МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА»

(САМАРСКИЙ УНИВЕРСИТЕТ)

Институт информатики и кибернетики

Кафедра информационных систем и технологий

Отчёт по лабораторной работе №3

по курсу «Базы данных».

Выполнил:  
Морозов И.К.

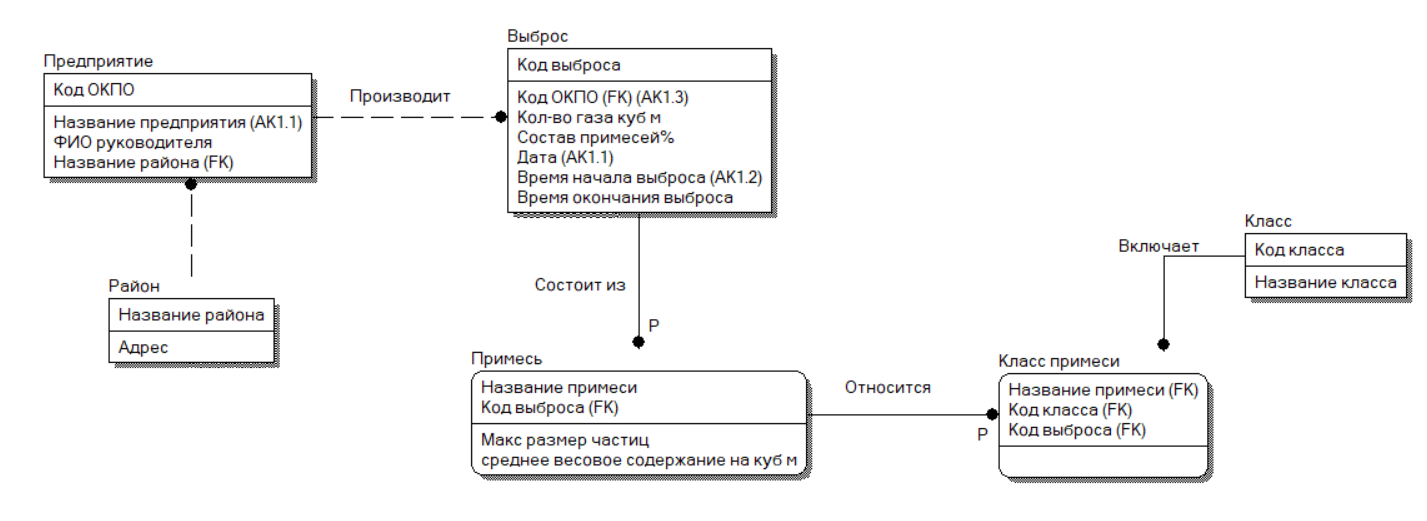
Группа: 6304-090301D

Проверила:  
Чигарина Е.И.

Самара 2023

Создать БД выбросов предприятий. Экологическая служба города ведет учет выбросов в атмосферу дымовых газов различными предприятиями. Каждое предприятие характеризуется кодом ОКПО (уникальным), названием, ФИО руководителя, районом, адресом, выброс – количеством кубометров газа, датой и временем начала и окончания выброса, а также составом примесей в процентах.

Примеси делятся на классы (код класса, название класса, например, твердые, жидкие и газообразные) и характеризуются в выбросе названием, максимальным размеxром частиц и средним весовым содержанием в одном кубометре. Одна и та же примесь может относиться к различным классам.



1. Создать и отобразить все записи из динамического, ключевого и статического курсоров. Обосновать разницу отображения обновленных данных в различных видах курсоров.

--Динамический курсор для обработки информации о выбросах произошедших после 2023-02-01 произведённые предприятием с кодом ОКПО 32874381

declare Dinamic\_cur cursor scroll dynamic

for select Код\_выброса, Колво\_газа\_м3, Состав\_примесей, Дата, Время\_начала\_выброса, Время\_окончания\_выброса

from Выброс where Дата > '2023-02-01' and Код\_ОКПО = 32874381

open Dinamic\_cur

declare @code varchar(8), @gasAmm numeric(18,3), @comp numeric(18,3), @date date, @timeStart time, @timeEnd time

fetch first from Dinamic\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

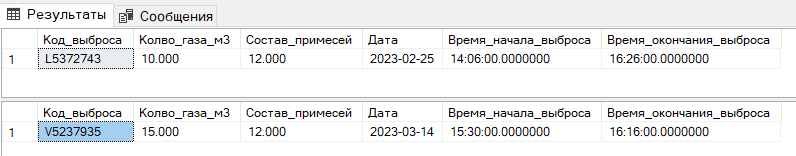
while @@FETCH\_STATUS = 0 begin

select [Код\_выброса] = @code, [Колво\_газа\_м3] = @gasAmm, [Состав\_примесей] = @comp, [Дата] = @date,

[Время\_начала\_выброса] = @timeStart, [Время\_окончания\_выброса] = @timeEnd

fetch next from Dinamic\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

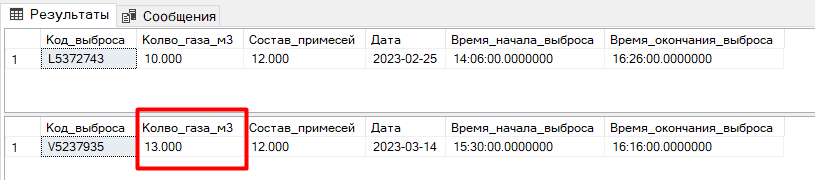
end



update Выброс

set Колво\_газа\_м3 = 13 where Время\_начала\_выброса = '15:30:00'

После обновления:



--Статический курсор для обработки информации о выбросах произошедших после 2023-02-01 произведённые предприятием с кодом ОКПО 32874381

declare Static\_cur cursor scroll static

for select Код\_выброса, Колво\_газа\_м3, Состав\_примесей, Дата, Время\_начала\_выброса, Время\_окончания\_выброса

from Выброс where Дата > '2023-02-01' and Код\_ОКПО = 32874381

open Static\_cur

declare @code varchar(8), @gasAmm numeric(18,3), @comp numeric(18,3), @date date, @timeStart time, @timeEnd time

fetch first from Static\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

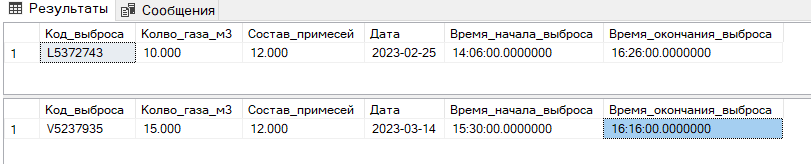
while @@FETCH\_STATUS = 0 begin

select [Код\_выброса] = @code, [Колво\_газа\_м3] = @gasAmm, [Состав\_примесей] = @comp, [Дата] = @date,

[Время\_начала\_выброса] = @timeStart, [Время\_окончания\_выброса] = @timeEnd

fetch next from Static\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

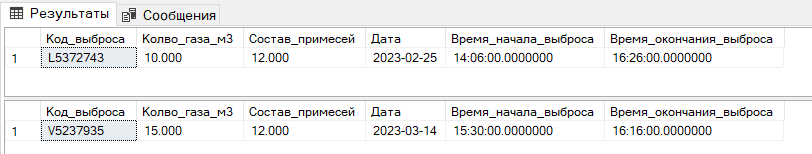
end



update Выброс

set Колво\_газа\_м3 = 13 where Время\_начала\_выброса = '15:30:00'

После обновления:



Для обновления информации в курсоре требуется закрыть и открыть его заново

close Static\_cur

open Static\_cur

declare @code varchar(8), @gasAmm numeric(18,3), @comp numeric(18,3), @date date, @timeStart time, @timeEnd time

fetch first from Static\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

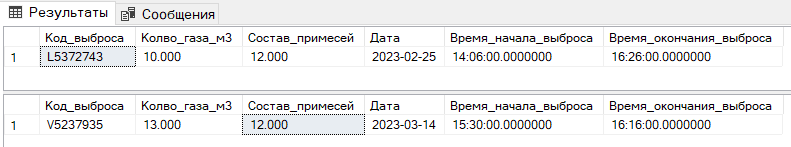
while @@FETCH\_STATUS = 0 begin

select [Код\_выброса] = @code, [Колво\_газа\_м3] = @gasAmm, [Состав\_примесей] = @comp, [Дата] = @date,

[Время\_начала\_выброса] = @timeStart, [Время\_окончания\_выброса] = @timeEnd

fetch next from Static\_cur into @code, @gasAmm, @comp, @date, @timeStart, @timeEnd

end



--Ключевой курсор для отображения информации о существующих классах

declare Key\_cur cursor scroll keyset

for SELECT Код\_класса, Название\_класса FROM Класс

open Key\_cur

declare @ClassCode varchar(8), @Name varchar(50)

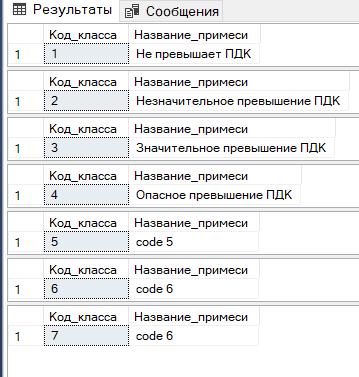
fetch first from Key\_cur into @ClassCode, @Name

while @@FETCH\_STATUS = 0 begin

select [Код\_класса] = @ClassCode, [Название\_примеси] = @Name

fetch next from Key\_cur into @ClassCode, @Name

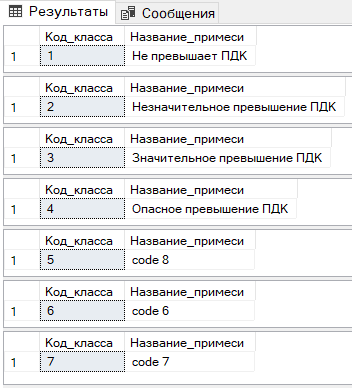
end



update Класс

set Название\_класса = 'code 8' where Код\_класса = 05

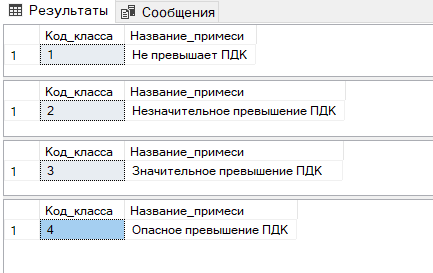
После обновления не ключевого поля:



update Класс

set Код\_класса = 08 where Код\_класса = 05

После обновления ключевого поля:



Вывод:

Динамические курсоры в tempdb хранят ссылки на данные, а не сами данные. Поэтому изменения данных отображаются при открытом курсоре.

Статические курсоры в tempdb хранят сами данные, созданные при первом открытии курсора. Поэтому изменённые данные не отображаются.

Ключевые курсоры в tempdb хранят значения ключей и ссылки на записи с этими ключами. Поэтому отображаются только изменения неключевых данных. При работе с ключевыми данными изменения происходят, но не отображаются, т.к. хранятся те значения ключей, которые были при создании курсора.

1. Создать виды: простой, составной и вид из вида. Отобразить данные из видов. Для составного вида переименовать поля и показать некоторые из них. Обосновать особенности опции WITH CHECK OPTION при работе с видами.

--Вид простой

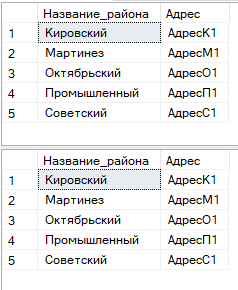
Create view v\_simple (Название\_района, Адрес) as

select Район.Название\_района, Район.Адрес from Район

where Адрес like('Адрес\_\_')

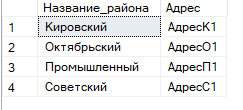
Добавление записи с правильным форматом

В виде и таблице:

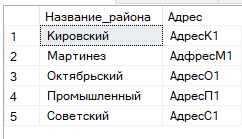


Добавление записи с неправильным форматом

В виде:



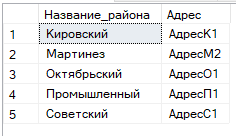
В таблице

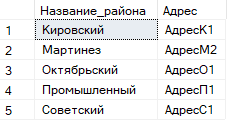


Изменение адреса на правильный формат

update v\_simple

set Адрес = 'АдресМ2' where Название\_района = 'Мартинез'

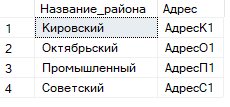




Изменение адреса на неправильный формат

update v\_simple

set Адрес = 'АдфресМ2' where Название\_района = 'Мартинез'

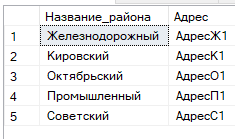


--Вид простой – хранит адреса с ‘корректным’ форматом

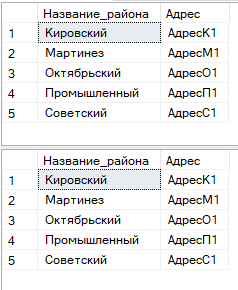
Create view v\_simple (Название\_района, Адрес) as

select Район.Название\_района, Район.Адрес from Район

where Адрес like('Адрес\_\_') with check option



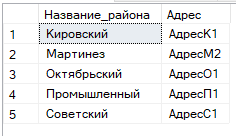
Попытка добавить запись с ‘правильным’ форматом

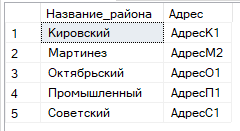


Изменение адреса на правильный формат

update v\_simple

set Адрес = 'АдресМ2' where Название\_района = 'Мартинез'

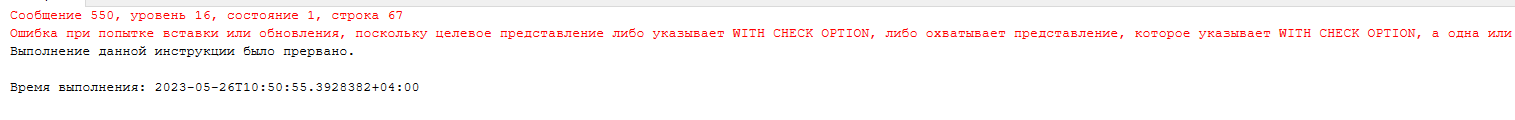




Попытка добавить запись с ‘неправильным’ форматом

insert into v\_simple values

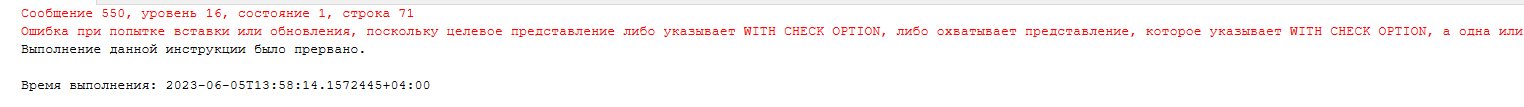
('Мартинез', 'АдфресМ1')



Изменение адреса на неправильный формат

update v\_simple

set Адрес = 'АдфресМ2' where Название\_района = 'Мартинез'



При использовании опции with check option происходит контроль условий вида: записи, не удовлетворяющие условиям вида, не могут быть добавлены или изменены, и выводится сообщение об ошибке.

Без использования – происходит добавление записи в табл., однако она не будет отображаться в виде.

--Вид составной - предприятия производящие выбросы

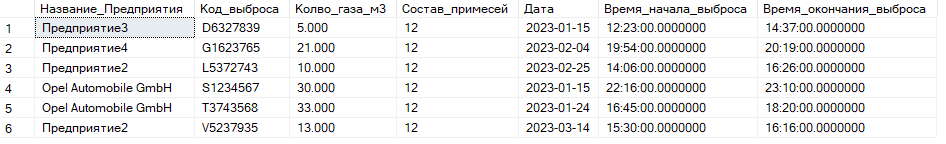
create view v\_compl(Название\_Предприятия, Код\_выброса, Колво\_газа\_м3, Состав\_примесей, Дата, Время\_начала\_выброса, Время\_окончания\_выброса) as

select П.Название\_Предприятия, В.Код\_выброса, В.Колво\_газа\_м3, В.Состав\_примесей, В.Дата, В.Время\_начала\_выброса, В.Время\_окончания\_выброса

from Предприятие П

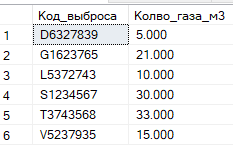
INNER JOIN Выброс В on П.Код\_ОКПО = В.Код\_ОКПО;

select \* from v\_compl



Вывод некоторых полей

select Код\_выброса,Колво\_газа\_м3 from v\_compl



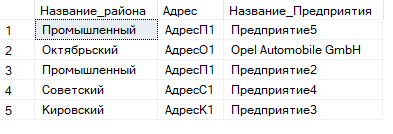
--Вид из вида – адреса компаний

create view v\_from\_v(Название\_района, Адрес, Название\_Предприятия) as

select v.Название\_района, v.Адрес, П.Название\_Предприятия

from v\_simple v

INNER JOIN Предприятие П on v.Название\_района = П.Название\_района;



1. Создать и вызвать хранимую процедуру с входным, выходным формальным параметром и входным параметром, имеющим значение по умолчанию. Реализовать вызов процедуры, используя позиционный, ключевой, смешанный виды передачи параметров.

--Хранимая процедура отображающая сколько всего газа выброшено компанией

create procedure GetEmissionsByCompany

@CompanyCode varchar(8),

@StartDate date,

@EndDate date = NULL,

@TotalEmissions numeric(18,3) output

as

begin

select @TotalEmissions = sum(Колво\_газа\_м3)

from Выброс

where Код\_ОКПО = @CompanyCode

and Дата >= @StartDate

and (Дата <= @EndDate OR @EndDate IS NULL);

end

--Позиционный способ передачи параметров

declare @Emissions numeric(18,3);

exec GetEmissionsByCompany '13246432', '2023-01-01', '2023-05-31', @Emissions OUTPUT;

select @Emissions as TotalEmissions;

--Ключевой способ передачи параметров

declare @Emissions numeric(18,3);

exec GetEmissionsByCompany

@CompanyCode = '13246432',

@StartDate = '2023-01-01',

@EndDate = '2023-05-31',

@TotalEmissions = @Emissions output;

select @Emissions as TotalEmissions;

--Смешанный способ передачи параметров

declare @Emissions numeric(18,3);

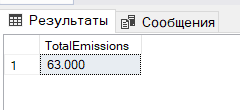
exec GetEmissionsByCompany '13246432', '2023-01-01',

@EndDate = default,

@TotalEmissions = @Emissions output;

select @Emissions as TotalEmissions;

Т.К переданные параметры одинаковые результат всех вызовов один и тот же



1. Создать и вызвать скалярную хранимую функцию.

--Скалярная функция для вычисления среднего размера частиц в примесях, производимых выбранной компанией

create function CalculateAverageParticleSize

(

@CompanyCode varchar(8),

@EmissionType varchar(50)

)

returns numeric(18, 3)

as

begin

declare @AverageParticleSize numeric(18, 3);

select @AverageParticleSize = avg(Макс\_размер\_частиц)

from Примесь p

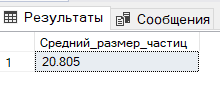
join Выброс v ON p.Код\_выброса = v.Код\_выброса

where v.Код\_ОКПО = @CompanyCode and p.Название\_примеси = @EmissionType;

return @AverageParticleSize;

end

Select [Средний\_размер\_частиц] = dbo.CalculateAverageParticleSize('32874381', 'соединения свинца')



1. Создать и вызвать функцию, возвращающую табличное значение.

--Табличная функция для отображения данных по выбросам и содержащимся в них примесях.

create function GetEmissionsData

(

@CompanyCode varchar(8),

@StartDate date,

@EndDate date

)

returns table

as

return

(

select v.Код\_выброса, v.Код\_ОКПО, v.Колво\_газа\_м3, v.Дата, p.Название\_примеси

from Выброс v

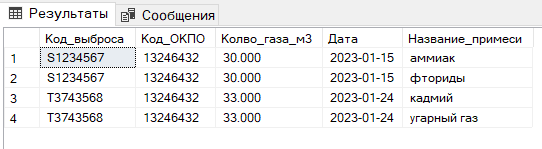
join Примесь p on v.Код\_выброса = p.Код\_выброса

where v.Код\_ОКПО = @CompanyCode and v.Дата >= @StartDate

and v.Дата <= @EndDate

)

select \* from dbo.GetEmissionsData('13246432', '2023-01-01', '2023-02-01');



1. Создать и проверить работу триггера, реализующего уникальность поля, не являющегося первичным ключом.

create trigger RestrictUniq --n

on Класс

for insert, update

as

begin

if(update(Название\_класса) and

(select count(\*) from Класс k join inserted i on i.Название\_класса = k.Название\_класса) >1)

begin

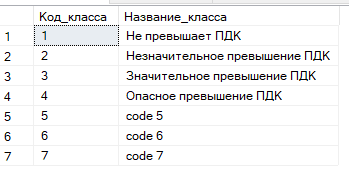
raiserror(N'Класс с таким именем уже существует',16,10)

rollback tran

end

end

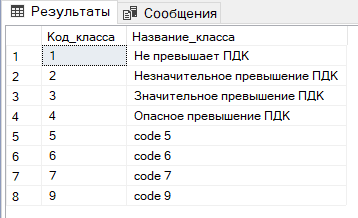
return



Для insert:

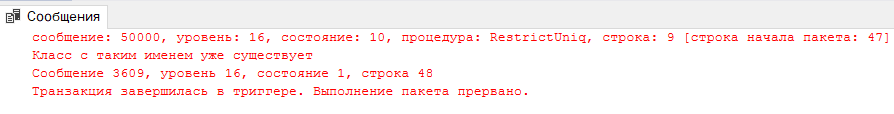
insert into Класс values

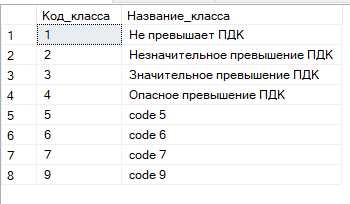
(09, 'code 9')



insert into Класс values

(10, 'code 5')

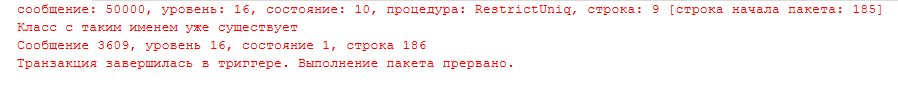




Для update:

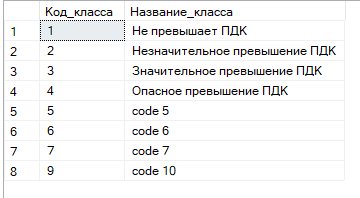
update Класс

set Название\_класса = 'code 5' where Код\_класса = 9



update Класс

set Название\_класса = 'code 10' where Код\_класса = 9



1. Создать и проверить работу триггера, реализующего ссылочную целостность restrict.

Для последующих операций потребуется создать новые таблицы:

create table Выброс1

(

Код\_выброса varchar(8) primary key,

Код\_ОКПО varchar(8),

Колво\_газа\_м3 numeric(18,3) not null,

Состав\_примесей numeric not null,

Дата date not null,

Время\_начала\_выброса time not null,

Время\_окончания\_выброса time not null,

constraint AltKeyNat unique(Время\_начала\_выброса, Дата, Код\_ОКПО)

)

go

create table Примесь1

(

Название\_примеси varchar(50),

Код\_выброса varchar(8),

Макс\_размер\_частиц numeric(18,3),

Среднее\_весовое\_содержание\_на\_м3 numeric(18,3),

constraint PrKeyNat primary key (Код\_выброса, Название\_примеси)

)

go

insert into Выброс1 values--&&&&

('S1234567','13246432', 30, 12, '2023-01-15', '22:16:00', '23:10:00'),

('L5372743','32874381', 10, 12, '2023-02-25', '14:06:00', '16:26:00'),

('V5237935','32874381', 15, 12, '2023-03-14', '15:30:00', '16:16:00'),

('G1623765','74593275', 21, 12, '2023-02-04', '19:54:00', '20:19:00'),

('D6327839','98308334', 5, 12, '2023-01-15', '12:23:00', '14:37:00'),

('T3743568','13246432', 33, 12, '2023-01-24', '16:45:00', '18:20:00');

go

insert into Примесь1 values

('аммиак','S1234567',14.1, 16.642),

('фториды','S1234567',17.3, 19.367),

('соединения свинца','L5372743',20.5, 30.325),

('ртуть','V5237935',19.11, 28),

('соединения свинца','V5237935',21.11, 29.54),

('сероводород','G1623765',15.433, 17.245),

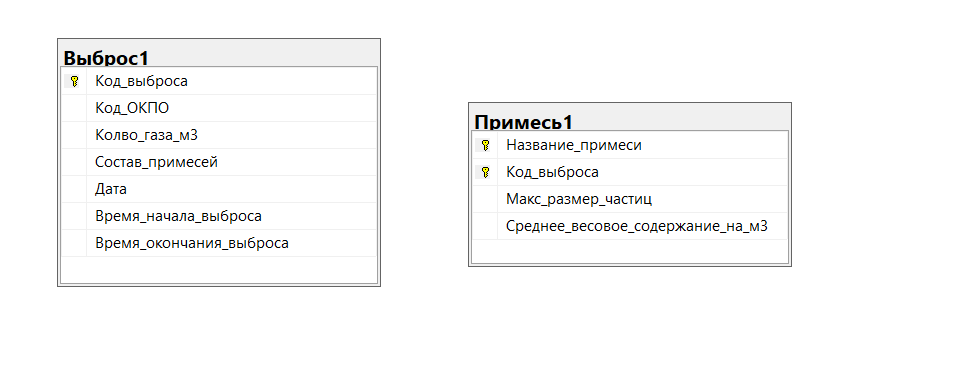
('альдегиды','G1623765',18.563, 25.542),

('озон','D6327839',18.345, 18.532),

('аммиак','D6327839',18.23, 17.53),

('кадмий','T3743568',21.135, 22.245),

('угарный газ','T3743568',16.653, 19.23);



--создание триггера

create trigger RestrictChange

on Выброс1

for delete, update

as

begin

if(exists( select Код\_выброса from deleted del join Примесь1 p on del.Код\_выброса = p.Код\_выброса))

begin

raiserror(N'Нельзя удалить или изменить запись на которую ссылается другая таблица',16,10)

rollback tran

end

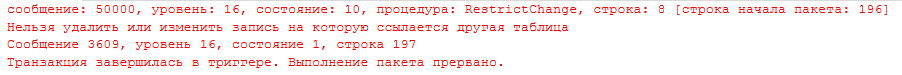
end

return

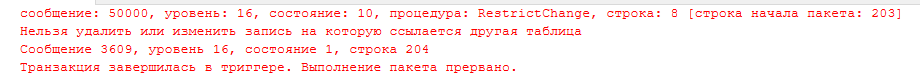
Попытка нарушить ссылочную целостность:

update Выброс1

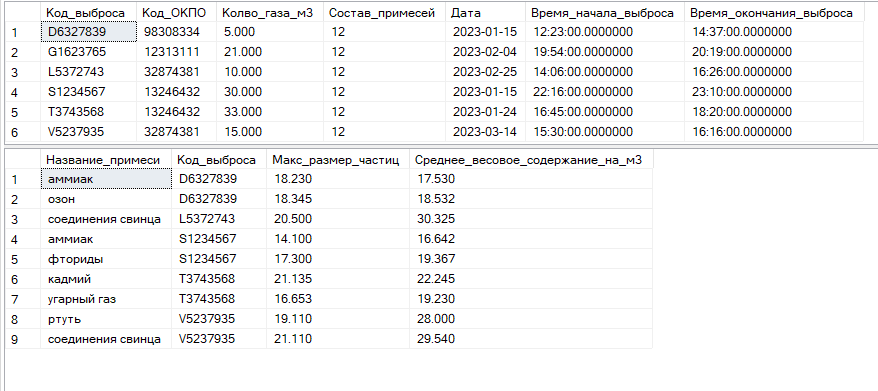
set Код\_ОКПО = '12313111' where Код\_выброса = 'G1623765'



delete from Выброс1 where Код\_выброса = 'G1623765'

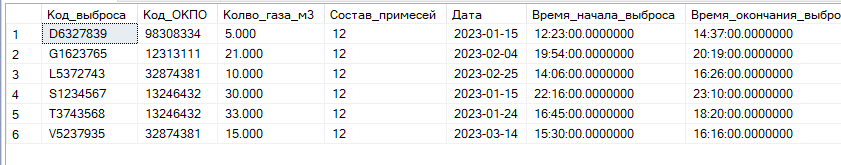


Не нарушая ссылочную целостность:

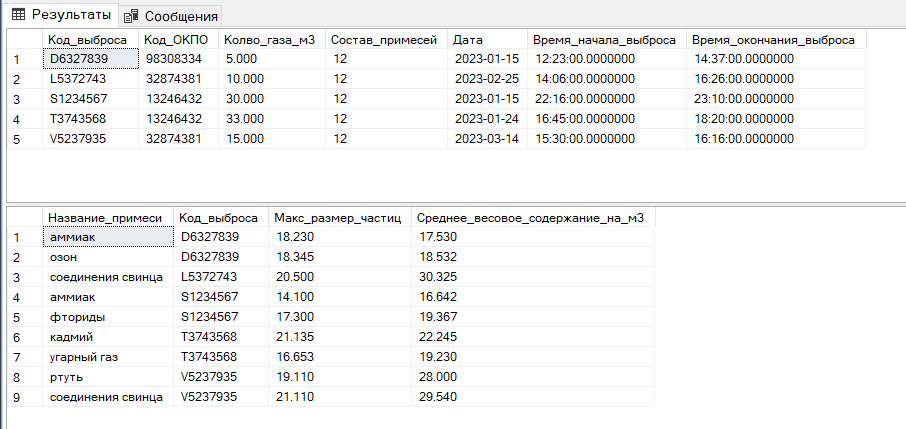


update Выброс1

set Код\_ОКПО = '12313111' where Код\_выброса = 'G1623765'



delete from Выброс1 where Код\_выброса = 'G1623765'



1. Создать и проверить работу триггера, реализующего ссылочную целостность cascade.

create trigger CascadeDelete

on Выброс1

for delete

as

begin

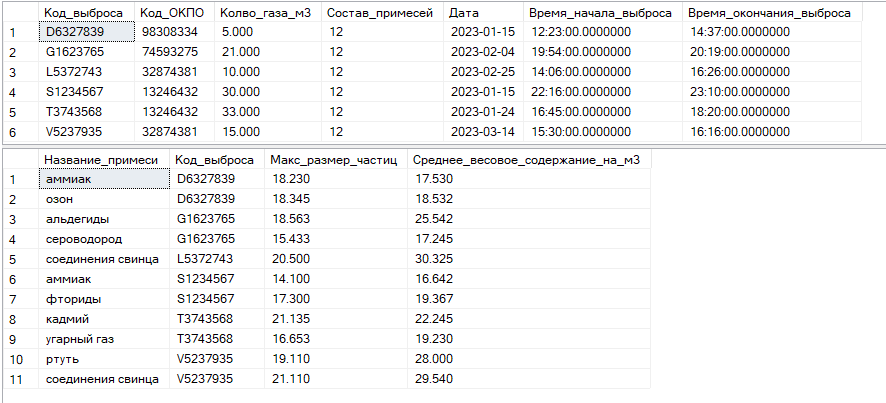
delete from Примесь1

where Код\_выброса in (select Код\_выброса from deleted);

end

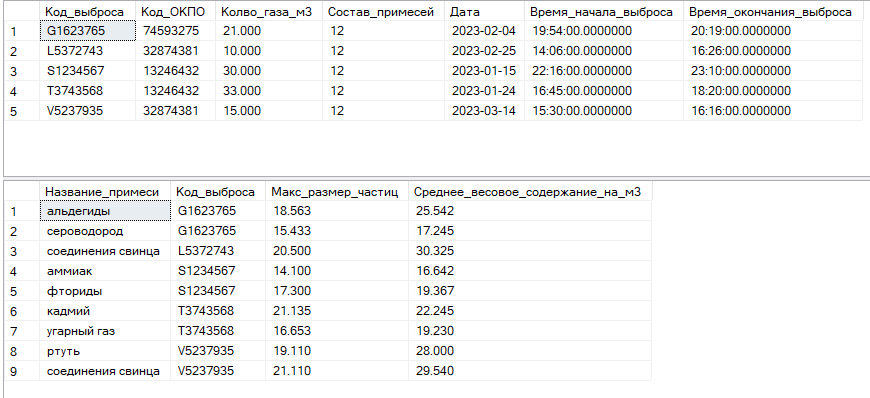
return

До удаления записи:



После:

delete top(1) from Выброс1



Для обновления:

CREATE TRIGGER CascadeUpdate

ON Выброс1

FOR UPDATE

AS

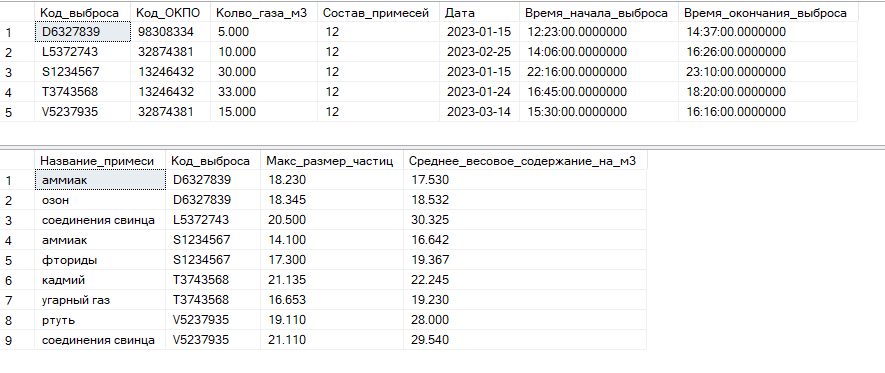
update Примесь1

set Код\_выброса = (select i.Код\_выброса from inserted i)

where Код\_выброса in (select d.Код\_выброса from deleted d)

return

До обновления записи



После:

update Выброс1

set Код\_выброса = 'G3457323' where Код\_выброса = 'D6327839'

