Ж	0031
10	НИКУЛИН РУСЛАН ИЛЬИЧ	000075	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031
11	РЕМИЗОВ МИРОН ГЕРМАНОВИЧ	000077	АРМАВ.ЗООВЕТТЕХНИКУМ	М	0031

Query batch completed. M708-SQL (8.0) stud1 (62) stud1 0:00:00 174 rows Ln 6, Col 14

Рис. 17

имена (т.е. с именем таблицы), в предложении from должны быть записаны четыре имени таблиц, а в предложении where - три условия соединения таблиц (таблицы poss с таблицей vuz по полю код учебного заведения; таблицы poss с таблицей family по полю код семейного положения; таблицы poss с таблицей kat_obuch по полю код категории обучения).

Таким образом, в верхней области окна необходимо ввести:

```

select fio, data_rogden, vuz.uch_zavedenie, pol,
       family.family_status, spec, kat_obuch. kat_obucheniya,
       mp, gp, mo, gok
from poss, vuz, family, kat_obuch
       where (gok='93' or gok='95')
       and spec between '000701' and '000705'
       and vuz_k=vuz.cod
       and sem_polog_k=family.cod
       and kat_obuch_k=kat_obuch.cod
order by 11, 6, 3

```

или

```

select fio, data_rogden, vuz.uch_zavedenie, pol,
       family.family_status, spec, kat_obuch. kat_obucheniya,
       mp, gp, mo, gok
from poss join vuz on vuz_k=vuz.cod

```



```

join family on sem_polog_k=family.cod
join kat_obuch on kat_obuch_k=kat_obuch.cod
where (gok='93' or gok='95')
      and spec between '000701' and '000705'
order by 11, 6, 3 [F5]

```

Отчет будет представлен в нижней области окна (см. рис. 18).

Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab3\3_3.sql

```

select fio, data_rogden, vuz. uch_zavedenie, pol,
       family.family_status, spec, kat_obuch. kat_obucheniya,
       mp, gp, mo, gok
from poss join vuz on vuz_k=vuz.cod
      join family on sem_polog_k=family.cod
      join kat_obuch on kat_obuch_k=kat_obuch.cod
where (gok='93' or gok='95')
      and spec between '000701' and '000705'
order by 11, 6, 3

```

	fio	data_rogden	uch_zavedenie	pol	family_
1	ХЛЕБАЛИН ЛЕОНИД ГЕОРГИЕВИЧ	050168	МИИТ	М	ХОЛОСТ
2	ЛЫСКОВ НИКОДИМ ЕФИМОВИЧ	260867	МИИТ	М	ХОЛОСТ
3	ХРИТОВ ОНИСИМ ЗАХАРОВИЧ	051068	МИИТ	М	ХОЛОСТ
4	МОРОЗОВ ПАНТЕЛЕЙ ЗИНОВЬЕВИЧ...	050666	МИИТ	М	ХОЛОСТ
5	КАЛАШНИКОВ ПЕТР ИВАНОВИЧ	021068	МИИТ	М	ХОЛОСТ
6	МУРУГОВ ПОТАП ИГОРЕВИЧ	101165	МИИТ	М	ХОЛОСТ

Query batch completed. M708-SQL (8.0) stud1 (62) stud1 0:00:00 6 rows Ln 7, Col 27

Рис. 18

Варианты заданий на выполнение лабораторной работы № 3 приведены в приложении II.

ЛАБОРАТОРНАЯ РАБОТА № 4

Цель:

1. Приобрести практические навыки выполнения запросов с подзапросами.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

Для использования подзапросов (оператор select) в операторах манипулирования данными (select, update, insert, delete) употребляют три

формата предложения where, указанных в конспекте курса (in, any/all, exists). Наиболее часто используется первый формат с функцией in.

1) Выполнение запроса по четырем таблицам БД с использованием подзапроса, например:

выдать список студентов, обучаемых в городе Москве, принятых на учебу в 1989 г.; схема выдачи отчета должна включать следующие поля: ФИО, дата рождения, специальность, категория обучения, название учебного заведения, месяц окончания, год окончания; данные отсортировать по учебным заведениям и специальностям.

В этом запросе требуется, во-первых: выдать значения двух полей, не принадлежащих структуре таблицы poss, а именно: названия учебных заведений и названия категорий обучений; во-вторых: учебные заведения должны быть московскими, т.е. необходимо дополнительно сделать подзапрос по таблице vuz_gorod, указав в списке выборки код учебного заведения, а в условии выборки - город Москва.

Поэтому в списке выборки запроса для полей учебное заведение и категория обучения должны быть указаны составные имена (т.е. с именем таблицы), в предложении from должны быть записаны три имени таблиц, а в предложении where для поля код учебного заведения должны быть использованы функция IN и подзапрос, а также должны быть указаны два условия соединения таблиц (таблицы poss с таблицей vuz по полю код учебного заведения; таблицы poss с таблицей kat_obuch по полю код категории обучения).

Таким образом, в верхней области окна необходимо ввести:

```
select fio, data_rogden, spec, kat_obuch.kat_obucheniya,
vuz.uch_zavedenie, mo, gok
from poss join kat_obuch on kat_obuch_k=kat_obuch.cod
      join vuz on vuz_k=vuz.cod
where vuz_k in
(select cod_vuza from vuz_gorod
      where cod_goroda='001')
and kat_obuch_k between '03' and '04'
and gp='89'
order by 5, 3 [F5]
```

где '001' - код города Москвы.

Отчет будет представлен в нижней области окна (см. рис. 19).

2) Выполнение запроса по пяти таблицам БД с использованием подзапроса со сложным условием выборки данных, например:

выдать список аспирантов и стажеров, обучаемых в городах Москва, Санкт-Петербург и Воронеж и заканчивающих учебу в 1993 и 1995 г.г.; схема выдачи отчета должна включать поля: ФИО, дата рождения, название

Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab4\4_1.sql

```

select fio, data_rogden, spec, kat_obuch.kat_obucheniya,
vuz.uch_zavedenie, mo, gok
from poss join kat_obuch on kat_obuch_k=kat_obuch.cod
join vuz on vuz_k=vuz.cod
where vuz_k in
(select cod_vuza from vuz_gorod
where cod_goroda='001')
and kat_obuch_k between '03' and '04'
and gp='89'
order by 5, 3

```

	fio	data_rogden	spec	kat_obucheniya	uch_zave
1	ЛОНОВАЯ ЭЛЛА ВИКТОРОВНА	000066	003107	СТУДЕНТ	МОСК.БЕ
2	СКАЧКОВ СЕРГЕЙ ВАДИМОВИЧ	000063	003107	СТУДЕНТ	МОСК.БЕ
3	МИШАЧЕВ СЕВАСТЬЯН БОРИСОВИЧ...	000063	003108	СТУДЕНТ	МОСК.БЕ
4	ПРОНИН ТИМУР ВАЛЕРЬЯНОВИЧ	000064	003108	СТУДЕНТ	МОСК.БЕ
5	ЯРЫГИН УЛЬЯН ВАСИЛЬЕВИЧ	000071	003108	СТУДЕНТ	МОСК.БЕ
6	БАВКИНА ТАМАРА ВАСИЛЬЕВНА	000066	003108	СТУДЕНТ	МОСК.БЕ
7	КРАПИН ФЕДОТ ВИКТОРОВИЧ	000065	003108	СТУДЕНТ	МОСК.БЕ
8	ЛЫСКОВ ФОМА ВИТАЛЬЕВИЧ	030861	003108	СТУДЕНТ	МОСК.БЕ
9	ХРИТОВ ХРИСТОФОР ВЛАДИМИРОВ...	120667	003108	СТУДЕНТ	МОСК.БЕ

Query batch completed. M708-SQL (8.0) stud1 (62) stud1 0:00:00 74 rows Ln 9, Col 13

Рис. 19

учебного заведения, пол, семейное положение, специальность, категория обучения, месяц приема, год приема, месяц окончания, год окончания; данные отсортировать по учебным заведениям и годам окончания.

В этом запросе требуется, во-первых: выдать значения трех полей, не принадлежащих структуре таблицы poss, а именно: названия учебных заведений, названия семейных положений и названия категорий обучения; во-вторых: учебные заведения должны быть московскими, Санкт-Петербургскими и воронежскими, т.е. необходимо сделать подзапрос по таблице vuz_gorod, указав в списке выборки код учебного заведения, а в условии выборки - три значения городов - Москва, Санкт-Петербург и Воронеж.

Поэтому в списке выборки запроса для полей учебное заведение, семейное положение и категория обучения должны быть указаны составные имена (т.е. с именем таблицы), в предложении from должны быть записаны четыре имени таблиц, а в предложении where для поля код учебного заведения должны быть использованы функция IN и подзапрос, а также должны быть указаны три условия соединения таблиц (таблицы poss с таблицей vuz по полю код учебного заведения; таблицы poss с таблицей

family по полю код семейного положения; таблицы poss с таблицей kat_obuch по полю код категории обучения).

Таким образом, в верхней области окна необходимо ввести:

```
select fio, data_rogden, vuz.uch_zavedenie, pol,
family.family_status, spec, kat_obuch.kat_obucheniya,
mp, gp, mo, gok
from poss join vuz on vuz_k=vuz.cod
      join family on sem_polog_k=family.cod
      join kat_obuch on kat_obuch_k=kat_obuch.cod
where vuz_k in
      (select cod_vuza from vuz_gorod
       where cod_goroda='001' or cod_goroda='002' or
       cod_goroda='006')
and (kat_obuch_k between '07' and '08' or
     kat_obuch_k between '10' and '11')
and gok in ('93', '95')
order by 3, 11
```

[F5]

The screenshot shows the M708-SQL stud1 interface. The top window displays the SQL query entered, which is identical to the one in the previous block. Below the query window, a table of results is displayed. The table has 6 columns: fio, data_rogden, uch_zavedenie, pol, and fami. The results are sorted by the 3rd column (uch_zavedenie) and then by the 11th column (fami). The first 10 rows are visible, showing names, birth dates, institutions, genders, and family names.

	fio	data_rogden	uch_zavedenie	pol	fami
1	МОРДУКОВ ВИТАЛИЙ ЗИНОВЬЕВИЧ...	000000	ВСЕС. КАРДИОЛОГИЧ. ЦЕНТР	М	ХОЛО
2	ДРУЖИНИН ГЕРМАН ИЛЬИЧ	000000	ВСЕС.Н/Ц.ХИРУРГИИ	М	ХОЛО
3	БОБРОВ ЕФРЕМ ЛЕОНИДОВИЧ	000000	ВСЕС.НИИ ГЛАЗНЫХ БОЛЕЗНЕЙ	М	ХОЛО
4	САХАРОВ ЕЛИЗАР ЛЬВОВИЧ	000000	ВСЕС.НИИ ГРИППА	М	ХОЛО
5	ЕРЕМЕЕВ ЕМЕЛЬЯН ВАЛЕРЬЯНОВИ...	000000	ВСЕС.НИИ РАСТЕНИЕВОДСТВА	М	ХОЛО
6	КАЗАКОВ ИСИДОР ДАВЫДОВИЧ	150161	ИНСТ. МАШИНОВЕДЕН. АНСССР	М	ЖЕНА
7	НАГОРНОВ КИРИЛЛ ДМИТРИЕВИЧ	150161	ИНСТ. МАШИНОВЕДЕН. АНСССР	М	ЖЕНА
8	МИНАЕВ КОНСТАНТИН ЕВГЕНЬЕВИ...	150161	ИНСТ. МАШИНОВЕДЕН. АНСССР	М	ЖЕНА
9	ПОПОВИЧ АЛЬФРЕД ВЛАДИМИРОВИ...	140554	ИНСТ. НЕЙРОХИРУРГИИ	М	ХОЛО
10	МАСЛОВ АНАТОЛИЙ ВЛАДЛЕНОВИЧ...	000057	ИНСТ. НЕЙРОХИРУРГИИ	М	ХОЛО

At the bottom of the interface, there is a status bar showing "Query batch completed.", "M708-SQL (8.0)", "stud1 (62)", "stud1", "0:00:00", "49 rows", and "Ln 1, Col 1".

Рис. 20

где '001' - код города Москвы;

'002' - код города Санкт-Петербург;
 '006' - код города Воронеж;
 '07' - код, соответствующий категории обучения "аспирант";
 '08' - код, соответствующий категории обучения "аспирант-заочник";
 '10' и '11' - коды, соответствующие категориям обучения "повышение квалификации" и "стажер".

Отчет будет представлен в нижней области окна (см. рис. 20).

Варианты заданий на выполнение лабораторной работы № 4 приведены в приложении III.

ЛАБОРАТОРНАЯ РАБОТА № 5

Цель:

1. Научиться выполнять одномерные статистические запросы по одной и по нескольким таблицам БД.
2. Практически освоить создание и использование простых хранимых процедур для выполнения запросов.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

Реляционные СУБД специально приспособлены для выполнения списковых и персональных запросов (в результирующей форме отчета содержится совокупность документов (записей) в виде списка значений атрибутов (полей)). Проблемы возникают при выполнении статистических запросов.

Статистическим называется запрос, в результирующей форме отчета которого содержится перечень различных значений одного, двух и более атрибутов с количественными характеристиками этих значений. Количественная характеристика отдельного значения атрибута (поля) означает количество экземпляров этого значения, встретившихся в результате выполнения запроса. Например, количество учащихся по каждому учебному заведению России.

Если количественные характеристики выдаются по одному атрибуту, то такой запрос называется одномерным статистическим запросом. Если количественные характеристики выдаются по двум атрибутам, т.е. выдаются количественные характеристики различных значений по первому атрибуту и, кроме того, для каждого значения первого атрибута выдаются количественные характеристики различных значений по второму атрибуту, то такой запрос называется двумерным статистическим запросом. Например, количество учащихся по каждому учебному заведению России и количество учащихся по каждому шифру специальностей в каждом из этих учебных заведений.

И так далее, если количественные характеристики выдаются по n атрибутам, такой запрос называется n -мерным статистическим запросом. Кроме этого, в статистических запросах может использоваться операция соединения таблиц, поэтому статистические данные могут выдаваться по атрибутам таблиц, связанных с той таблицей, по которой выполняется запрос. Например, количество учащихся по каждому городу России (для этого запроса необходимо соединение таблиц `ross` и `vuz_gorod` по полю код учебного заведения).

Первый Вариант запроса к базе данных

Выдать статистику по приему учащихся 1990 и 1991 годов в России по всем специальностям. Отчет должен начинаться с общей цифры - всего по приему 90-91 г.г. Далее - таблица, включающая перечень шифров специальностей с цифрами, не равными нулю (цифра - это количество учащихся по указанной специальности; если количество учащихся по специальности равно нулю, то этот шифр специальности не включается в отчет).

В этом запросе требуется, во-первых: выдать общее количество учащихся, принятых на учебу в 1990-91 г.г.; во-вторых: распределить это количество по разным шифрам специальностей, т.е. определить количественные характеристики для каждого шифра (значения) специальностей.

Другими словами, необходимо выполнить два запроса:

1) выполнить запрос с условием выборки по году приема (значения '90' и '91'), используя в списке выборки агрегатную функцию `count(*)`, предназначенную для подсчета общего количества записей по запросу;

2) выполнить запрос с условием выборки по году приема (значения '90' и '91'), используя предложение `group by` для группирования значений по полю "специальность" и агрегатную функцию `count(*)` в списке выборки, предназначенную для оценки количества значений специальностей в каждой группе.

Таким образом, в верхней области окна необходимо ввести:

- 1) `select count(*) from poss`
`where gp='90' or gp='91' [F5]`

В нижней области окна будет представлена цифра, равная общему количеству учащихся, принятых на учебу в 1990 и 1991 г.г. (см. рис. 21).

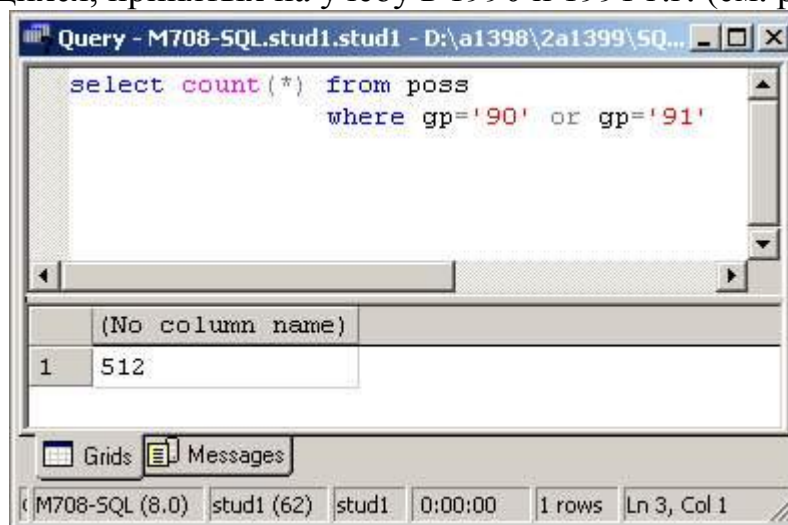


Рис. 21

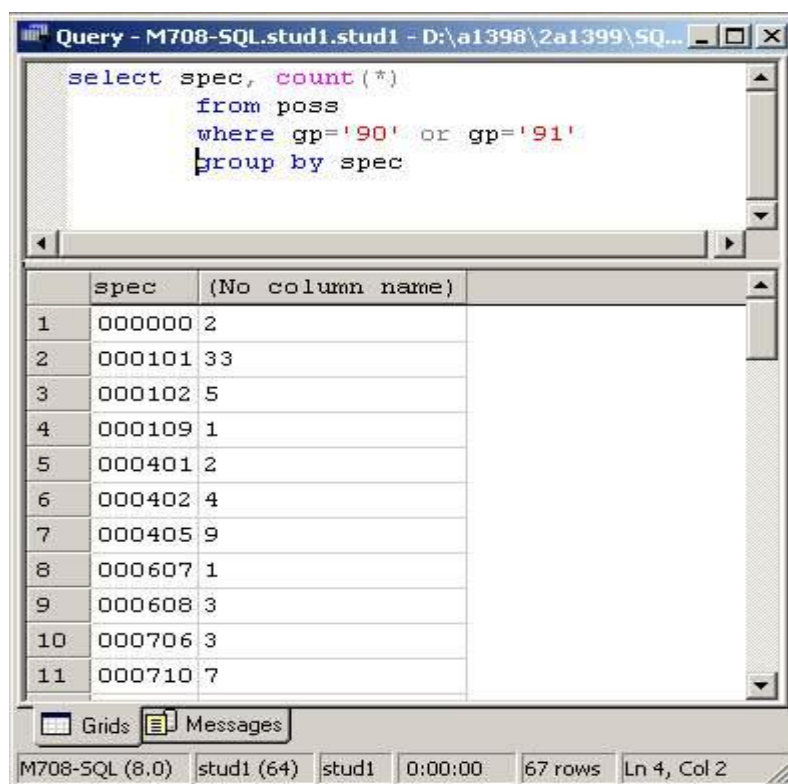


Рис. 22

- 2) `select spec, count(*) from poss`
`where gp='90' or gp='91'`
`group by spec [F5]`

Отчет будет представлен в нижней области окна (см. рис. 22).

Замечание. Имя поля, по которому выдается статистика, задается и в списке выборки, и в структуре group by.

Второй вариант запроса к базе данных

Выдать статистику по выпускникам 1996 года в России по всем учебным заведениям. Отчет должен начинаться с общей цифры - всего по выпуску 1996 года. Далее - таблица, включающая перечень названий учебных заведений с цифрами, не равными нулю (цифра - это количество выпускников 1996 года по указанному учебному заведению; если количество выпускников 1996 года в учебном заведении равно нулю, то это учебное заведение в отчет не включается).

Этот запрос выполняется аналогично запросу первого варианта, за исключением того, что в отчете необходимо выдать значения атрибута (названия учебных заведений), не принадлежащего структуре таблицы poss. Другими словами, в этом запросе необходимо в предложении where (условии выборки) добавить условие соединения двух таблиц (poss и vuz) по полю код учебного заведения, в предложении from имя таблицы vuz, а в списке выборки и в предложении group by указать составное имя для поля название учебного заведения (с именем таблицы).

Поэтому здесь необходимо выполнить два следующих запроса:

1) выполнить запрос с условием выборки по году окончания (значение '96'), используя в списке выборки агрегатную функцию count(*), предназначенную для подсчета общего количества записей по запросу;

2) выполнить запрос с условием выборки по году окончания (значение '96') и с условием соединения таблиц poss и vuz по полю код учебного заведения, используя предложение group by для группирования значений по полю "название учебного заведения" и агрегатную функцию count(*) в списке выборки, предназначенную для оценки количества значений (названий учебных заведений) в каждой группе.

Таким образом, в верхней области окна необходимо ввести:

```
1)  select count(*) from poss
      where gok='96' [F5]
```

В нижней области окна будет представлена цифра, равная общему количеству выпускников 1996 года в России (см. рис. 23).

```
2)  select vuz.uch_zavedenie, count(*)
      from poss join vuz on vuz_k=vuz.cod
      where gok='96'
      group by vuz.uch_zavedenie [F5]
```

Отчет будет представлен в нижней области окна (см. рис. 24).

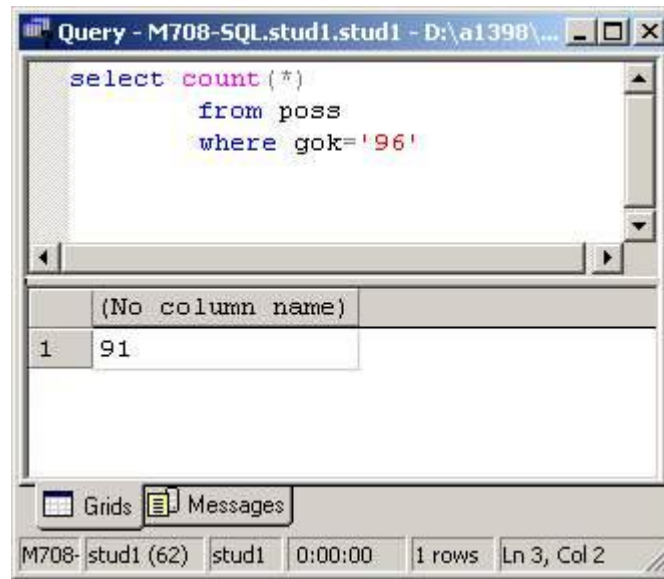


Рис. 23

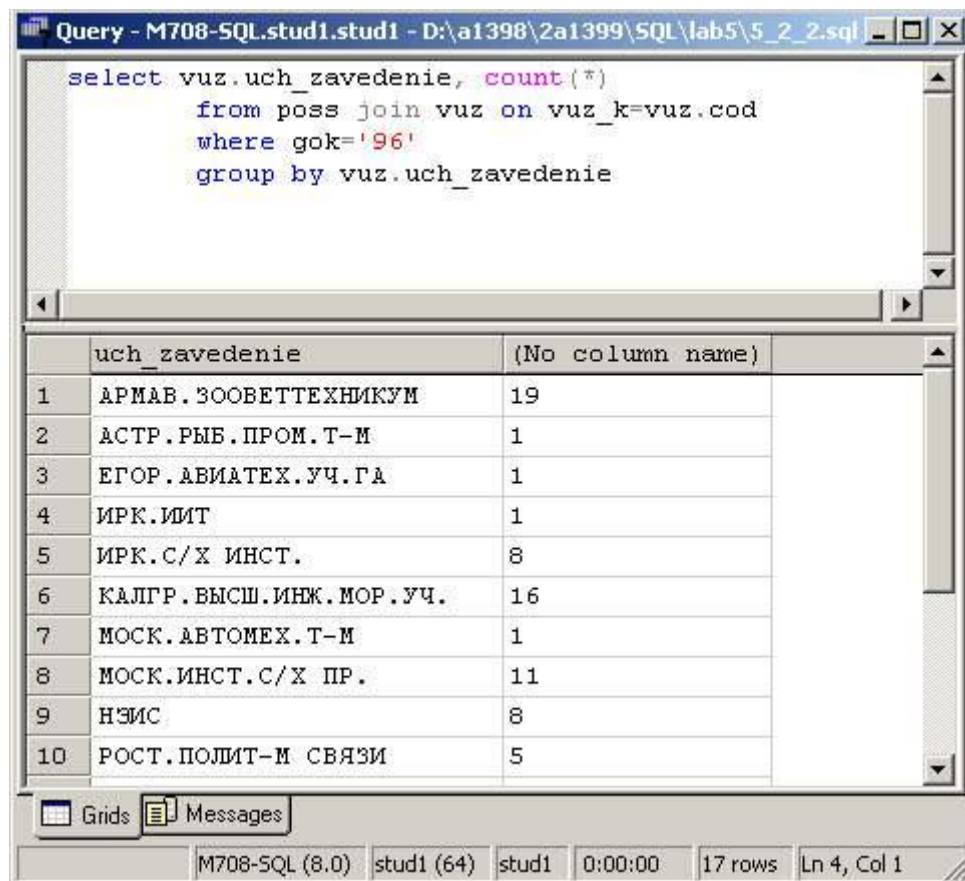


Рис. 24

Третий вариант запроса к базе данных

Выдать статистику по учащимся во всех городах России. Отчет должен начинаться с общей цифры - всего учащихся по России. Далее - таблица, включающая перечень названий городов с цифрами, не равными нулю

(цифра - это количество учащихся в указанном городе; если количество учащихся в городе равно нулю, то этот город в отчет не включается).

Этот запрос выполняется аналогично запросу второго варианта, за исключением того, что вместо названий учебных заведений необходимо в отчете выдать названия городов. Для этого нужно вместо соединения таблиц `ross` и `vuz` по полю код учебного заведения выполнить соединение таблиц `ross` и `vuz_gorod` по полю код учебного заведения, так как поле "название города" принадлежит структуре таблицы `vuz_gorod`.

Поэтому здесь необходимо выполнить два следующих запроса:

1) выполнить запрос без условия выборки, используя в списке выборки агрегатную функцию `count(*)` для подсчета общего количества учащихся России;

2) выполнить запрос с условием соединения таблиц `ross` и `vuz_gorod` по полю код учебного заведения, используя предложение `group by` для группирования значений по полю "название города" и агрегатную функцию `count(*)` в списке выборки для оценки количества значений (названий городов) в каждой группе.

Таким образом, в верхней области окна необходимо ввести:

1) `select count(*) from ross` [F5]

В нижней области окна будет представлена цифра, равная общему количеству учащихся в России (см. рис. 25).

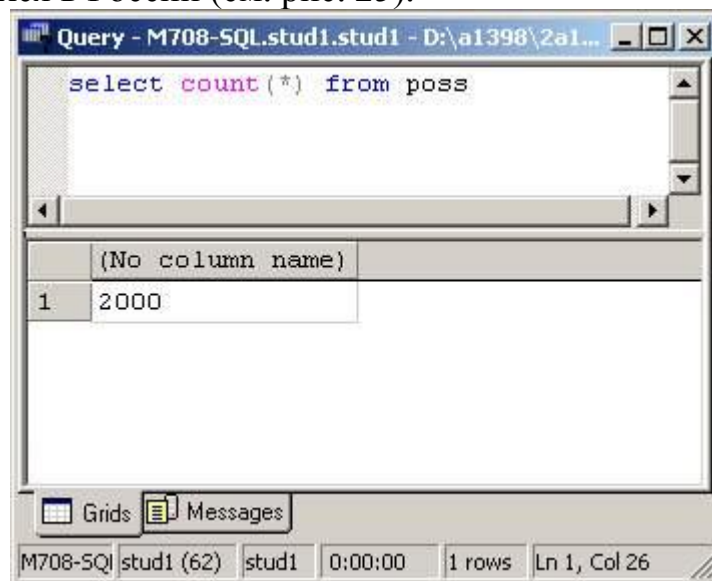


Рис. 25

2) `select vuz_gorod.nazvanie_goroda, count(*)`
`from ross join vuz_gorod on vuz_k=vuz_gorod.cod_vuza`
`group by vuz_gorod.nazvanie_goroda` [F5]

Отчет будет представлен в нижней области окна (см. рис. 26).

The screenshot shows a window titled "Query - M708-SQL.stud1.stud1 - D:\a1398\2a1399\SQL\lab5\5_3_2.sql". The query text is:

```
select vuz_gorod.nazvanie_goroda, count(*)
  from poss join vuz_gorod on vuz_k=vuz_gorod.cod_vuza
 group by vuz_gorod.nazvanie_goroda
```

Below the query, a table displays the results. The table has two columns: "nazvanie_goroda" and "(No column name)". The data is as follows:

	nazvanie_goroda	(No column name)
1	АРМАВИР	47
2	АСТРАХАНЬ	95
3	ВОЛГОГРАД	50
4	ЕГОРЬЕВСК	50
5	ИВАНОВО	66
6	ИРКУТСК	116
7	КАЗАНЬ	4
8	КАЛИНИНГРАД	82
9	КОКИНО	1

At the bottom of the window, there are tabs for "Grids" and "Messages". The status bar at the bottom indicates "M708-SQL (8.0) stud1 (65) stud1 0:00:00 22 rows Ln 2, Col 30".

Рис. 26

Возможности SQL Query Analyser в плане выполнения сложных запросов к БД весьма ограничены (ограничения связаны с трудоемкостью набора длинных предложений SELECT и с реляционной ограниченностью языка SQL). Одним из способов преодоления ограничений является использование хранимых процедур, описанных в конспекте курса.

Для выполнения сложного запроса требуется, как правило, набирать очень длинное предложение SELECT (несколько предложений SELECT). На это необходимо довольно большой промежуток времени. Если сложный запрос делается очень часто, то его предложение SELECT (предложения SELECT) можно оформить как хранимую процедуру (ввести один раз), а затем для выполнения этого запроса запустить эту процедуру.

Для выполнения одномерных статистических запросов в SQL Query Analyser требуется задавать два предложения SELECT и получать два результата. Чтобы объединить оба результата в один отчет, необходимо задать оба предложения SELECT в одной хранимой процедуре.

Пример создания процедуры (смотри первый вариант запроса):

```
create procedure zapr1 as
  select count(*) from poss where gp='90' or gp='91'
  select spec, count(*) from poss
  where gp='90' or gp='91' group by spec      [F5]
```

Для выполнения этого запроса нужно запустить эту процедуру, т.е. в верхней области окна необходимо ввести:

```
zapr1      [F5]
```

Отчет будет представлен в нижней области окна (см. рис. 27).

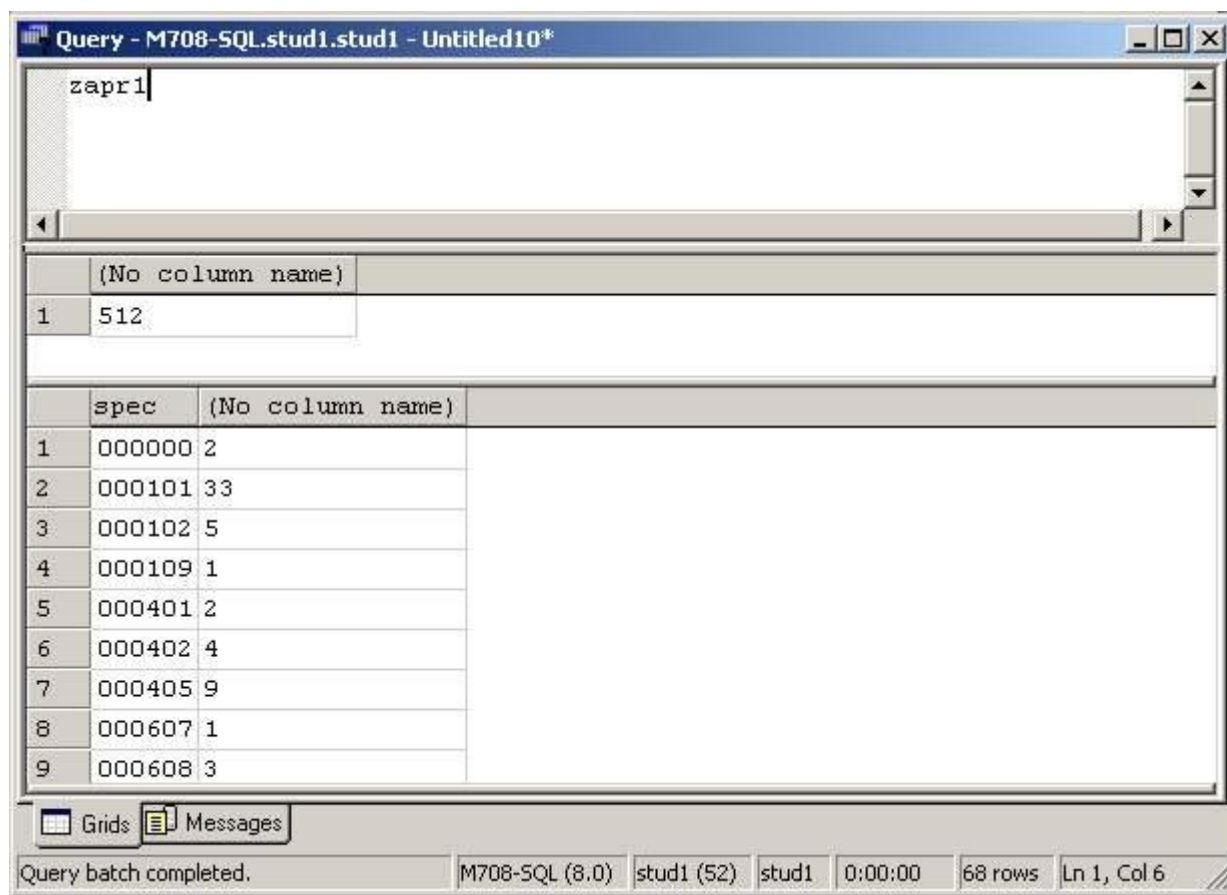


Рис. 27

Замечание. Если во время набора команды `create procedure` будут допущены синтаксические ошибки, то после нажатия [F5] прекомпилятор в нижней половине окна выдаст сообщение об ошибке.

В этом случае процедура не будет загружена в БД. Убедиться в том, что процедура загружена в БД, можно с помощью команды `sp_help`.

Варианты заданий на выполнение лабораторной работы № 5 приведены в приложении IV. Задание необходимо выполнить двумя способами: 1) в диалоговом SQL; 2) с помощью хранимой процедуры.

ЛАБОРАТОРНАЯ РАБОТА № 6

Цель:

1. Практическое освоение операторов `grant/revoke`, `union` и `create view` для получения доступа к защищенным данным.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

1) Создать входные имена SQL-сервера VAF и APB из БД master (имя sa).

Для этого в верхней области окна необходимо ввести:

sp_addlogin VAF, teacher, stud1 [F5]

sp_addlogin APB, freedom, stud1 [F5]

2) Создать пользователей БД MPEI1 из БД MPEI1 (имя STUD1).

Для этого в верхней области окна необходимо ввести:

sp_adduser VAF [F5]

sp_adduser APB [F5]

3) Предоставить командное полномочие на выполнение команды create table пользователю VAF. Для этого в верхней области окна необходимо ввести:

grant create table to VAF [F5]

4) Предоставить объектное полномочие на выполнение команд select, update, insert в таблице, созданной пользователем STUD1, пользователю APB. Для этого в верхней области окна необходимо ввести:

grant select, update, insert, delete
on stud1.poss to APB [F5]

5) Выполнить запрос по двум информационным таблицам (например: poss и ukr): выдать данные по учащимся России и Украины, заканчивающим учебу в 95, 97 и 98 годах.

Для того, чтобы выполнить этот запрос, например пользователю APB, который имеет объектное полномочие на работу с таблицей poss, но не имеет полномочий на таблицу ukr, необходимо:

а) владельцу БД stud1 (STUD1) создать и загрузить таблицу с именем ukr (по аналогии с таблицей poss);

б) предоставить объектное полномочие на выполнение команды select в таблице ukr пользователю APB, т.е. ввести команду:

grant select on stud1.ukr to APB [F5]

в) выйти из SQL Query Analyser (disconnect) и войти в SQL Query Analyser с именем APB;

г) так как таблицы poss и ukr имеют одинаковую структуру, то для этого запроса можно использовать оператор union, т.е. ввести команду:

select * from stud1.poss where gok in ('95', '97', '98')
union

select * from stud1.ukr where gok in ('95', '97', '98') [F5]

В нижней области окна будет предоставлен отчет (см. рис. 28).

Query - COMP.stud1.APB - E:\TODAY\bd\2a1399\SQL\lab6\6_3.sql

```
select * from stud1.ross where gok in ('95', '97', '98')
union
select * from stud1.ukr where gok in ('95', '97', '98')
```

	nomer	fio	data_rogden	pol	sem_polog
1	134480	ЛЫСЦОВ АНТИП КИРИЛЛОВИЧ	200167	М	0
2	134481	ГАСПАРЯН ВИКЕНТИЙ МАКСИМОВИ...	120864	М	0
3	134833	ИВАНКИН КИРИЛЛ АФАНАСЬЕВИЧ	000064	М	0
4	134851	ЕРЕМЕЕВ УСТИН ЕФИМОВИЧ	000067	М	0
5	134880	ПРОНИН ЮЛИАН РОМАНОВИЧ	010468	М	0
6	134979	ХРАМОВ НАЗАР ВЛАДЛЕНОВИЧ	061058	М	0
7	135145	КУЗЬМИН АРТУР ОЛЕГОВИЧ	000066	М	0
8	135926	МОРГАЧЕВ ЛУКЬЯН ВАЛЕНТИНОВИ...	111168	М	0
9	137229	СИДОРКИН АЛЕКСЕЙ ОЛЕГОВИЧ	050866	М	0
10	146413	КУЦАНКИН ЕМЕЛЬЯН ВИКТОРОВИЧ...	041070	М	0
11	146414	ВЛАСОВ ЛЕВ ГАВРИЛОВИЧ	300461	М	0

Grids Messages

COMP (8.0) APB (51) stud1 0:00:00 174 rows Ln 3, Col 21

Рис. 28

д) отменить предоставленное пользователю APB полномочие на таблицу ukr, т.е. выйти из SQL Query Analyser (disconnect), войти в SQL Query Analyser с именем STUD1 и ввести команду:

revoke select on ukr from APB [F5]

6) Создать представление с именем pr1 на основе таблицы ukr.

Допустим, что пользователь APB не знает имена полей таблицы ukr, но знает, что структура этой таблицы аналогична структуре таблицы ross. Для работы с таблицей ukr ему необходимо создать следующее представление:

```
create view pr1
(nomer1, fio1, data_rogden1, pol1, sem_polog_k1,
spec1, kat_obuch_k1, vuz_k1, mp1, gp1, mo1, gok1)
as select * from ukr [F5]
```

Замечание. Предварительно пользователю APB владельцем БД STUD1 должны быть предоставлены полномочия на использование команды create view и команды select в таблице ukr:

grant create view to APB [F5]

grant select on ukr APB [F5]

7) Предоставить вспомогательную таблицу vuz_gorod и четыре таблицы словаря БД в общее пользование. Для этого необходимо ввести

следующие команды grant для каждой таблицы (команды вводятся владельцем БД STUD1):

grant all on vuz_gorod to public	[F5]
grant all on vuz to public	[F5]
grant all on family to public	[F5]
grant all on kat_obuch to public	[F5]
grant all on gorod to public	[F5]

Задание:

- 1) Из БД master (входное серверное имя **sa**, пароль **compute**) создать 2 входных серверных имени, имена произвольны;
- 2) Используя эти имена, в своей БД (выйти из БД master и войти в свою) создать двух пользователей вашей БД;
- 3) Предоставить первому пользователю командное полномочие (произвольное), а второму – объектное (тоже произвольное);
- 4) Создать представление на основе сложного оператора SELECT (типа оператора SELECT из лабораторной работы №3 или №4).

ЛАБОРАТОРНАЯ РАБОТА № 7

Цель:

1. Знакомство с использованием контрольных ограничений и правил для поддержки целостности данных в БД.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

Самым простым способом поддержки целостности данных является создание правил или контрольных ограничений, причем двух типов:

- на уровне поля;
 - на уровне таблицы.
- (см. конспект курса).

1) Создание контрольных ограничений на уровне поля.

Рассмотрим структуру таблицы poss:

- а) поле номер (номер) - контрольным ограничением для него является промежуток чисел от 1 до 700000;

- б) поле фамилия, имя, отчество (fio) - контрольным ограничением для него является запрет использования символов "." (точка) и "-" (минус);
- в) поле дата рождения (data_rogden) разбивается на три подполя:
 1. день рождения (d_rogd) - контрольным ограничением является промежуток символов от "01" до "31";
 2. месяц рождения (m_rogd) - контрольным ограничением является промежуток символов от "01" до "12";
 3. год рождения (g_rogd) - контрольным ограничением является промежуток символов от "00" до "99";
- г) поле пол (pol) - контрольным ограничением для него является использование либо символа "М", либо символа "Ж";
- д) поле код семейного положения (sem_polog_k) - контрольным ограничением для него является список из четырех значений "0", "1", "2", "3";
- е) поле специальность (spec) - контрольным ограничением для него является промежуток символов от "000000" до "190046";
- ж) поле код категории обучения (kat_obuch_k) - контрольным ограничением является промежуток символов от "01" до "17";
- з) поле код учебного заведения (vuz_k) - контрольным ограничением является промежуток чисел от 128955 до 8199999;
- и) поле месяц приема (mp) - контрольным ограничением является промежуток символов от "01" до "12";
- к) поле год приема (gp) - контрольным ограничением является промежуток символов от "00" до "99";
- л) поле месяц окончания (mo) - контрольным ограничением является промежуток символов от "01" до "12";
- м) поле год окончания (gok) - контрольным ограничением является промежуток символов от "00" до "99".

На основе перечисленных для каждого поля структуры таблицы poss контрольных ограничений можно создать таблицу с именем poss1. Для этого в верхней области окна необходимо ввести:

```
create table poss1
(nomer integer check (nomer between 1 and 700000),
fio char (40) not null check (fio not like '%.%' or fio not like '%-%'),
d_rogd char(2) check (d_rogd between '01' and '31'),
m_rogd char(2) check (m_rogd between '01' and '12'),
g_rogd char(2) check (g_rogd between '00' and '99'),
pol char(1) check (pol like 'M' or pol like 'Ж'),
sem_polog_k char(1) check (sem_polog_k in ('0', '1', '2', '3')),
spec char(6) check (spec between '000000' and '190046'),
kat_obuch_k char(2) check (kat_obuch_k between '01' and '17'),
vuz_k integer check (vuz_k between 128955 and 8199999),
```

```

mp char(2) check (mp between '01' and '12'),
gp char(2) check (gp between '00' and '99'),
mo char(2) check (mo between '01' and '12'),
gok char(2) check (gok between '00' and '99'))    [F5]

```

Проверка действий контрольных ограничений осуществляется оператором insert (или update), т.е. при вводе записей в таблицу.

2) Создание контрольных ограничений на уровне таблицы.

Рассмотрим некоторые значения поля "категория обучения":

- а) значения "п/ф(вуз)" и "п/ф(техн.)" (коды "01" и "02") обозначают обучающегося на подготовительных факультетах (в вузе и техникуме), срок обучения которого не превышает одного года;
- б) значения "студент" и "студент-заочник" (коды "03" и "04") обозначают обучающегося в высшем учебном заведении, срок обучения которого не превышает 5 или 6 лет;
- в) значения "учащийся" и "бакалавр" (коды "05" и "17") обозначают учащегося либо среднего учебного заведения, либо высшего учебного заведения, срок обучения которого не превышает 4 или 5 лет;
- г) значение "аспирант" (код "07") обозначает обучающегося в аспирантуре, срок обучения которого 3 года;
- д) значение "аспирант-заочник" (код "08") обозначает обучающегося в аспирантуре, срок обучения которого 4 года;
- е) значения "повышение квалификации" и "стажер" (код "10" и "11") обозначают обучающегося в высшем учебном заведении, срок обучения которого не превышает 1 или 2 года.

Срок обучения можно подсчитать, если найти разницу между годом окончания учебного заведения и годом приема. Для этого нужно использовать следующее выражение:

```
convert (integer, gok) - convert (integer, gp)
```

Для создания рассмотренных контрольных ограничений воспользуемся возможностью создания контрольных ограничений на уровне таблицы в команде create table таблицы с именем poss2:

```

create table poss2
(nomer integer check (nomer between 1 and 700000),
fio char (40) not null check (fio not like '%.%' or fio not like '%-%'),
d_rogd char(2) check (d_rogd between '01' and '31'),
m_rogd char(2) check (m_rogd between '01' and '12'),
g_rogd char(2) check (g_rogd between '00' and '99'),
pol char(1) check (pol like 'M' or pol like 'Ж'),
sem_polog_k char(1) check (sem_polog_k in ('0', '1', '2', '3')),
spec char(6) check (spec between '000000' and '190046'),
kat_obuch_k char(2) check (kat_obuch_k between '01' and '17'),

```

```

vuz_k integer check (vuz_k between 128955 and 8199999),
mp char(2) check (mp between '01' and '12'),
gp char(2) check (gp between '00' and '99'),
mo char(2) check (mo between '01' and '12'),
gok char(2) check (gok between '00' and '99'),
constraint kat_obuch_const check
  ((kat_obuch_k in ('01', '02') and
    (convert (integer, gok) - convert (integer, gp))=1) or
    (kat_obuch_k in ('03', '04') and
    (convert (integer, gok) - convert (integer, gp)) in (5, 6)) or
    (kat_obuch_k in ('05', '17') and
    (convert (integer, gok) - convert (integer, gp)) in (4, 5)) or
    (kat_obuch_k='07' and (convert (integer, gok) -
    convert (integer, gp))=3) or
    (kat_obuch_k='08' and (convert(integer, gok) -
    convert (integer, gp))=4) or
    (kat_obuch_k in('10', '11') and
    (convert (integer, gok) - convert (integer, gp)) in (1, 2)))) [F5]

```

Проверка действий контрольных ограничений осуществляется оператором insert (или update), т.е. при вводе записей в таблицу.

3) Создание правила

Если контрольное ограничение на уровне поля не было создано в команде create table или созданное контрольное ограничение не совсем удовлетворяет владельца БД, то это ограничение можно создать с помощью команды create rule.

Например, контрольное ограничение для поля kat_obuch_k было задано с помощью функции between '01' and '17' в команде create table poss1, а в текущем сеансе работы владельцу БД понадобилось уточнить конкретные значения этого поля, например задать следующий список: '01', '02', '03', '04', '05', '07', '08', '10', '11', '17'. Это делается с помощью следующей команды:

```
create rule rule1 as @kat_obuch in ('01', '02', '03', '04', '05', '07', '08', '10', '11', '17') [F5]
```

После создания правила rule1 (загрузки его в БД как объекта) необходимо связать его с таблицей БД poss1 с помощью процедуры sp_bindrule (см. конспект курса):

```
sp_bindrule 'rule1', 'poss1.[kat_obuch_k]'
```

Проверка действия правила осуществляется оператором insert (или update), т.е. при вводе записей в таблицу.

Задание:

- 1) Создать контрольные ограничения на уровне поля, описанные выше, для вашей информационной таблицы в вашей БД;
- 2) Создать следующие контрольные ограничения на уровне таблицы:
 - А) значение поля «Год рождения» меньше значения поля «Год приема», которое, в свою очередь, меньше значения поля «Год окончания»;
 - Б) если значение поля «Код категории обучения» равно '07', '08' или '09' (аспирант, аспирант-заочник или ординатура), то первые два символа в значении поля «Специальность» больше '00'; в остальных случаях (по всем другим значениям поля «Код категории обучения») первые два символа значения поля «Специальность» равны '00';
 - В) если значение поля «Пол» равно «Ж», то фамилия в поле «Фамилия Имя Отчество» должна заканчиваться на букву «А», иначе, если значение поля «Пол» равно «М», то фамилия в поле «Фамилия Имя Отчество» не должна заканчиваться на букву «А»;
- 3) Создать правило для вашей информационной таблицы, описанное выше.

ЛАБОРАТОРНАЯ РАБОТА № 8

Цель:

1. Практически освоить создание и использование сложных (с элементами программирования) хранимых процедур.
2. Научиться использовать хранимые процедуры для выполнения двумерных и трехмерных статистических запросов.

Справочный материал:

1. Конспект курса.

Упражнения

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

Вариант двумерного статистического запроса к БД

Выдать статистику по всем учебным заведениям и категориям обучения учащихся России, принятых на учебу в 1990 году.

В этом запросе требуется, во-первых: выдать общее количество учащихся, принятых на учебу в 1990 году; во-вторых: распределить это количество по разным учебным заведениям, т.е. для каждого учебного заведения определить количественные характеристики; в-третьих: каждую

количественную характеристику каждого учебного заведения распределить по разным категориям обучения, т.е. определить количественные характеристики каждой категории обучения в каждом учебном заведении.

Другими словами, если использовать только SELECT - предложения, то необходимо выполнить три запроса (и получить три разных отчета):

- 1) запрос получения общего количества учащихся, принятых в 1990 году;
- 2) запрос с условием выборки по году приема (значение '90'), с предложением group by для группирования значений по полю "учебное заведение" и с агрегатной функцией count(*) в списке выборки для оценки количества значений учебных заведений в каждой группе;
- 3) запрос с условием выборки по году приема (значение '90'), с предложением group by для группирования значений по двум полям: "учебное заведение" и "категория обучения" и с агрегатной функцией count(*) в списке выборки для оценки количества значений категорий обучения в каждой группе.

Замечание. То, что функция count(*) подсчитывает количество значений в группах для последнего указанного в списке выборки поля, является ограничением языка SQL, поэтому для подсчета статистики по двум полям (атрибутам) требуется выполнять два отдельных запроса.

Перечисленные выше запросы выглядят следующим образом:

- 1) `select count(*) from poss where gp='90'`
- 2) `select vuz.uch_zavedenie,
count(*)
from poss, vuz
where gp='90' and vuz_k=vuz.cod
group by vuz.uch_zavedenie`
- 3) `select vuz.uch_zavedenie, kat_obuch.kat_obucheniya,
count(*)
from poss, vuz, kat_obuch
where gp='90' and vuz_k=vuz.cod
and kat_obuch_k=kat_obuch.cod
group by vuz.uch_zavedenie, kat_obuch.kat_obucheniya`

Замечание. Значения полей "учебное заведение" и "категория обучения" при выдаче отчета должны быть раскодированы, поэтому в запросах используется операция соединения таблиц.

То, что результат двумерного статистического запроса расположен в трех разных отчетах, не совсем удовлетворяет пользователя. Для исправления этого нужно использовать программирование в хранимой процедуре.

Для объединения всех количественных характеристик значений атрибутов и общего итога по запросу в один отчет необходимо в теле

хранимой процедуры использовать курсор (см. конспект курса). В качестве предложения select для курсора нужно взять запрос номер 3), в котором определяются количества значений по последнему в списке выборки атрибуту (в данном запросе "категория обучения"). Суммируя количественные характеристики значений категорий обучения можно получить количественные характеристики соответствующих значений учебных заведений ("учебное заведение" - первый атрибут в списке выборки), а также общий итог по запросу (т.е. по приему 1990 года).

Для создания процедуры в верхней области окна необходимо ввести текст процедуры (команда create procedure):

```
create procedure zapr8 as
declare @yz char (25) /* переменная для названия учебного заведения */
declare @ko char (16) /* переменная для названия категории обучения */
declare @it1 int /* переменная для итога по полю категория обучения */
declare @it2 int /* переменная для итога по полю учебное заведение */
declare @itall int /* переменная для итога по всему запросу */
declare @yz1 char (25) /* переменная для названия учебного заведения */
/* объявление курсора для двумерного статистического запроса */
declare y cursor for
select vuz.uch_zavedenie, kat_obucheniya, count (*)
from poss join vuz on vuz_k=vuz.cod
      join kat_obuch on kat_obuch_k=kat_obuch.cod
where gp='90'
group by vuz.uch_zavedenie, kat_obuch.kat_obucheniya
select @it2=0 /* начальное значение итога по полю уч. заведение=0 */
select @itall=0 /* начальное значение итога по запросу = 0 */
open y /* открытие курсора */
fetch y into @yz, @ko, @it1
/* считывание первого итога по полю категория обучения */
if (@@fetch_status=-2) begin
print 'Ошибка при выполнении первого FETCH'
close y /* закрытие курсора и останов процедуры в случае ошибки */
return
end
if (@@fetch_status=-1) begin
print 'Данные не найдены'
close y /* закрытие курсора и останов процедуры в случае отсутствия
данных по запросу */
return
end
select @yz1=@yz /* запоминание названия учебного заведения в @yz1 */
print @ko+' '+str(@it1) /* печать названия категории обучения и итога */
```

```

select @it2=@it2+@it1 /* подсчет итога по полю учебное заведение */
select @itall=@itall+@it1 /* подсчет общего итога по запросу */
/* цикл обработки запроса */
while (@@fetch_status=0)
begin
    fetch y into @yz, @ko, @it1 /* считывание очередного итога по
полю категория обучения */
    if (@yz1 !=@yz) begin /* если название учебного заведения
поменялось, то печать */
        print 'учебное заведение ' + @yz1 + ' - ' +str(@it2) /* старого
названия учебного заведения и итога */
        select @yz1=@yz/*присвоение нового названия учебного заведения */
        select @it2=0 /* новый итог = 0 */
        end
        if (@@fetch_status=-1) break /* при окончании данных немедленный
выход из цикла */
        print @ko + ' - ' +str(@it1) /* печать очередных названия категории
обучения и итога */
        select @it2=@it2+@it1 /* подсчет итога по полю учебное заведение */
        select @itall=@itall+@it1 /* подсчет общего итога по запросу */
        end
        close y /* закрытие курсора по окончании цикла */
        if (@@fetch_status=-2) begin
            print 'Ошибка при выполнении FETCH'
            return /* останов процедуры в случае ошибки */
        end
        print 'учебное заведение ' + @yz + ' - ' +str(@it2) /* печать последнего
итога по учебному заведению */
        print '-----'
        print 'всего по запросу-' +str(@itall) /*печать общего итога по запросу */
        deallocate y /* освобождение курсора */
        return [F5]

```

Для выполнения запроса нужно запустить эту процедуру, т.е. в верхней области окна необходимо ввести:

zapr8 [F5]

Отчет будет представлен в нижней области окна (см. рис. 29).

учебное заведение	специальность	количество
ВСЕС.Н/Ц.ХИРУРГИИ	-	1
АСПИРАНТ	1	-
учебное заведение ИНСТ. ВИРУСОЛОГИИ АМНСССР	-	1
АСПИРАНТ	1	-
учебное заведение ИНСТ. НЕЙРОХИРУРГИИ	-	1
АСПИРАНТ	1	-
учебное заведение ИНСТ. ПЕДИАТРИИ	-	1
АСПИРАНТ	2	-
учебное заведение ИНСТ. СЕРД-СОСУД.ХИРУРГИИ	-	2
АСПИРАНТ	1	-
учебное заведение МОСК.ВЕТ.АКАД.	-	1
АСПИРАНТ	2	-
учебное заведение МОСК.ГИДРОМЕЛ.ИНСТ.	-	2
АСПИРАНТ	4	-
учебное заведение МОСК.ИНСТ.С/Х ПР.	-	4
АСПИРАНТ	3	-
учебное заведение С.-П. С/Х ИНСТ.	-	3
АСПИРАНТ	1	-
учебное заведение С.-П. ФАРМ.ИНСТ.	-	1
АСПИРАНТ	1	-
учебное заведение ЦЕНТР. НИИ ЭПИДЕМИОЛОГИИ	-	1
АСПИРАНТ	2	-
учебное заведение ЦЕНТР.ИНСТ.УСОВЕРШ.ВРАЧЕЙ	-	2
АСПИРАНТ-ЗАОЧНИК	1	-
учебное заведение ВСЕС Н/П МАТЕРИ И РЕБЕНКА	-	1

Рис. 29

Вариант трехмерного статистического запроса к БД

Выдать статистику по всем учебным заведениям, специальностям и категориям обучения учащихся России

Трехмерный статистический запрос выполняется аналогично двумерному, за исключением того, что в этом запросе и в списке выборки, и в предложении group by добавляется еще одно поле (атрибут "специальность"). Другими словами, в этом запросе требуется, во-первых: выдать общее количество учащихся в России; во-вторых: распределить это количество по разным учебным заведениям, т.е. определить количественные характеристики для каждого учебного заведения; в-третьих: каждую количественную характеристику каждого учебного заведения распределить по разным шифрам специальностей, т.е. определить количественные характеристики каждой специальности в каждом учебном заведении; в-

четвертых: каждую количественную характеристику каждого шифра специальности (в каждом учебном заведении) распределить по разным категориям обучения, т.е. определить количественные характеристики каждой категории обучения для каждого шифра специальности (в каждом учебном заведении).

Если использовать только SELECT - предложения, то потребуется выполнить четыре отдельных запроса и получить четыре разных отчета (по аналогии с двумерным статистическим запросом), что совершенно не удовлетворит пользователя. Поэтому для реализации этого запроса необходимо использовать курсор и программирование в хранимой процедуре.

В качестве предложения select для курсора берется следующее предложение:

```
select vuz.uch_zavedenie, spec, kat_obuch.kat_obucheniya,
count(*)
from poss, vuz, kat_obuch
where vuz_k=vuz.cod and kat_obuch_k=kat_obuch.cod
group by vuz. uch_zavedenie, spec, kat_obuch. kat_obucheniya
```

Данный курсор позволяет получать количественные характеристики разных категорий обучения (последний атрибут в списке выборки и в предложении group by). Если суммировать эти количественные характеристики, то можно получить и количественные характеристики соответствующих шифров специальностей, и количественные характеристики соответствующих учебных заведений, и общий итог по запросу.

Для создания процедуры в верхней области окна необходимо ввести текст процедуры (команда create procedure):

```
create procedure zapr9 as
declare @yz char (25) /* переменная для названия учебного заведения */
declare @sp char (6) /* переменная для шифра специальности */
declare @ko char (16) /* переменная для названия категории обучения */
declare @it1 int /* переменная для итога по полю категория обучения */
declare @it2 int /* переменная для итога по полю специальность */
declare @it3 int /* переменная для итога по полю учебное заведение */
declare @itall int /* переменная для итога по всему запросу */
declare @yz1 char (25) /* переменная для названия учебного заведения */
declare @sp1 char (6) /* переменная для шифра специальности */
/* объявление курсора для трехмерного статистического запроса */
declare y cursor for
select vuz.uch_zavedenie, spec,
kat_obuch.kat_obucheniya, count (*)
from poss join vuz on vuz_k=vuz.cod
```

```

join kat_obuch on kat_obuch_k=kat_obuch.cod
group by vuz.uch_zavedenie, spec, kat_obuch. kat_obucheniya
select @it2=0 /*начальное значение итога по полю специальность = 0 */
select @it3=0 /*начальное значение итога по полю учебное
заведение=0*/
select @itall=0 /*начальное значение итога по запросу = 0 */
open y /* открытие курсора */
fetch y into @yz, @sp, @ko, @it1 /* считывание первого итога по
полю категория обучения */
if (@@fetch_status=-2) begin
print 'Ошибка при выполнении первого FETCH'
close y /* закрытие курсора и останов процедуры в случае ошибки */
return
end
if (@@fetch_status=-1) begin
print 'Данные не найдены'
close y /* закрытие курсора и останов процедуры в случае отсутствия
данных */
return /* по запросу */
end
select @yz1=@yz/*запоминание названия учебного заведения в @yz1 */
select @sp1=@sp /* запоминание шифра специальности в @sp1 */
print @ko + ' - ' + str(@it1) /* печать названия категории обучения и
итога */
select @it2=@it2+@it1 /* подсчет итога по полю специальность */
select @it3=@it3+@it1 /* подсчет итога по полю учебное заведение */
select @itall=@itall+@it1 /* подсчет общего итога по запросу */
/* цикл обработки запроса */
while (@@fetch_status=0) begin
fetch y into @yz, @sp, @ko, @it1 /* считывание очередного итога по
полю категория обучения */
if (@yz1!=@yz) begin /* если название учебного заведения поменялось,
то печать */
print 'учебное заведение ' + @yz1 + ' - ' + str(@it3) /* старого названия
учебного заведения и итога */
select @yz1=@yz /* присвоение нового названия учебного заведения */
select @it3=0 /* новый итог = 0 */
end
if (@sp1!=@sp) begin /*если шифр специальности поменялся, то
печать*/
print 'специальность ' + @sp1 + ' - ' + str(@it2) /* старого шифра
специальности и итога */

```

```

select @sp1=@sp /* присвоение нового шифра специальности */
select @it2=0 /* новый итог = 0 */
end
if (@@fetch_status=-1) break /* при окончании данных немедленный
выход из цикла */
print @ko + ' - ' +str(@it1) /* печать очередных названия категории
обучения и итога */
select @it2=@it2+@it1 /* подсчет итога по полю специальность */
select @it3=@it3+@it1 /* подсчет итога по полю учебное заведение */
select @itall=@itall+@it1 /* подсчет общего итога по запросу */
end
close y /* закрытие курсора по окончании цикла */
if (@@fetch_status=-2) begin
print 'Ошибка при выполнении FETCH'
return /* останов процедуры в случае ошибки */
end
print 'специальность ' + @sp + ' - ' + str(@it2) /* печать последнего
итога по специальности */
print 'учебное заведение ' + @yz + ' - ' + str(@it3) /* печать последнего
итога по учебн. заведению */
print 'всего по запросу -'+str(@itall)/*печать общего итога по запросу */
deallocate y /* освобождение курсора */
return [F5]

```

Для выполнения запроса нужно запустить эту процедуру, т.е. в верхней области окна необходимо ввести:

zapr9 [F5]

Отчет будет представлен в нижней области окна (см. рис. 30).

Варианты заданий на выполнение лабораторной работы № 8 приведены в приложении V.

ЛАБОРАТОРНАЯ РАБОТА № 9

Цель:

1. Практически освоить создание и использование элементарных триггеров.
2. Научиться использовать predetermined таблицы в теле триггеров.

Справочный материал:

1. Конспект курса.

Упражнения:

Перед выполнением лабораторной работы необходимо войти в сеть, затем в систему MS SQL Server и SQL Query Analyser, а также установить

связь пользователя с базой данных (аналогично описанному сценарию в лабораторной работе № 1).

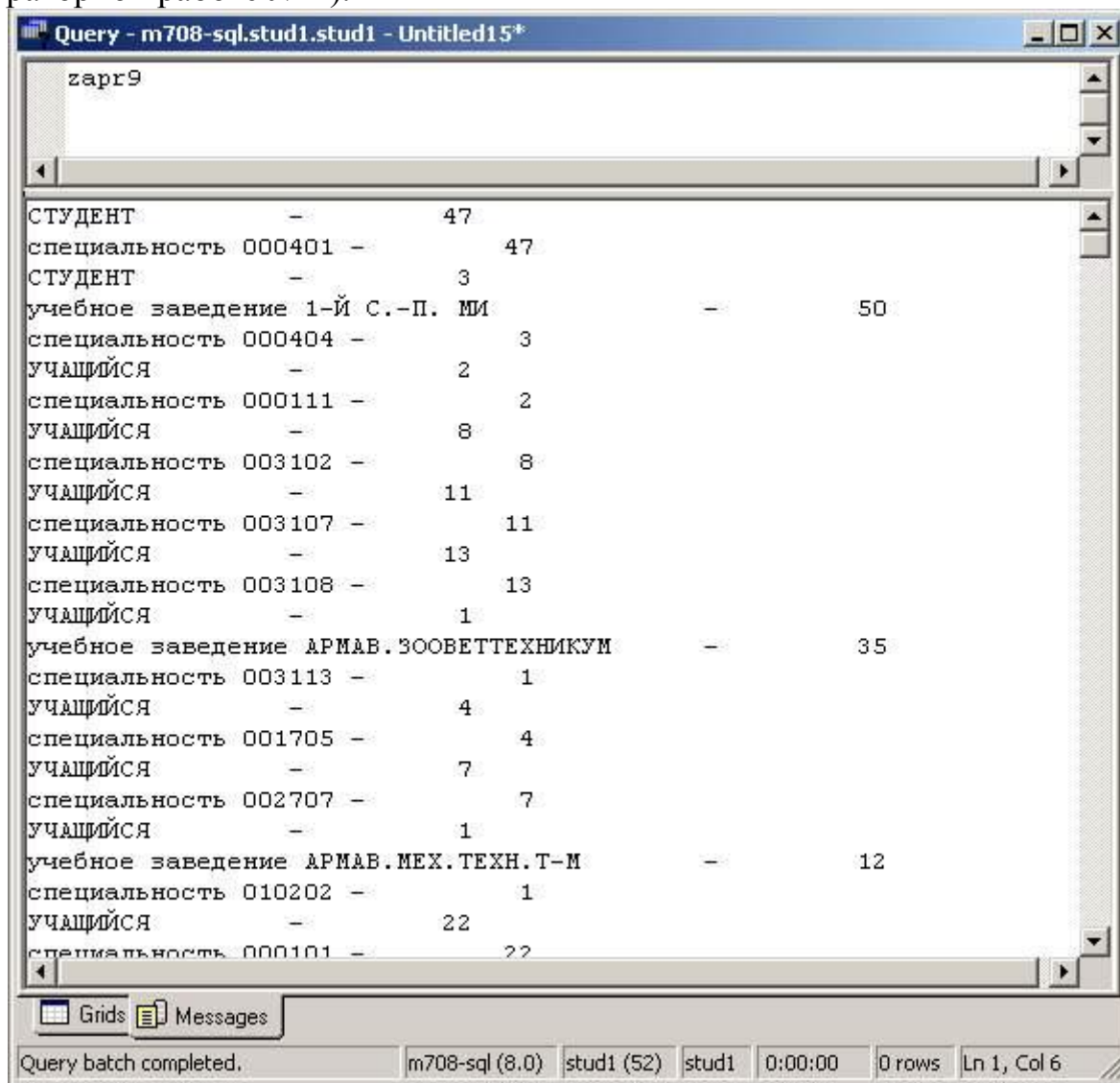


Рис. 30

Упражнение 1

Разработать триггер, который срабатывает при выполнении любого из трех операторов: update, delete, insert в таблице vuz, с целью выдачи сообщения о необходимости выполнения такого же оператора в таблице vuz_gorod.

Для создания триггера в верхней области окна необходимо ввести:

```
create trigger vuztrig1 on vuz
```

```
for insert, update, delete as
```

```
select 'Необходимо выполнить такую же коррекцию в таблице  
vuz_gorod' [F5]
```

Если при наборе этой команды будет допущена синтаксическая ошибка, то в нижней области окна будет выдано сообщение об ошибке.

Для того, чтобы убедиться, что триггер создан, можно воспользоваться командой `sp_help`. Чтобы проверить срабатывание этого триггера, необходимо ввести один из трех операторов (`update`, `delete`, `insert`) для таблицы `vuz`, например:

в верхней области окна необходимо ввести:

```
insert vuz values
(7962902, 'СТАВР. МЕД. ИНСТ.')      [F5]
delete vuz where cod = 7962902      [F5]
```

В нижней области окна будет выдано сообщение из триггера `vuztrig1` (см. рис. 31).

В теле триггеров можно использовать predefined таблицы с именами `inserted` и `deleted` (см. конспект курса).

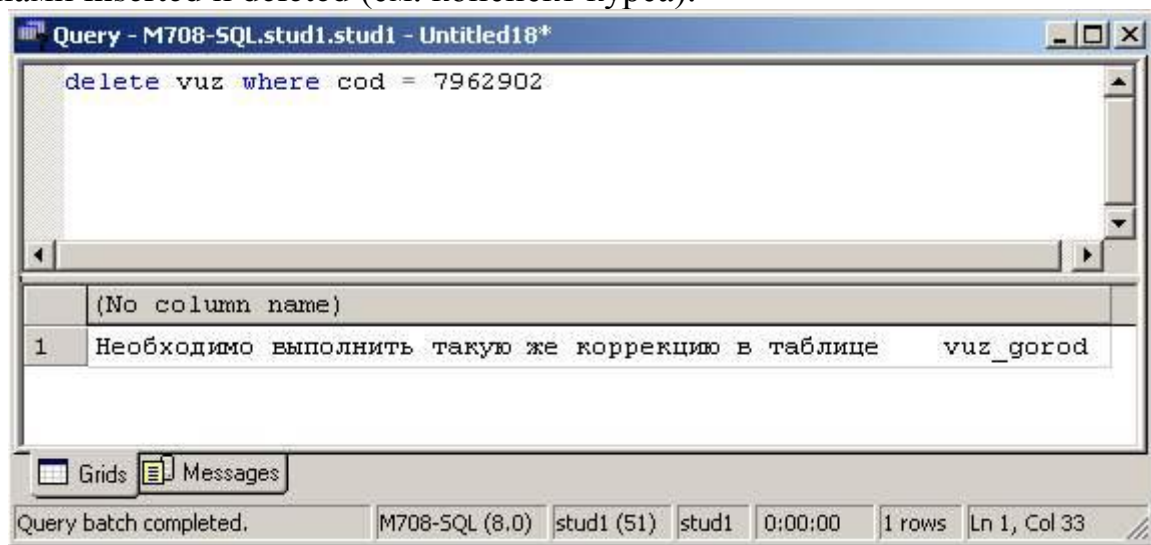


Рис. 31

Упражнение 2

Разработать триггер, который срабатывает при добавлении новых записей в таблицу `vuz_gorod`, с целью выдачи сообщения о необходимости добавления новой записи в таблицу `vuz`.

Для создания триггера в верхней области окна необходимо ввести:

```
create trigger vuztrig2 on vuz_gorod
for insert as
if (select count(*) from inserted)>0
select 'Необходимо добавить запись в таблицу vuz' [F5]
```

Чтобы проверить срабатывание этого триггера, необходимо ввести оператор `insert` для таблицы `vuz_gorod`, например:

в верхней области окна необходимо ввести:

```
insert vuz_gorod values
(7115842, 'С.-П. ИИЖТ', '002', 'САНКТ-ПЕТЕРБУРГ', '01', 'РОССИЯ')
[F5]
```

В нижней области окна будет выдано сообщение из триггера vuztrig2 (см. рис. 32).

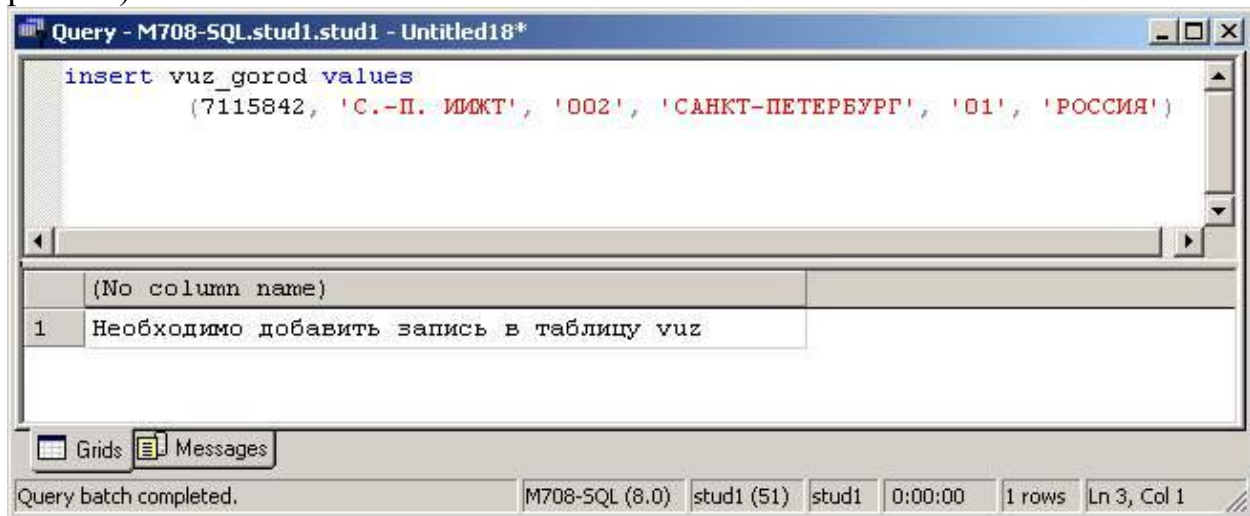


Рис. 32

Упражнение 3

Разработать триггер, который срабатывает при удалении записей из таблицы vuz_gorod, с целью выдачи сообщения о необходимости удаления старой записи из таблицы vuz.

Для создания триггера в верхней области окна необходимо ввести:

`create trigger vuztrig3 on vuz_gorod`

`for delete as`

`if (select count(*) from deleted)>0`

`select 'Необходимо удалить старую запись из таблицы vuz' [F5]`

Чтобы проверить срабатывание этого триггера, необходимо ввести оператор delete для таблицы vuz_gorod, например:

в верхней области окна необходимо ввести:

`delete vuz_gorod`

`where cod_vuza=7115842 [F5]`

В нижней области окна будет выдано сообщение из триггера vuztrig3 (см. рис. 33).

Упражнение 4

Разработать триггер, который срабатывает при вводе в таблицу poss1 неправильных значений в поля vuz_k (учебное заведение - код) и spes (шифр специальности), с целью выдачи сообщений об ошибках в соответствующих полях.

Таблица poss1 имеет такую же структуру, что и таблица poss (за исключением поля "дата рождения"), и предназначена для накопления новых записей с последующим их копированием в таблицу poss (в таблицу poss необходимо вводить безошибочные записи).

Для создания этого триггера необходимо создать две таблицы, каждая из которых состоит из одного поля: 1) поле, содержащее все коды учебных заведений; 2) поле, содержащее все шифры специальностей. Другими

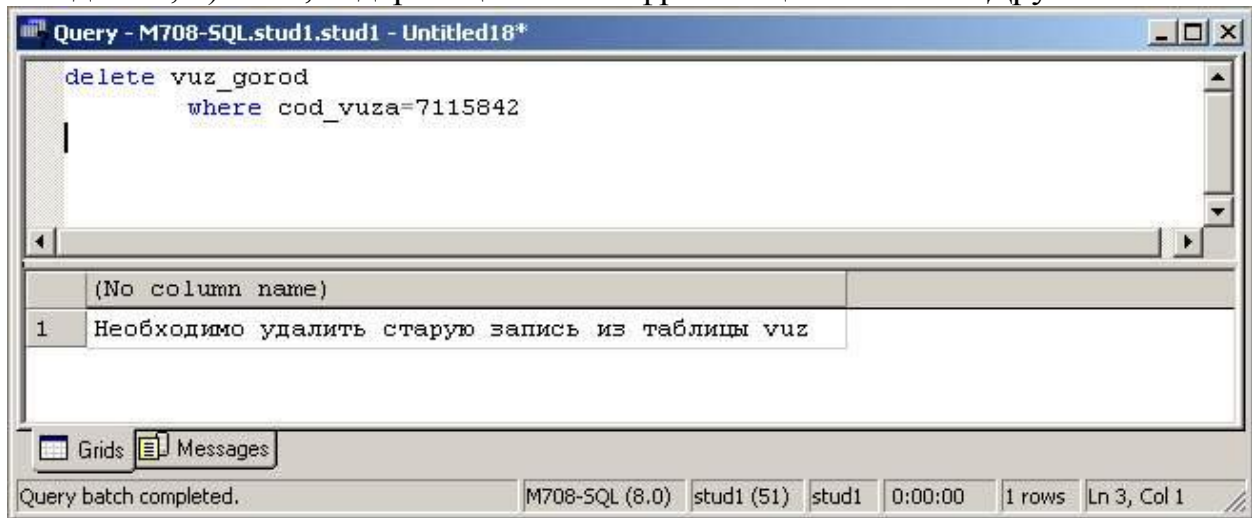


Рис. 33

словами, должны быть созданы домены (области допустимых значений) для этих двух полей.

Для создания таблицы только кодов учебных заведений необходимо в верхней области окна ввести:

```
create table codv (vkod integer) [F5]
```

а для создания таблицы только шифров специальностей в верхней области окна необходимо ввести:

```
create table cods (skod char(6)) [F5]
```

Для загрузки этих таблиц данными необходимо использовать файлы, содержащие пакеты из команд insert типа:

```
insert codv values (<код учебного заведения>)...
```

```
insert cods values ('<шифр специальности>')...
```

После этого для создания триггера в верхней области окна необходимо ввести:

```
create trigger vuz_spec on poss1
for insert as
if (select count(*) from codv, inserted
where codv.vkod=inserted.vuz_k)=0
select 'значение поля vuz_k не принадлежит домену'
if (select count(*) from cods, inserted
where cods.skod=inserted.spec)=0
select 'значение поля spec не принадлежит домену' [F5]
```

Триггер vuz_spec сравнивает введенное значение поля vuz_k или spec со всеми значениями поля в таблице codv (всеми кодами учебных заведений)

или в таблице cods (всеми шифрами специальностей) и в случаях несовпадения выдает соответствующие сообщения.

Чтобы проверить срабатывание этого триггера, необходимо ввести оператор insert для таблицы poss1, содержащий либо неправильный код учебного заведения, либо неправильный шифр специальности, либо то и другое, например (ошибочные коды подчеркнуты):

в верхней области окна необходимо ввести:

```
insert poss1 values
```

```
insert poss1 values
```

```
(699991, 'ЖУРАВЛЕВ АНДРЕЙ АНДРЕЕВИЧ', '01','02','75', 'М', '0',  
'001991', '05', 0129999, '08', '93', '03', '98') [F5]
```

В нижней области окна будет выдано сообщение из триггера vuz_spec (см.рис. 34).

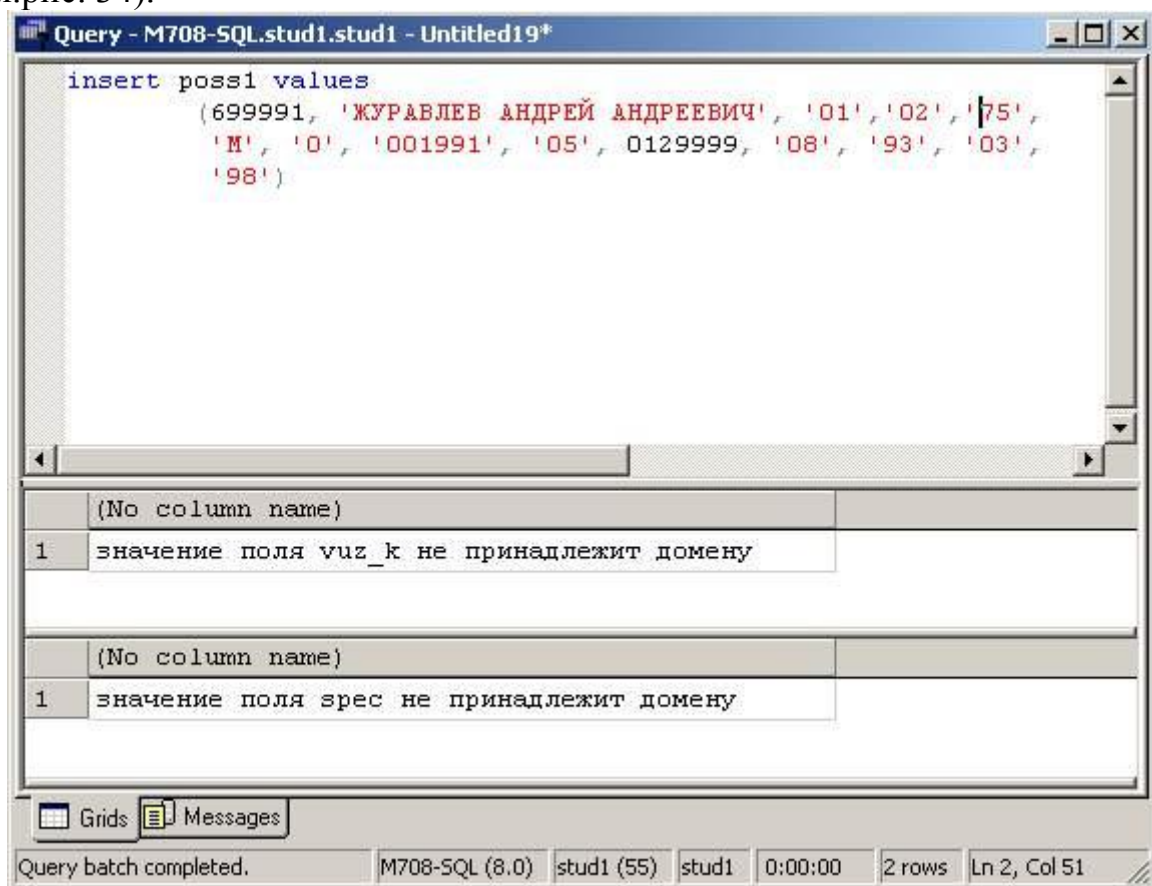


Рис. 34

Задание:

- 1) Создать триггер на проверку правильности ввода значений в одно из полей вашей информационной таблицы: «Год приема» (значения от '00' до '99'), «Год окончания» (значения от '00' до '99'), «Код категории обучения» (значения от '01' до '17'), «Код семейного положения» (значения от '0' до '3'); значения этих полей должны быть заданы в отдельной таблице вашей БД;

- 2) Создать 3 отдельных триггера (для операторов INSERT, UPDATE и DELETE) для автоматического внесения изменений в таблицу VUZ (поля «Код вуза» и «Название вуза»), если вносятся такие же изменения в таблицу VUZ_GOROD.

ПРИЛОЖЕНИЕ I

ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №2

Задание 1

Выдать список студентов и аспирантов вашего государства только 1966 года рождения. Отсортировать данные по учебным заведениям и категориям обучения.

Задание 2

Выдать список выпускников 93 и 94 г.г. вашего государства по специальностям от 0401 до 0409. Отсортировать данные по специальностям и годам окончания.

Задание 3

Выдать список учащихся вашего государства, принятых на учебу в 89 и 91 г.г. только 67 года рождения. Отсортировать данные по годам приема и учебным заведениям.

Задание 4

Выдать список учащихся вашего государства по специальностям от 0601 до 0620 и только мужского пола. Отсортировать данные по специальностям и учебным заведениям.

Задание 5

Выдать список учащихся вашего государства только женского пола, незамужних и по специальности 0401. Отсортировать данные по учебным заведениям.

Задание 6

Выдать список выпускников 93 и 95 г.г. вашего государства и только 1964 года рождения. Отсортировать данные по годам окончания и учебным заведениям.

Задание 7

Выдать список студентов и стажеров вашего государства и только мужского пола. Отсортировать данные по категориям обучения и учебным заведениям.

Задание 8

Выдать список студентов вашего государства, принятых на учебу в 90 и 91 г.г. Отсортировать данные по учебным заведениям и годам приема.

Задание 9

Выдать список выпускников 94 года вашего государства по специальностям 0101, 0201 и 0401. Отсортировать данные по специальностям и учебным заведениям.

Задание 10

Выдать список учащихся вашего государства только мужского пола

и с фамилиями, начинающимися на букву К. Отсортировать данные по учебным заведениям.

Задание 11

Выдать список учащихся 1965 года рождения вашего государства по специальностям от 0201 до 0230. Данные отсортировать по специальностям и учебным заведениям.

Задание 12

Выдать список учащихся вашего государства только женского пола и с именами, не заканчивающимися на букву Я. Данные отсортировать по специальностям.

Задание 13

Выдать список учащихся вашего государства, принятых на учебу в 1990 году и по всем специальностям, кроме 0101 и 0301. Данные отсортировать по учебным заведениям и специальностям.

Задание 14

Выдать список выпускников 95 и 96 г.г. вашего государства по всем годам рождения, кроме 1963 г. Данные отсортировать по годам окончания и учебным заведениям.

Задание 15

Выдать список студентов и аспирантов вашего государства и с фамилиями, начинающимися на букву М. Данные отсортировать по категориям обучения и учебным заведениям.

ПРИЛОЖЕНИЕ II

ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №3

Задание 1

Выдать список студентов и аспирантов вашего государства и с фамилиями, начинающимися на букву М. Отчет должен включать все поля, кроме полей НОМЕР и СЕМ_ПОЛОЖЕНИЕ. Поля КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по категориям обучения и учебным заведениям.

Задание 2

Выдать список выпускников 95 и 96 г.г. вашего государства по всем годам рождения, кроме 1963. Отчет должен включать все поля, кроме полей НОМЕР и СЕМ_ПОЛОЖЕНИЕ. Поля КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по годам окончания и учебным заведениям.

Задание 3

Выдать список учащихся вашего государства, принятых на учебу в 1990 году и по всем специальностям, кроме 0101 и 0301. Отчет должен включать все поля, кроме МЕСЯЦ_ПРИЕМА, ГОД_ПРИЕМА, НОМЕР. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по учебным заведениям и специальностям.

Задание 4

Выдать список учащихся вашего государства только женского пола и с именами, не заканчивающимися на букву А. Отчет должен включать все поля, кроме полей НОМЕР и ПОЛ. Поля СЕМ_ПОЛОЖЕНИЕ_К, ВУЗ_К и КАТ_ОБУЧЕНИЯ_К должны быть раскодированы. Данные отсортировать по специальностям.

Задание 5

Выдать список учащихся 1965 года рождения вашего государства по специальностям от 0201 до 0230. Отчет должен включать все поля, кроме полей НОМЕР и ДАТА_РОЖДЕНИЯ. Поля СЕМ_ПОЛОЖЕНИЕ_К, ВУЗ_К и КАТ_ОБУЧЕНИЯ_К должны быть раскодированы. Данные отсортировать по специальностям и учебным заведениям.

Задание 6

Выдать список учащихся вашего государства только мужского пола и с фамилиями, начинающимися на букву К. Отчет должен включать все поля, кроме полей НОМЕР и ПОЛ. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по учебным заведениям.

Задание 7

Выдать список выпускников 94 года вашего государства по специальностям 0101, 0201 и 0401. Отчет должен включать все поля, кроме полей НОМЕР, МЕСЯЦ_ОКОНЧАНИЯ и ГОД_ОКОНЧАНИЯ. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по специальностям и учебным заведениям.

Задание 8

Выдать список студентов и аспирантов вашего государства только 66 года рождения. Отчет должен включать все поля, кроме полей НОМЕР и ДАТА_РОЖДЕНИЯ. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по учебным заведениям и категориям обучения.

Задание 9

Выдать список выпускников 93 и 94 г.г. вашего государства по специальностям от 0401 до 0409. Отчет должен включать все поля, кроме полей НОМЕР и СЕМ_ПОЛОЖЕНИЕ_К. Поля КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по специальностям и годам окончания.

Задание 10

Выдать список учащихся вашего государства, принятых на учебу в 89 и 91 г.г. только 67 года рождения. Отчет должен включать все поля, кроме полей НОМЕР и ДАТА_РОЖДЕНИЯ. Поля СЕМ_ПОЛОЖЕНИЕ_К, ВУЗ_К и КАТ_ОБУЧЕНИЯ_К должны быть раскодированы. Данные отсортировать по годам приема и учебным заведениям.

Задание 11

Выдать список учащихся вашего государства по специальностям от 0601 до 0620 и только мужского пола. Отчет должен включать все поля, кроме полей НОМЕР и ПОЛ. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по специальностям и учебным заведениям.

Задание 12

Выдать список учащихся вашего государства только женского пола, незамужних и по специальности 0401. Отчет должен содержать все поля, кроме полей НОМЕР, ПОЛ, СЕМ_ПОЛОЖЕНИЕ_К и СПЕЦИАЛЬНОСТЬ. Поля КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по учебным заведениям.

Задание 13

Выдать список выпускников 93 и 95 г.г. вашего государства и только 1964 года рождения. Отчет должен включать все поля, кроме полей НОМЕР и ДАТА_РОЖДЕНИЯ. Поля СЕМ_ПОЛОЖЕНИЕ_К, КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные

отсортировать по годам окончания и учебным заведениям.

Задание 14

Выдать список студентов и стажеров вашего государства и только мужского пола. Отчет должен включать все поля, кроме полей НОМЕР, ПОЛ и СЕМ_ПОЛОЖЕНИЕ_К. Поля КАТ_ОБУЧЕНИЯ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по категориям обучения и учебным заведениям.

Задание 15

Выдать список студентов вашего государства, принятых на учебу в 90 и 91 г.г. Отчет должен включать все поля, кроме полей НОМЕР и КАТ_ОБУЧЕНИЯ_К. Поля СЕМ_ПОЛОЖЕНИЕ_К и ВУЗ_К должны быть раскодированы. Данные отсортировать по учебным заведениям и годам приема.

ПРИЛОЖЕНИЕ III

ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №4

Задание 1

Выдать список учащихся САНКТ-ПЕТЕРБУРГА (код 002) только медицинских специальностей (шифры от 000401 до 000420 и от 140001 до 150050). В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и СЕМ_ПОЛОЖЕНИЕ_К. Данные отсортировать по учебным заведениям, категориям обучения и специальностям.

Задание 2

Выдать список студентов и аспирантов вашего государства только 1965 года рождения и только холостых. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР, ДАТА_РОЖДЕНИЯ и СЕМ_ПОЛОЖЕНИЕ_К. Данные отсортировать по категориям обучения и учебным заведениям.

Задание 3

Выдать список учащихся всех городов вашего государства, кроме столицы, и только мужского пола. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и ПОЛ. Данные отсортировать по учебным заведениям, годам окончания и специальностям.

Задание 4

Выдать список выпускников 93 и 94 г.г. вашего государства и только студентов. В отчете должен быть заголовок и все поля, кроме полей: НОМЕР и КАТ_ОБУЧЕНИЯ_К. Данные отсортировать по годам окончания, специальностям и учебным заведениям.

Задание 5

Выдать список учащихся вашего государства, принятых на учебу в 87, 89 и 91 г.г., имеющих год рождения 1964 и по специальностям 0101, 0201, 0301, 0401 и 0501. В отчете должен быть соответствующий заголовок и все поля, кроме НОМЕР и ДАТА_РОЖДЕНИЯ. Данные отсортировать по годам приема, специальностям и учебным заведениям.

Задание 6

Выдать список учащихся и стажеров вашего государства по первым десяти группам специальностей (от 0101 до 1099). В отчете должен быть соответствующий заголовок и все поля, кроме полей МЕСЯЦ_ПРИЕМА, МЕСЯЦ_ОКОНЧАНИЯ и НОМЕР. Данные отсортировать по категориям обучения, специальностям и учебным заведениям.

Задание 7

Выдать список выпускников 92 и 95 г. г. вашего государства только экономических специальностей (шифры от 0601 до 0720 и от 080001 до

080099) и только мужчин. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и ПОЛ. Данные отсортировать по годам окончания, учебным заведениям и специальностям.

Задание 8

Выдать список учащихся столицы вашего государства и только женского пола. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР, ПОЛ и СЕМ_ПОЛОЖЕНИЕ_К. Данные отсортировать по учебным заведениям и специальностям.

Задание 9

Выдать список выпускников 94 и 95 г.г. вашего государства только студенческих специальностей (категории обучения: студент и студент-заочник, шифры специальностей от 0101 до 3199). В отчете должен быть соответствующий заголовок и все поля, кроме полей: НОМЕР и КАТ_ОБУЧЕНИЯ_К. Данные отсортировать по годам окончания, специальностям и учебным заведениям.

Задание 10

Выдать список учащихся столицы вашего государства только экономических специальностей (шифры от 0601 до 0720 и от 080001 до 080099). В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и СЕМ_ПОЛОЖЕНИЕ_К. Данные отсортировать по учебным заведениям, категориям обучения и специальностям.

Задание 11

Выдать список учащихся вашего государства, принятых на учебу 88 и 90 г.г., имеющих год рождения 1966 и по специальностям 0601, 0701, 1001, 1501 и 1801. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и ДАТА_РОЖДЕНИЯ. Данные отсортировать по годам приема, специальностям и учебным заведениям.

Задание 12

Выдать список учащихся всех городов вашего государства, кроме столицы, и только мужского пола. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и ПОЛ. Данные отсортировать по учебным заведениям, годам окончания и специальностям.

Задание 13

Выдать список студентов и учащихся вашего государства по вторым десяти группам специальностей (категории обучения от '03' до '05' и шифры специальностей от 1101 до 2099). В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР, МЕСЯЦ_ПРИЕМА и МЕСЯЦ_ОКОНЧАНИЯ. Данные отсортировать по категориям обучения, специальностям и учебным заведениям.

Задание 14

Выдать список выпускников 92 и 95 г.г. вашего государства только медицинских специальностей (шифры от 0401 до 0420 и от 140001

до 150099) и только мужчин. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР и ПОЛ. Данные отсортировать по годам окончания, учебным заведениям и специальностям.

Задание 15

Выдать список студентов и аспирантов вашего государства только 1965 года рождения и только холостых. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР, ДАТА_РОЖДЕНИЯ и СЕМ_ПОЛОЖЕНИЕ_К. Данные отсортировать по категориям обучения и учебным заведениям.

Задание 16

Выдать список выпускников 93 и 94 г.г. вашего государства только мужского пола и только 1963 года рождения. В отчете должен быть соответствующий заголовок и все поля, кроме полей НОМЕР, ПОЛ и ДАТА_РОЖДЕНИЯ. Данные отсортировать по годам окончания, учебным заведениям и специальностям.

ПРИЛОЖЕНИЕ IV

ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНЫХ РАБОТ №5, №7

Задание 1

Выдать статистику по учебным заведениям вашего государства только прием 1990 года. Отчет должен начинаться заголовком и цифрой - общим количеством записей по запросу (по приему 90-го года). Далее таблица, включающая перечень учебных заведений с цифрами, не равными нулю.

Задание 2

Выдать статистику по городам вашего государства. Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу. Далее таблица, включающая перечень городов с цифрами, не равными нулю.

Задание 3

Выдать статистику по вашему государству по следующим специальностям: 0202, 0203, 0205, 0206 (общее число по каждой специальности), а также в каких учебных заведениях и сколько обучается по этим специальностям. Отчет должен начинаться с соответствующего заголовка. Далее четыре таблицы, где каждая таблица начинается с заголовка: СПЕЦИАЛЬНОСТЬ <шифр> - ВСЕГО <цифра> (не равная нулю), а заканчивается перечнем учебных заведений с цифрами, не равными нулю.

Задание 4

Выдать статистику по учебным заведениям вашего государства и только по лицам мужского пола. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу (всего мужчин). Далее таблица, включающая перечень учебных заведений с цифрами, не равными нулю.

Задание 5

Выдать статистику по всем специальностям вашего государства и только по студентам, заканчивающим учебу в 1994 году. Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу (всего студентов-выпускников 1994 г.). Далее таблица, включающая перечень шифров специальностей с цифрами, не равными нулю.

Задание 6

Выдать статистику по годам приема учащихся вашего государства. Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу. Далее таблица, включающая перечень годов приема с цифрами, не равными нулю.

Задание 7

Выдать статистику по годам окончания учебы учащихся вашего государства. Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу. Далее таблица, включающая перечень годов окончания с цифрами, не равными нулю.

Задание 8

Выдать статистику по категориям обучения учащихся вашего государства только прием 1991 года. Отчет должен начинаться заголовком и цифрой - общим количеством записей по запросу (по приему 91-го года). Далее таблица, включающая перечень категорий обучения с цифрами, не равными нулю.

Задание 9

Выдать статистику по вашему государству по следующим специальностям: 0401, 0402, 0403 и 0405 (общее число по каждой специальности), а также количества мужчин и женщин по каждой из этих специальностей. Отчет должен начинаться с соответствующего заголовка. Далее таблица из четырех строчек, где каждая строчка включает шифр специальности и 3 цифры: всего, количество мужчин и количество женщин.

Задание 10

Выдать статистику по семейному положению учащихся вашего государства, заканчивающих учебу в 1994 или в 1995 году. Отчет должен начинаться с соответствующего заголовка. Далее - две таблицы, где каждая таблица начинается с заголовка: ГОД ВЫПУСКА <год> - ВСЕГО <цифра> (не равная нулю), а заканчивается перечнем семейных положений с цифрами, не равными нулю.

Задание 11

Выдать статистику по городам вашего государства и только по студентам (исключая другие категории обучения). Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу (всего студентов). Далее таблица, включающая перечень городов с цифрами, не равными нулю.

Задание 12

Выдать статистику по учебным заведениям вашего государства, включая цифры по выпуску 1994 года. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее таблица, включающая перечень названий учебных заведений с двумя цифрами (у каждого учебного заведения): 1) всего по учебному заведению (не равная нулю); 2) выпуск 1994 года.

Задание 13

Выдать статистику по категориям обучения учащихся вашего государства, исключая студентов-заочников и аспирантов-заочников. Отчет должен начинаться с соответствующего заголовка и цифры - общего

количества записей по запросу. Далее таблица, включающая перечень требуемых категорий обучения с цифрами, не равными нулю.

Задание 14

Выдать статистику по специальностям 4-ой, 5-ой и 6-ой групп (т. е. специальности от 0401 до 0699) учащихся вашего государства. Отчет должен начинаться с соответствующего заголовка и цифры - общего количества записей по запросу. Далее таблица, включающая перечень требуемых шифров специальностей с цифрами, не равными нулю.

Задание 15

Выдать статистику по учебным заведениям столицы вашего государства. Отчет должен начинаться с заголовка, включающего название города и цифру - общее количество записей по запросу (всего по столице). Далее таблица, включающая перечень требуемых учебных заведений с цифрами, не равными нулю.

ПРИЛОЖЕНИЕ V

ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ №8

Задание 1

Выдать статистику по городам и специальностям вашего государства. Отчет должен начинаться заголовком и цифрой - общим количеством записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: ГОРОД <название> - ВСЕГО <цифра> (где <цифра> не равна нулю), а заканчивается перечнем шифров специальностей с цифрами, не равными нулю.

Задание 2

Выдать статистику по городам и учебным заведениям вашего государства. Отчет должен начинаться соответствующим заголовком. Далее должна следовать совокупность таблиц, каждая из которых включает перечень учебных заведений одного города с цифрами, не равными нулю (последняя строка таблицы представляет собой: ГОРОД <название> - ВСЕГО <цифра>, где <цифра> не равна 0). Заканчивается отчет цифрой - общим количеством записей по запросу.

Задание 3

Выдать статистику по городам и категориям обучения вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: ГОРОД <название> - ВСЕГО <цифра>, где <цифра> $\neq 0$, а заканчивается перечнем категорий обучения с цифрами, не равными нулю.

Задание 4

Выдать статистику по всем учебным заведениям и категориям обучения вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: <название учебного заведения> - <цифра>, не равная нулю, а заканчивается перечнем категорий обучения с цифрами, не равными нулю.

Задание 5

Выдать статистику по учебным заведениям и категориям обучения вашего государства, включая цифры по выпуску 1993 года и по приему 1990 года по каждой категории обучения. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: <название учебного заведения> - <цифра>, не равная нулю, а заканчивается перечнем категорий обучения с тремя цифрами: 1) всего (не равная нулю); 2)

выпуск 1993 г.; 3) прием 1990 г.

Задание 6

Выдать статистику по учебным заведениям и годам окончания вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: <название учебного заведения> <цифра>, не равная нулю, а заканчивается перечнем годов окончания с цифрами, не равными нулю.

Задание 7

Выдать статистику по выпуску 1994 года вашего государства по всем учебным заведениям и категориям обучения. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу (выпуску 1994 г.). Далее перечень таблиц, где каждая таблица начинается с заголовка: <название учебного заведения> - <цифра>, не равная нулю, а заканчивается перечнем категорий обучения с цифрами, не равными нулю.

Задание 8

Выдать статистику по годам приема и по специальностям учащихся вашего государства. Отчет должен начинаться с заголовка и цифры общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: <год приема> - <цифра>, не равная нулю, а заканчивается перечнем шифров специальностей с цифрами, не равными нулю.

Задание 9

Выдать статистику по категориям обучения и учебным заведениям вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: <категория обучения> - <цифра>, не равная нулю, а заканчивается перечнем учебных заведений с цифрами, не равными нулю.

Задание 10

Выдать статистику по выпуску 1993 года вашего государства по всем учебным заведениям и специальностям. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу (выпуску 1993 г.). Далее перечень таблиц, где каждая таблица начинается с заголовка: <название учебного заведения> - <цифра>, не равная нулю, а заканчивается перечнем шифров специальностей с цифрами, не равными нулю.

Задание 11

Выдать статистику по приему 1991 года вашего государства по всем категориям обучения и городам. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу (по приему 91-го года). Далее совокупность таблиц, где каждая таблица начинается с заголовка: <название категории обучения> - <цифра>, не равная нулю, а заканчивается

перечнем названий городов с цифрами, не равными нулю.

Задание 12

Выдать статистику по специальностям и городам вашего государства. Отчет должен начинаться заголовком и цифрой - общим количеством записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: СПЕЦИАЛЬНОСТЬ <шифр> - ВСЕГО <цифра>, не равная нулю, а заканчивается перечнем названий городов с цифрами, не равными нулю.

Задание 13

Выдать статистику по городам и годам окончания вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: ГОРОД <название> - ВСЕГО <цифра>, не равная нулю, а заканчивается перечнем годов окончания с цифрами, не равными нулю.

Задание 14

Выдать статистику по лицам мужского пола вашего государства по всем городам и учебным заведениям. Отчет должен начинаться соответствующим заголовком. Далее должна следовать совокупность таблиц, каждая из которых включает перечень учебных заведений одного города с цифрами, не равными нулю (последняя строка таблицы представляет собой: ГОРОД <название> - ВСЕГО <цифра>, где <цифра> $\neq 0$). Заканчивается отчет цифрой - общим количеством записей по запросу (т. е. всего мужчин).

Задание 15

Выдать статистику по специальностям и учебным заведениям вашего государства. Отчет должен начинаться с заголовка и цифры - общего количества записей по запросу. Далее совокупность таблиц, где каждая таблица начинается с заголовка: СПЕЦИАЛЬНОСТЬ <шифр> - ВСЕГО <цифра>, не равная нулю, а заканчивается перечнем названий учебных заведений с цифрами, не равными нулю.

ПРИЛОЖЕНИЕ VI

ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ "УЧАЩИЙСЯ"

Предметная область включает в себя шесть объектов:

- 1) Информационный объект "Учащиеся России";
- 2) вспомогательный объект "вузы-города-страны СНГ";
- 3) объект "коды и значения учебных заведений";
- 4) объект "коды и значения семейного положения";
- 5) объект "коды и значения категории обучения";
- 6) объект "коды и значения городов".

В БД каждому объекту предметной области соответствует своя таблица.

Структура информационной таблицы "Россия" включает 12 полей:

Номер - цифровое поле (целое число из 6 цифр);
Фамилия Имя Отчество - символьное поле (40 символов);
Дата рождения - символьное поле (6 символов);
Пол - символьное поле (1 символ);
Семейное положение (код) - символьное поле (1 символ);
Специальность - символьное поле (6 символов);
Категория обучения (код) - символьное поле (2 символа);
Учебное заведение (код) - цифровое поле (целое число из 7 цифр);
Месяц приема - символьное поле (2 символа);
Год приема - символьное поле (2 символа);
Месяц окончания - символьное поле (2 символа);
Год окончания - символьное поле (2 символа).

Структура вспомогательной таблицы ВУЗы включает в себя 6 полей:

Учебное заведение (код) - цифровое поле (целое число из 7 цифр);
Учебное заведение (значение) - символьное поле (25 символов);
Город (код) - символьное поле (3 символа);
Город (значение) - символьное поле (20 символов);
Страна (код) - символьное поле (2 символа);
Страна (значение) - символьное поле (15 символов).

Структура оставшихся 4-х таблиц (таблиц словаря БД) одинакова и состоит из двух полей: код и раскодированное значение.

Структура таблицы "Коды и значения учебных заведений":

Код - цифровое поле (целое число из 7 цифр);

Значение - символьное поле (25 символов).

Структура таблицы "Коды и значения семейного положения":

Код - символьное поле (1 символ);

Значение - символьное поле (12 символов).

Структура таблицы "Коды и значения категории обучения":

Код - символьное поле (2 символа);

Значение - символьное поле (16 символов).

Структура таблицы "Коды и значения городов":

Код - символьное поле (3 символа);

Значение - символьное поле (20 символов).

ПРИЛОЖЕНИЕ VII

СХЕМА БД "УЧАЩИЙСЯ"



СОДЕРЖАНИЕ

Лабораторная работа №1	3
Лабораторная работа №2	10
Лабораторная работа №3	19
Лабораторная работа №4	23
Лабораторная работа №5	27
Лабораторная работа №6	34
Лабораторная работа №7	37
Лабораторная работа №8	41
Лабораторная работа №9	48
Приложение I	55
Приложение II	57
Приложение III	60
Приложение IV	63
Приложение V	66
Приложение VI	69
Приложение VII	71

Учебно-методическое издание

Федин В.А., Бурцев А.П., Мотина Н.А., Рогова Н.Г.

Лабораторный практикум
по курсу
“Базы данных”

Редактор издательства Е.Н. Касьянова

Темплан издания МЭИ 2002(І), метод.

Подписано к печати

Формат 60х84/16

Физ. печ. л. 4,5

Тираж 50

Изд. №

Заказ

Издательство МЭИ, 111250, Москва, Красноказарменная д.14

Отпечатано в типографии