**Cоздание таблиц БД, загрузка их данными и создание индексов.**

**1) Создание таблицы с именем cir.**

create table cir

(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]

**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]

**5) Создание таблицы с именем kat\_obuch:**

create table kat\_obuch

(cod char (2),

kat\_obucheniya char (16)) [F5]

**6) Создание таблицы с именем gorod:**

create table gorod

(cod char (3),

nazvanie\_goroda char (20)) [F5]

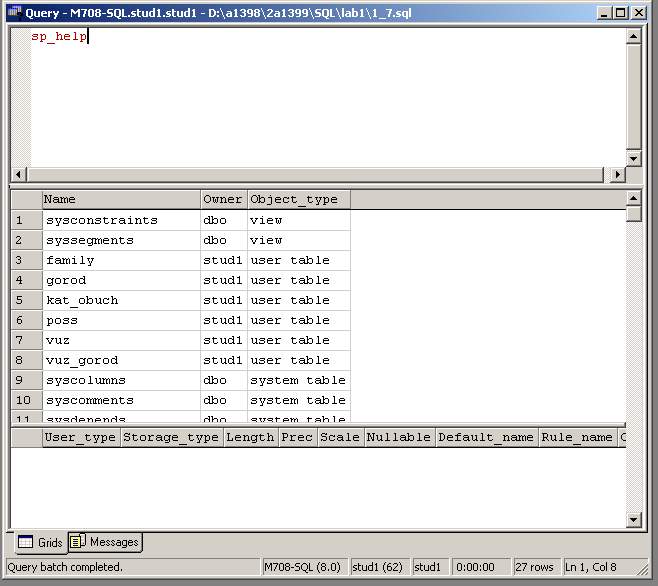


Рис. 4

**7) Загрузка таблицы cir.**

a)insert cir values

(110002,'КОРАБЛИН КАСЬЯН КОНСТАНТИНОВИЧ ','181067','М',

'0','002105','03',2169003,'09','90','09','95')

go

SELECT \* FROM cir [F5]

b)insert kat\_obuch values

('01','П/Ф(ВУЗ) ')

SELECT \* FROM kat\_obuch [F5]

c)insert vuz\_gorod values

(0128955,'НЕВИННОМЫС.ЭНЕРГ.Т-М ','063','НЕВИННОМЫССК ','01','РОССИЯ ')

go

SELECT \* FROM vuz\_gorod [F5]

d)insert vuz values

(2966500,'АБРАМЦЕВСК.ХУД.ПРОМ.УЧ. ')

go

SELECT \* FROM vuz [F5]

e)insert family values

('0','ХОЛОСТ ')

go

SELECT \* FROM family [F5]

f)insert gorod values

('000','НЕ УКАЗАН ')

go

SELECT \* FROM gorod [F5]

**8) Создание индексов для таблицы cir.**

а) create unique index nomind on cir (nomer) [F5]

б) create index polind on cir (pol) [F5]

в) create index semind on cir (sem\_polog\_k) [F5]

г) create index specind on cir (spec) [F5]

д) create index katind on cir (kat\_obuch\_k) [F5]

е) create index vuzind on cir (vuz\_k) [F5]

ж) create index gpind on cir (gp) [F5]

з) create index goind on cir (gok) [F5]

**9) Создание индексов для остальных таблиц БД.**

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]

г) create unique index codsind on family (cod) [F5]

д) create unique index codkind on kat\_obuch (cod) [F5]

е) create unique index codgind on gorod (cod) [F5]

ж) create unique index codvind on vuz (cod) [F5]

**Использование оператора SELECT и некоторых функций для выполнения простых запросов к БД.**

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

select distinct nazvanie\_goroda from vuz\_gorod where cod\_stran='10' [F5]

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

Рис. 5

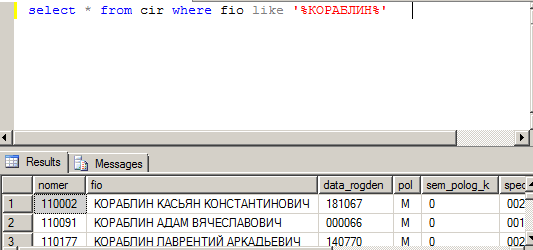
select fio from cir where kat\_obuch\_k between '03' and '04' [F5]

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

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

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

select \* from cir where fio like '%КОРАБЛИН%' [F5]



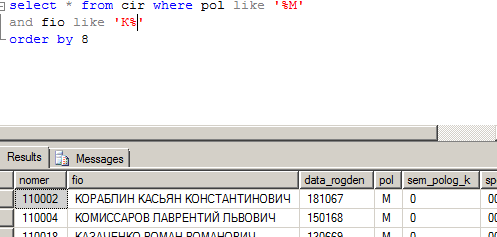
**Персональное задание: задание 10**

Выдать список учащихся вашего государства только мужского пола и с фамилиями, начинающимися на букву К. Отсортировать данные по уче­бным заведениям.

select \* from cir where pol like '%М'

and fio like 'К%' [F5]

order by 8 [F5]

****

**Использование операции соединения нескольких таблиц БД при выполнении сложных списковых запросов.**

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

select fio, vuz.uch\_zavedenie

from cir, vuz

where gok='96'

and vuz\_k=vuz.cod

order by 2 [F5]

или

select fio, vuz.uch\_zavedenie

from cir join vuz on vuz\_k=vuz.cod

where gok='96'

order by 2 [F5]

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

select fio, data\_rogden, vuz.uch\_zavedenie,

pol, spec, kat\_obuch. kat\_obucheniya, mo, gok

from cir, 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 cir join vuz on vuz\_k=vuz.cod

join kat\_obuch on kat\_obuch\_k=kat\_obuch.cod

where gp='91'

order by 3,5

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

select fio, data\_rogden, vuz. uch\_zavedenie, pol,

family.family\_status, spec, kat\_obuch. kat\_obucheniya,

mp, gp, mo, gok

from cir, 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 [F5]

или

select fio, data\_rogden, vuz. uch\_zavedenie, pol,

family.family\_status, spec, kat\_obuch. kat\_obucheniya,

mp, gp, mo, gok

from cir 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]

**Персональное задание: 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.

select fio, data\_rogden, vuz. uch\_zavedenie, pol,

family.family\_status, spec, kat\_obuch. kat\_obucheniya,

mp, gp, mo, gok

from cir 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

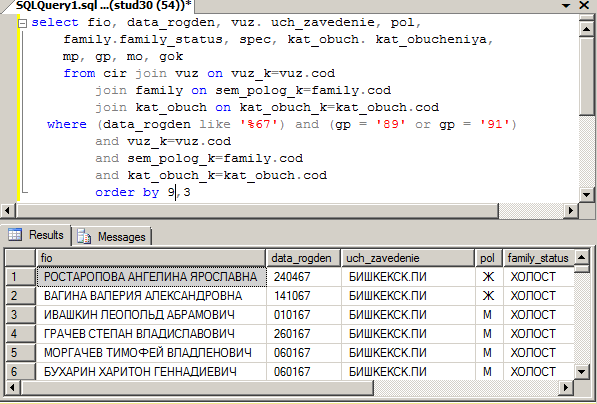
where (data\_rogden like '%67') and (gp = '89' or gp = '91')

and vuz\_k=vuz.cod

and sem\_polog\_k=family.cod

and kat\_obuch\_k=kat\_obuch.cod

order by 9,3 [F5]



**Выполнение запросов с подзапросами.**

**Упражнения:**

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:

select fio, data\_rogden, spec, kat\_obuch.kat\_obucheniya,

vuz.uch\_zavedenie, mo, gok

from cir 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.

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

select fio, data\_rogden, vuz.uch\_zavedenie, pol,

family.family\_status, spec, kat\_obuch.kat\_obucheniya,

mp, gp, mo, gok

from cir 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]

**Персональное задание: 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.

select fio, data\_rogden, vuz.uch\_zavedenie, pol, spec, kat\_obuch.kat\_obucheniya,

mp, gp, mo, gok

from cir join vuz on vuz\_k=vuz.cod

join kat\_obuch on kat\_obuch\_k=kat\_obuch.cod

where (spec between '000601' and '000720') or

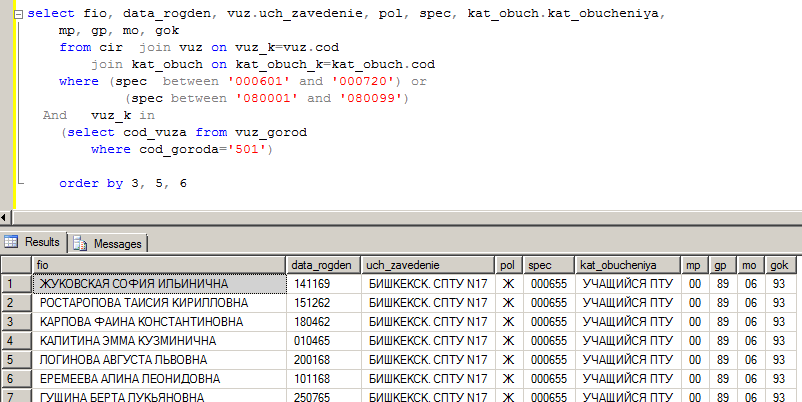
(spec between '080001' and '080099')

And vuz\_k in

(select cod\_vuza from vuz\_gorod

where cod\_goroda='501')

order by 3, 5, 6 [F5]



**Выполнение одномерных статистических запросов по одной и по нескольким таблицам БД.**

.

**Упражнения:**

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).

Таким образом, в верхней области окна необходимо ввести:

1) select count(\*) from cir

where gp='90' or gp='91' [F5]

2) select spec, count(\*) from cir

where gp='90' or gp='91'

group by spec [F5]

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).

1) select count(\*) from cir

where gok='96' [F5]

2) select vuz.uch\_zavedenie, count(\*)

from cir join vuz on vuz\_k=vuz.cod

where gok='96'

group by vuz.uch\_zavedenie [F5]

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).

1) select count(\*) from cir [F5]

2) select vuz\_gorod.nazvanie\_goroda, count(\*)

from cir join vuz\_gorod on vuz\_k=vuz\_gorod.cod\_vuza

group by vuz\_gorod.nazvanie\_goroda [F5]

create procedure zapr1 as

select count(\*) from cir where gp='90' or gp='91'

select spec, count(\*) from cir

where gp='90' or gp='91' group by spec [F5]

**Персональное задание: 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.

select gok as 'Год выпуска',count(\*) as 'Всего' from cir where gok='94'

group by gok

select gok as 'Год выпуска',family.family\_status as 'Семейное положение', count(\*) as 'Кол-во'

from cir join family on sem\_polog\_k=family.cod

where gok='94'

group by gok,family.family\_status

select gok as 'Год выпуска',count(\*) as 'Всего' from cir where gok='95'

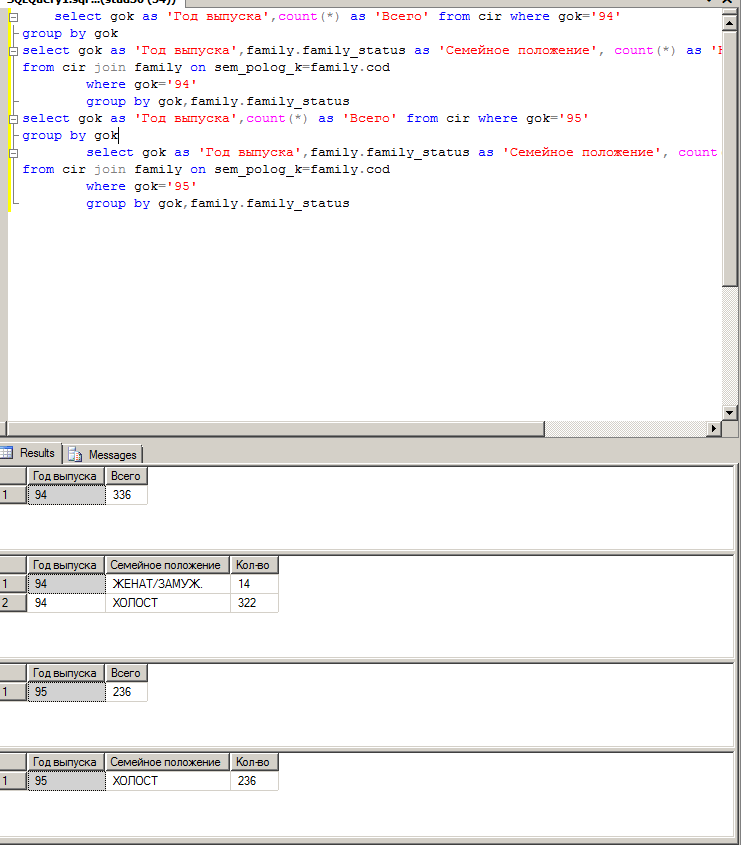
group by gok

select gok as 'Год выпуска',family.family\_status as 'Семейное положение', count(\*) as 'Кол-во'

from cir join family on sem\_polog\_k=family.cod

where gok='95'

group by gok,family.family\_status



**Практическое освоение операторов grant и create view для получения доступа к защищенным данным.**

**Упражнения:**

**1) Создать представление с именем pr1 на основе таблицы cir.**

Допустим, что пользователь АРВ не знает имена полей таблицы cir, но знает, что структура этой таблицы аналогична структуре таблицы poss. Для работы с таблицей cir 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 cir [F5]

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

grant create view to APB [F5]

grant select on cir APB [F5]

**2) Предоставить вспомогательную таблицу vuz\_gorod и четыре таблицы словаря БД в общее пользование.**

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]

**Персональное задание**

Создать представление на основе сложного оператора SELECT (типа оператора SELECT из лабораторной работы №3 или №4)

create view pr20

(fio1, data\_rogden1, pol1,

spec1, kat\_obuch\_k1, vuz\_k1, mp1, gp1)

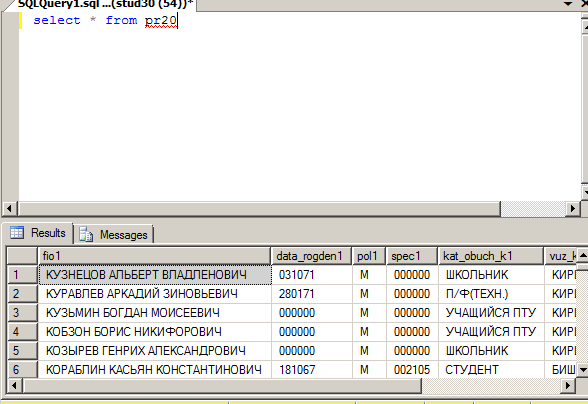
as select fio, data\_rogden, pol, spec, kat\_obuch. kat\_obucheniya, vuz.uch\_zavedenie, mp,gp

from cir join vuz on vuz\_k=vuz.cod

join kat\_obuch on kat\_obuch\_k=kat\_obuch.cod

where pol like '%М'

and fio like 'К%' [F5]



**Использование контрольных ограничений и правил для поддержки целостности данных в БД.**

**Упражнения:**

1) Создание контрольных ограничений на уровне поля.

create table cir1

(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 '?'),

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') [F5]

**Проверка**

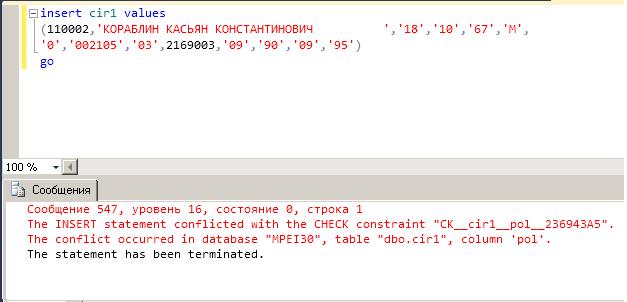
Insert cir2 values

(110002, ‘коравлин касьян константинович’,

‘18’ ‘10’ , ‘67’, ‘м’,

‘0’, 002105’, ’03’, 2169003, ‘09’, ‘90’, ‘95’)

go



2) Создание контрольных ограничений на уровне таблицы.

create table cir2

(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 table\_constr check

((convert (integer, g\_rogd) < convert (integer, gp))

and (convert(integer,gp) < convert(integer,gok))

and

(((kat\_obuch\_k between '07' and '09') and (not (spec like '00%')))

or

(not (kat\_obuch\_k between '07' and '09') and spec like '00%'))

and

((pol='Ж' and fio like '%А % %')

or

(pol='М' and fio not like '%А % %')))) [F5]

**3) Создание правила**

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

sp\_bindrule 'rule1', 'cir2.[kat\_obuch\_k]' [F5]

**Проверка**

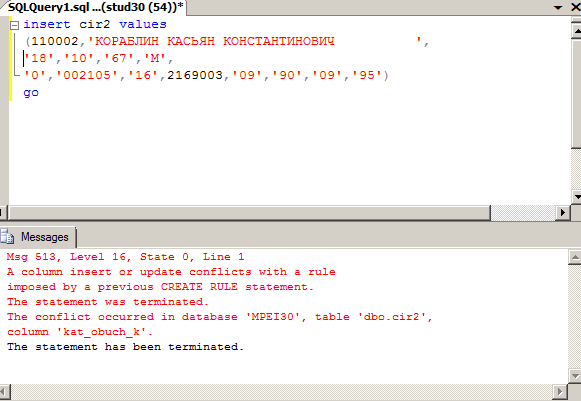
Insert cir2 values

(110002, ‘коравлин касьян константинович’,

‘18’ ‘10’ , ‘67’, ‘м’,

‘0’, 002105’, ’16’, 2169003, ‘09’, ‘90’, ‘95’)

Go



**Использование сложных хранимых процедур (с элементами программирования) для выполнения двумерных и трехмерных статистических запросов к БД.**

**Персональное задание: 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

create procedure z10 as

declare @yz char (25)

declare @sp char (6)

declare @it1 int

declare @it2 int

declare @itall int

declare @yz1 char (25)

declare y cursor for

select vuz.uch\_zavedenie, spec, count (\*)

from cir join vuz on vuz\_k=vuz.cod

where gok='93'

group by vuz.uch\_zavedenie, spec

order by vuz.uch\_zavedenie, spec

select @it2=0

select @itall=0

open y

fetch y into @yz, @sp, @it1

if (@@fetch\_status=-2) begin

print ‘Ошибка при выполнении первого FETCH’

close y

return

end

if (@@fetch\_status=-1) begin

print ‘Данные не найдены’

close y

return

end

select @yz1=@yz

print @sp+' -'+str(@it1)

select @it2=@it2+@it1

select @itall=@itall+@it1

while (@@fetch\_status=0)

begin

fetch y into @yz, @sp, @it1

if (@yz1 !=@yz) begin

print 'учебное заведение' + @yz1 + ' - ' +str(@it2)

select @yz1=@yz

select @it2=0

end

if (@@fetch\_status=-1) break

print @sp + ' - ' +str(@it1)

select @it2=@it2+@it1

select @itall=@itall+@it1

end

close y

if (@@fetch\_status=-2) begin

print ‘Ошибка при выполнении первого FETCH’

return

end

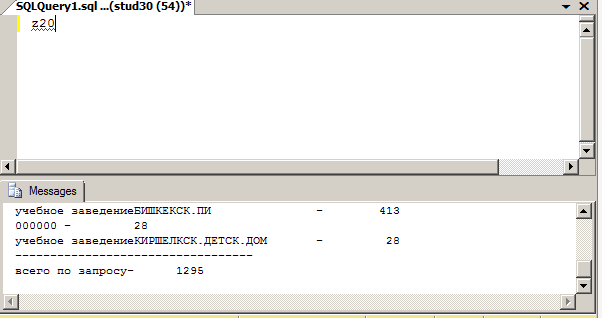
print 'учебное заведение' + @yz + ' - ' +str(@it2)

print '----------------------------------'

print 'всего по запросу-'+str(@itall)

deallocate y

return [F5]



**Создание и использование элементарных триггеров, а также**

**использование предопределенных таблиц в теле триггеров.**

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]

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]

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]

**Персональное задание: задание 10**

create table codko (kocod char(2))

С/\*заполним таблицу категориями обучения от 01 до 15\*/

declare @i integer

set @i=1

while (@i<16)

begin

insert into codko values(@i)

set @i=@i+1

end [F5]

create trigger trig\_ko on cir1

for insert as

if (select count(\*) from codko, inserted

where codko.kocod=inserted.kat\_obuch\_k)=0

select 'Недопустимое значение поля категория обучения' [F5]

Для оператора INSERT:

create trigger insert\_trig on vuz\_gorod

for insert as

begin

insert into vuz (cod,uch\_zavedenie)

select cod\_vuza, nazvanie\_vuza from inserted

select 'Произведено автоматическое внесение изменений в таблицу VUZ'

end [F5]

Для оператора DELETE:

create trigger delete\_trig on vuz\_gorod

for delete as

begin

delete vuz from deleted, vuz

where vuz.cod=deleted.cod\_vuza

select 'Произведено автоматическое внесение изменений в таблицу VUZ'

end [F5]

Для оператора UPDATE:

create trigger update\_trig on vuz\_gorod

for update as

update vuz

set vuz.cod = inserted.cod\_vuza, vuz.uch\_zavedenie = inserted.nazvanie\_vuza

from deleted, vuz, inserted

where ((deleted.cod\_vuza = vuz.cod)

or (deleted.nazvanie\_vuza = vuz.uch\_zavedenie))

select 'Произведено автоматическое внесение изменений в таблицу VUZ'

Свой триггер:

create trigger delete\_trig2 on kat\_obuch

for delete as

begin

delete cir1 from deleted, cir1

where cir1.kat\_obuch\_k=deleted.cod

select 'Произведено автоматическое внесение изменений в таблицу CIR1'

end

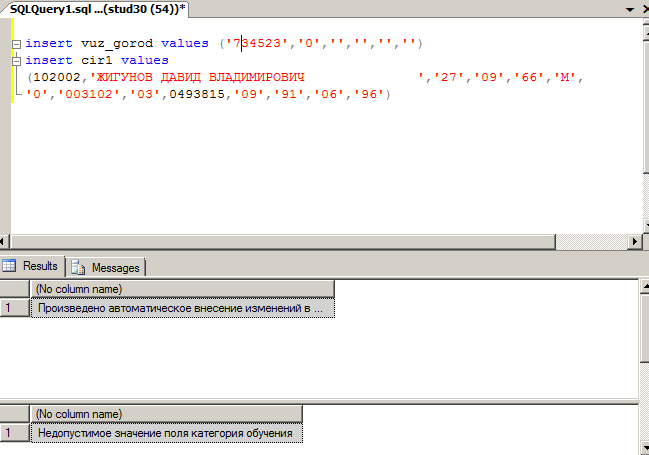
**Проверка**

insert cir1 values

(102002,'ЖИГУНОВ ДАВИД ВЛАДИМИРОВИЧ ','27','09','66','M',

'0','003102','03',0493815,'09','91','06','96')

go

insert vuz\_gorod values ('034523','0','','','','')

**Отображение данных из таблицы БД в Windows-окне.**

**VB.NET** позволяет отображать, анализировать и изменять информацию в базах данных. Для этой цели имеется модель доступа к данным, которая называется **ADO.NET**. Эта модель позволяет работать с мощными базами данных, созданными одной из СУБД, например, **Microsoft Access**, **Microsoft FoxPro**, **Paradox**, **Oracle** и **Microsoft SQL Server**. Далее работа с базой данных будет рассмотрена на примере базы данных СУБД **MS SQL Sever**.

В **ADO.NET** для управления информацией в базе данных необходимо:

1. Подключиться (**Tools – Connect to Database**) к базе данных.

2. Создать набор данных (**Dataset**), содержащий копию данных одной или нескольких таблиц, с которыми планируется работать.

3. Создать и связать с набором данных элементы управления **DataGridView**, предназначенные для отображения и редактирования информации в таблицах набора данных.

Для начала работы выполните следующие действия:

1. Создайте свою рабочую папку.

3. Откройте приложение **Microsoft Visual Studio**.

4. Создайте новый проект.

5. Добавьте в проект еще две формы Form2 и Form3. Для добавления каждой формы следует выполнить команду меню **Project - Add Windows Form**. В открывшемся окне **Add New Item** в поле **Templates** следует выбрать шаблон **Windows Form** и нажать на кнопку **Add**.

6. Действуя аналогично, добавьте в проект модуль **Module1**. Отличие лишь в том, что вместо шаблона **Windows Form** в этом случае следует выбрать шаблон **Module**.

7. Введите код двух инструкций модуля:

|  |
| --- |
| **Module** Module1  Dim frm2 As New Form2  Dim frm3 As New Form3  End **Module** |

Назначение каждой из этих двух инструкций состоит в том, что каждая из них создает экземпляр соответствующей формы. Экземпляр формы Form1 создается системой автоматически.

8. Задайте свойствам **Text** форм **Form1**, **Form2** и **Form3** соответствующие значения: **«Ф.И.О»**, **«Категория обучения»** и **«ВУЗ»**.

9. Поместите на форме **Form1** кнопку **Button1** и кнопку **Button2**. После запуска проекта отображается форма **Form1**. Кнопка **Button1** позволит отобразить форму **Form2**, кнопка **Button2** позволит отобразить форму **Form3**.

10. Поместите на форме **Form2** кнопку **Button1**. Эта кнопка позволит после отображения формы Form2 при необходимости скрыть ее.

11. Поместите на форме **Form3** кнопку **Button1**. Эта кнопка позволит после отображения формы Form3 при необходимости скрыть ее.

12. Свяжите с кнопками **Button1** и **Button2** формы **Form1** код:

|  |
| --- |
| Private Sub Button1\_Click(ByVal sender As System.Object, \_ ByVal e As System.EventArgs) Handles Button1.Click  Form2.Show()  End Sub  Private Sub Button2\_Click(ByVal sender As System.Object, \_ ByVal e As System.EventArgs) Handles Button2.Click  Form3.Show()  End Sub |

13. Свяжите с кнопками **Button1** каждой из форм **Form2** и **Form3** код:

|  |
| --- |
| Private Sub Button1\_Click(ByVal sender As System.Object, \_ ByVal e As System.EventArgs) Handles Button1.Click  Form1.Show()  End Sub |

**Код Form1**

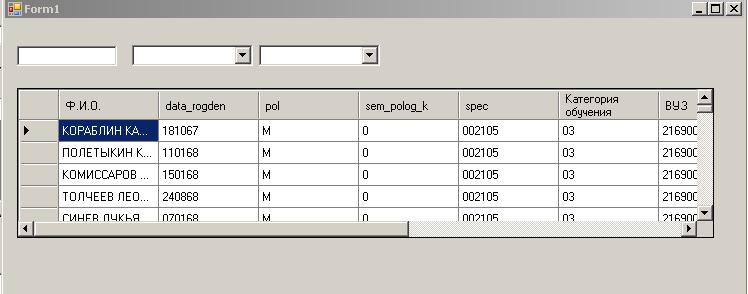
|  |
| --- |
| Public Class Form1  Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load  'TODO: данная строка кода позволяет загрузить данные в таблицу "DataSet1.cir". При необходимости она может быть перемещена или удалена.  Me.CirTableAdapter.Fill(Me.DataSet1.cir)  End Sub  Private Sub TextBox1\_TextChanged(ByVal sender As Object, \_  ByVal e As System.EventArgs) Handles TextBox1.TextChanged  If TextBox1.Text = String.Empty Then  CirBindingSource.RemoveFilter()  Else  CirBindingSource.Filter = "Ф.И.О. like '%" + \_  TextBox1.Text + "%'"  End If  End Sub  Private Sub ComboBox1\_SelectedIndexChanged(ByVal sender \_  As System.Object, ByVal e As \_  System.EventArgs) Handles \_  ComboBox1.SelectedIndexChanged  If ComboBox1.Text = String.Empty Then  CirBindingSource.RemoveFilter()  Else  CirBindingSource.Filter = "Категория обучения = " & \_  ComboBox1.SelectedValue.ToString  End If  End Sub  Private Sub ComboBox2\_SelectedIndexChanged(ByVal sender \_  As System.Object, ByVal e As System.EventArgs) Handles \_  ComboBox2.SelectedIndexChanged  If ComboBox2.Text = String.Empty Then  CirBindingSource.RemoveFilter()  Else  CirBindingSource.Filter = "ВУЗ= " & \_  ComboBox2.SelectedValue.ToString  End If  End Sub  End Class |

**Код Form2**

|  |
| --- |
| Public Class Form2  Private Sub Form2\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load  'TODO: данная строка кода позволяет загрузить данные в таблицу "DataSet1.kat\_obuch". При необходимости она может быть перемещена или удалена.  Me.Kat\_obuchTableAdapter.Fill(Me.DataSet1.kat\_obuch)  End Sub  End Class |

**Код Form3**

|  |
| --- |
| Public Class Form3  Private Sub Form3\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load  'TODO: данная строка кода позволяет загрузить данные в таблицу "DataSet1.vuz". При необходимости она может быть перемещена или удалена.  Me.VuzTableAdapter.Fill(Me.DataSet1.vuz)  End Sub  Private Sub DataGridView1\_CellContentClick(ByVal sender As System.Object, ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick  End Sub  End Class |

**Результаты Form1**