Лабораторная работа №7

Триггер 1.

Назначение: При изменении бренда меняется производитель

CREATE TRIGGER При\_изменении\_бренда

ON товар

AFTER UPDATE

AS

DECLARE @ID INT

SELECT @ID = (SELECT id FROM inserted)

UPDATE Товар

SET Прозводитель = 'Китай'

WHERE ID = @ID

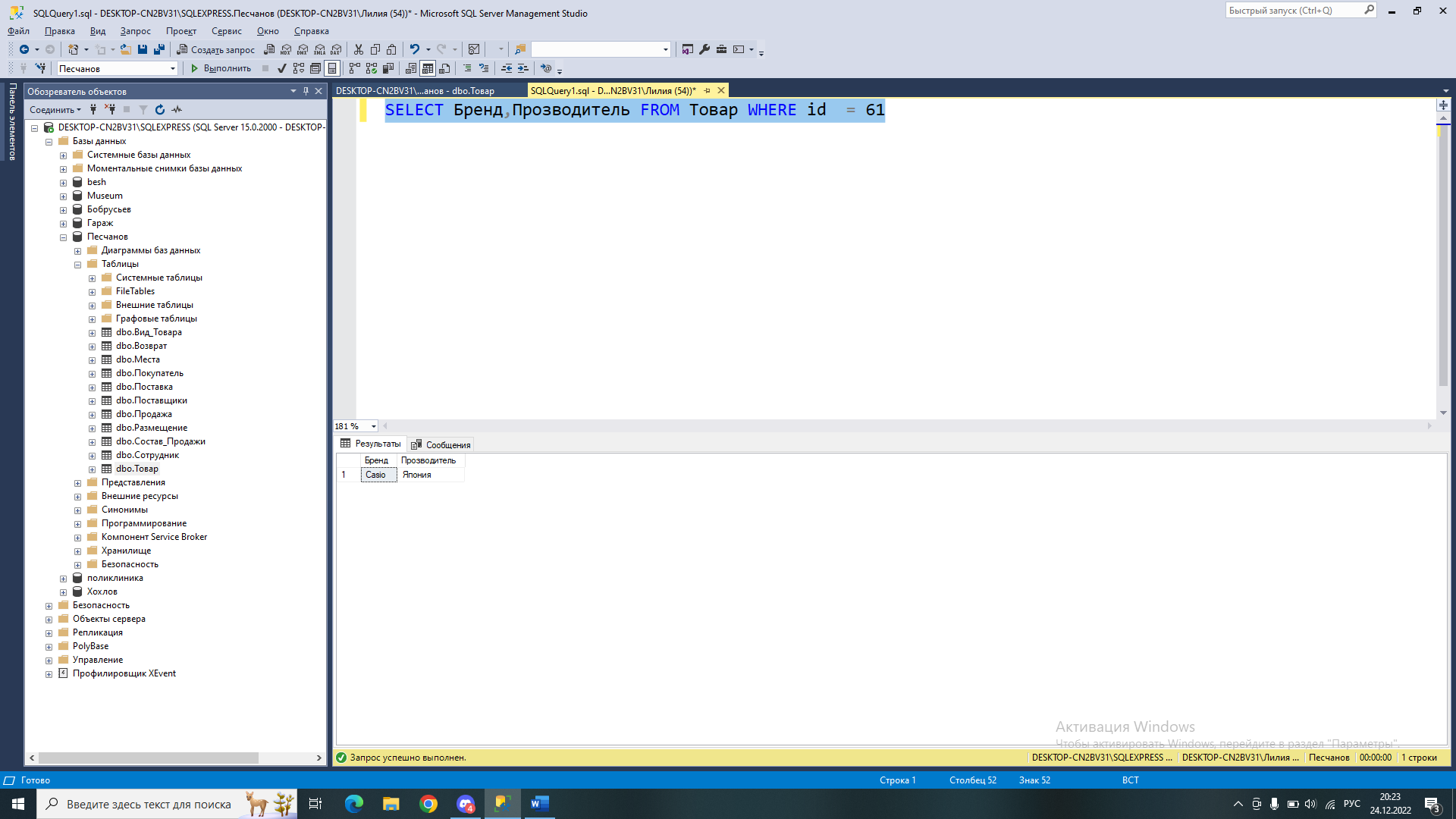
GO

SELECT Бренд,Прозводитель FROM Товар WHERE id = 61

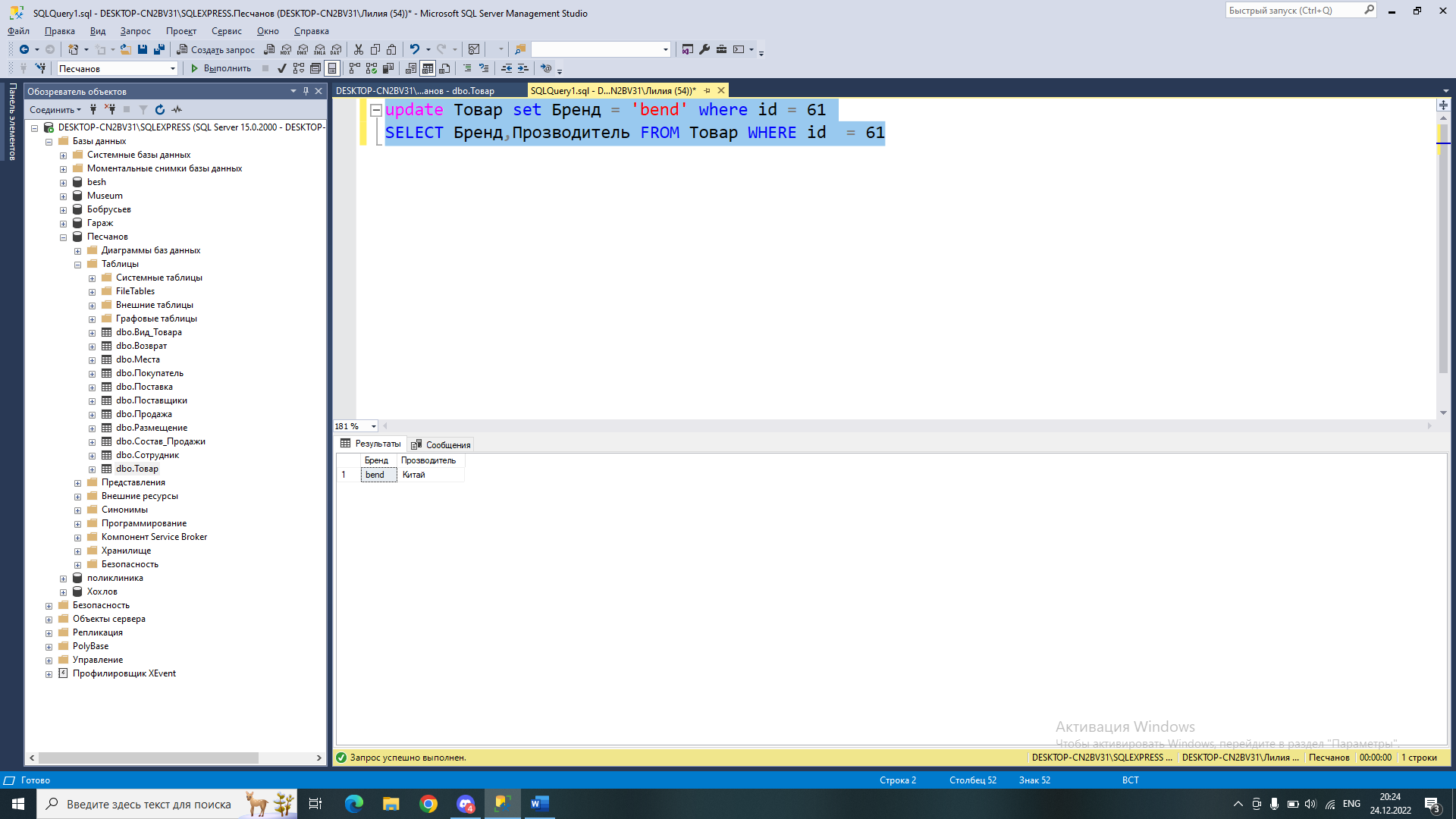
update Товар set Бренд = 'bend' where id = 61

SELECT Бренд,Прозводитель FROM Товар WHERE id = 61

ТОВАР ДО



ТОВАР ПОСЛЕ



Триггер 2.

Назначение: при изменении даты поставки меняется дата приобретения

CREATE TRIGGER При\_изменении\_даты\_поставки

ON поставка

INSTEAD OF UPDATE

AS

DECLARE @ID INT

SELECT @ID = (SELECT id\_Товара FROM deleted)

UPDATE Товар

SET Дата\_Приобретения = '25.05.2023'

WHERE ID = @ID

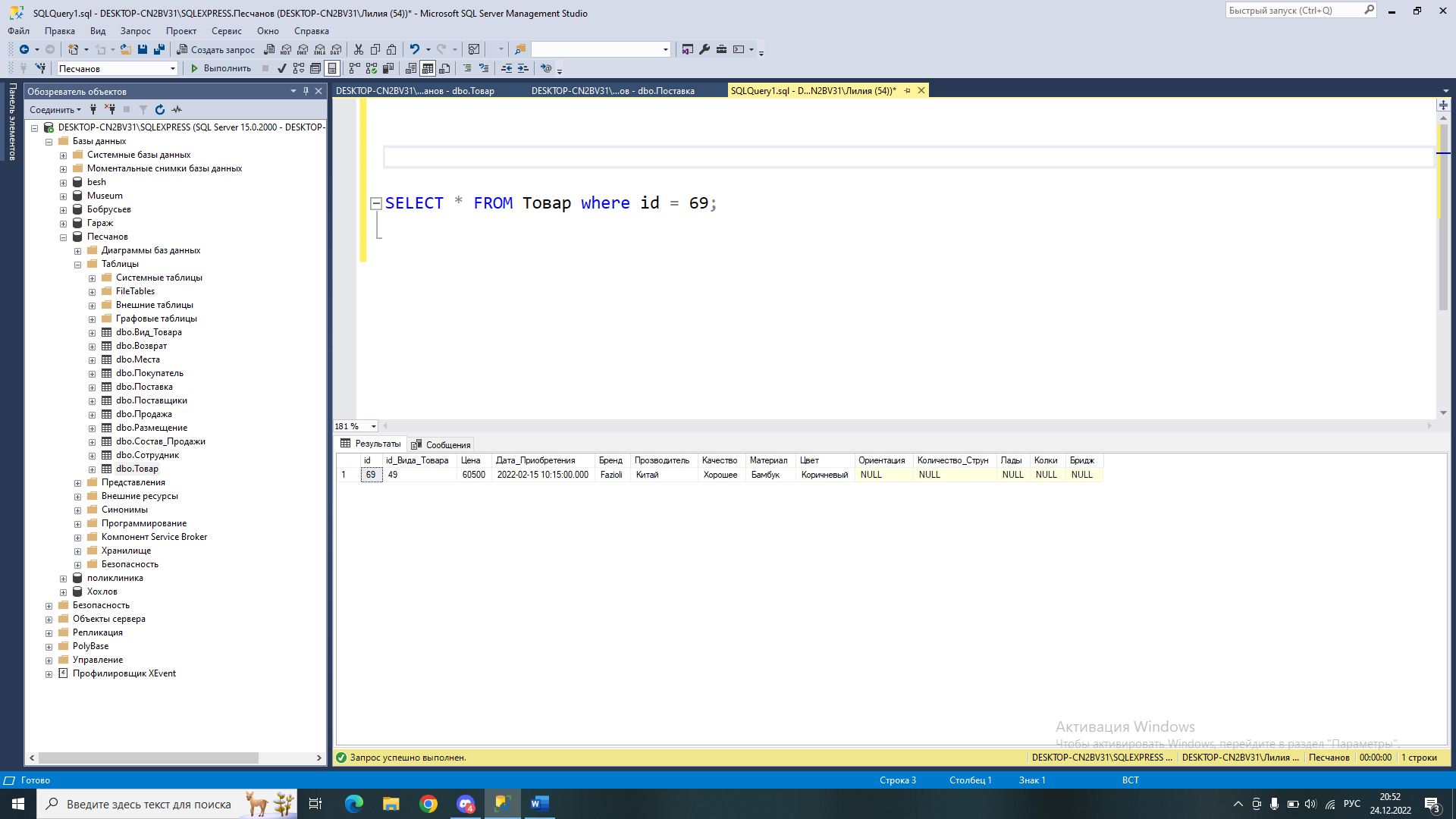
GO

SELECT \* FROM Товар where id = 69;

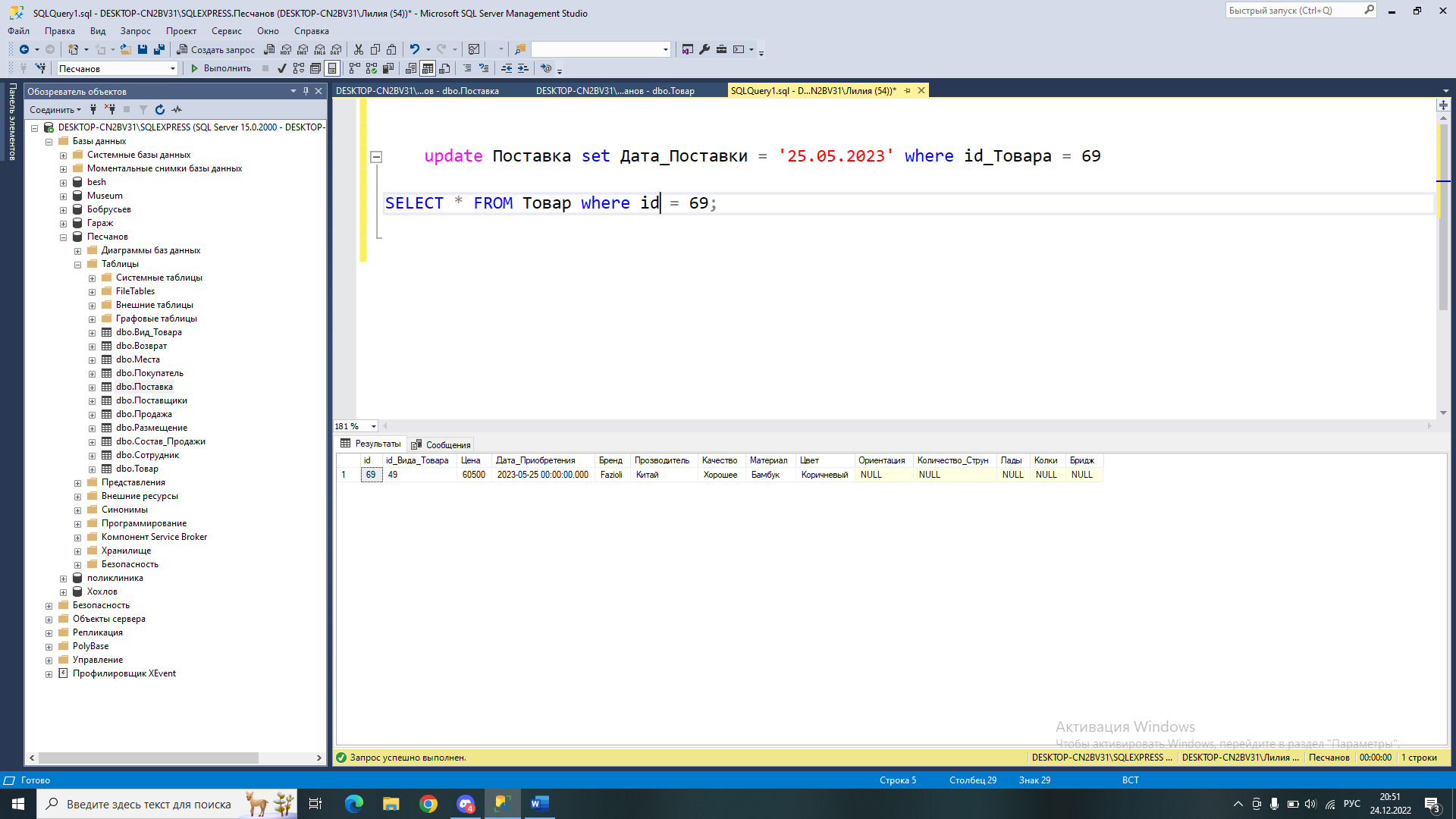
update Поставка set Дата\_Поставки = '25.05.2023'

SELECT \* FROM Товар where id = 69;

ТОВАР ДО:



ТОВАР ПОСЛЕ:



Триггер 3.

Назначение: при изменении номера комнаты меняется номер стеллажа

CREATE TRIGGER Новое\_место

ON места

AFTER update

AS

DECLARE @ID INT

SELECT @ID = (SELECT id FROM deleted)

update Места set Номер\_Стеллажа= '5' where id = @ID

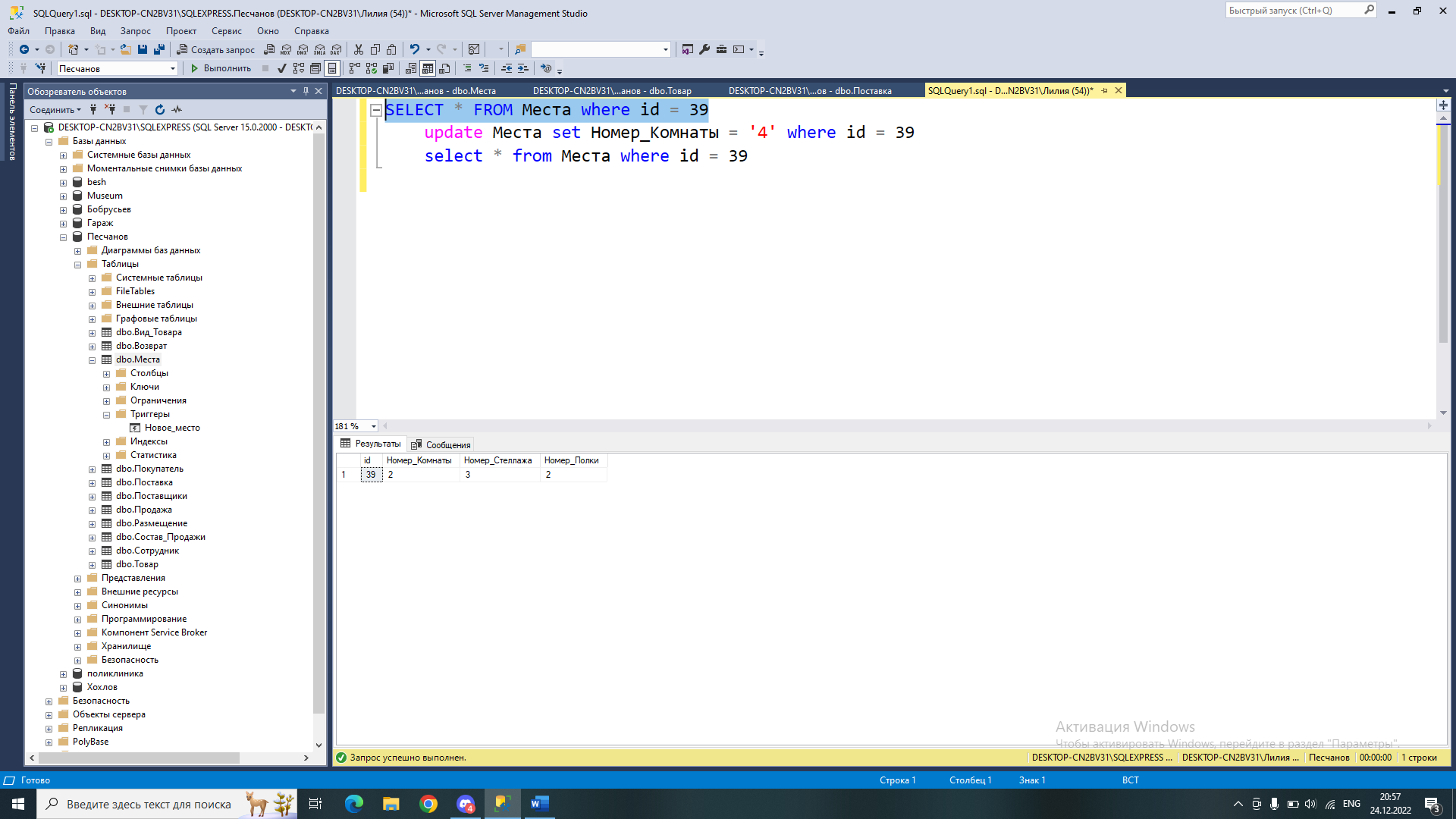
GO

SELECT \* FROM Места where id = 39

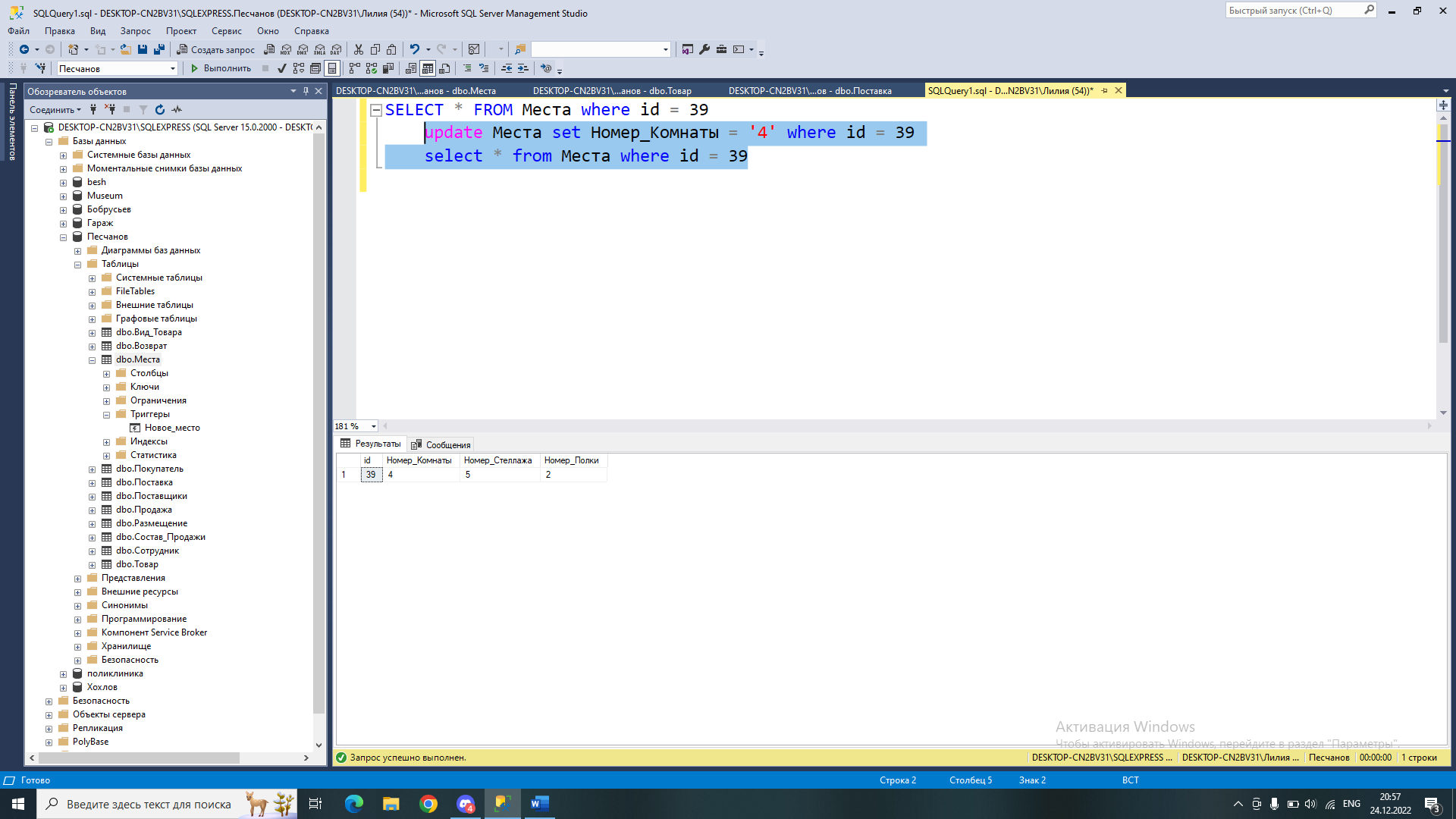
update Места set Номер\_Комнаты = '4' where id = 39

select \* from Места where id = 39

МЕСТА **ДО**



МЕСТА **ПОСЛЕ**



Триггер 4.

Назначение: при изменении материала меняется цена состава продажи

CREATE TRIGGER При\_изменении\_материала

ON товар

AFTER UPDATE

AS

DECLARE @ID INT

SELECT @ID = (SELECT [ID] FROM inserted)

update Состав\_Продажи set Цена = '10000' where id\_Товара = @ID

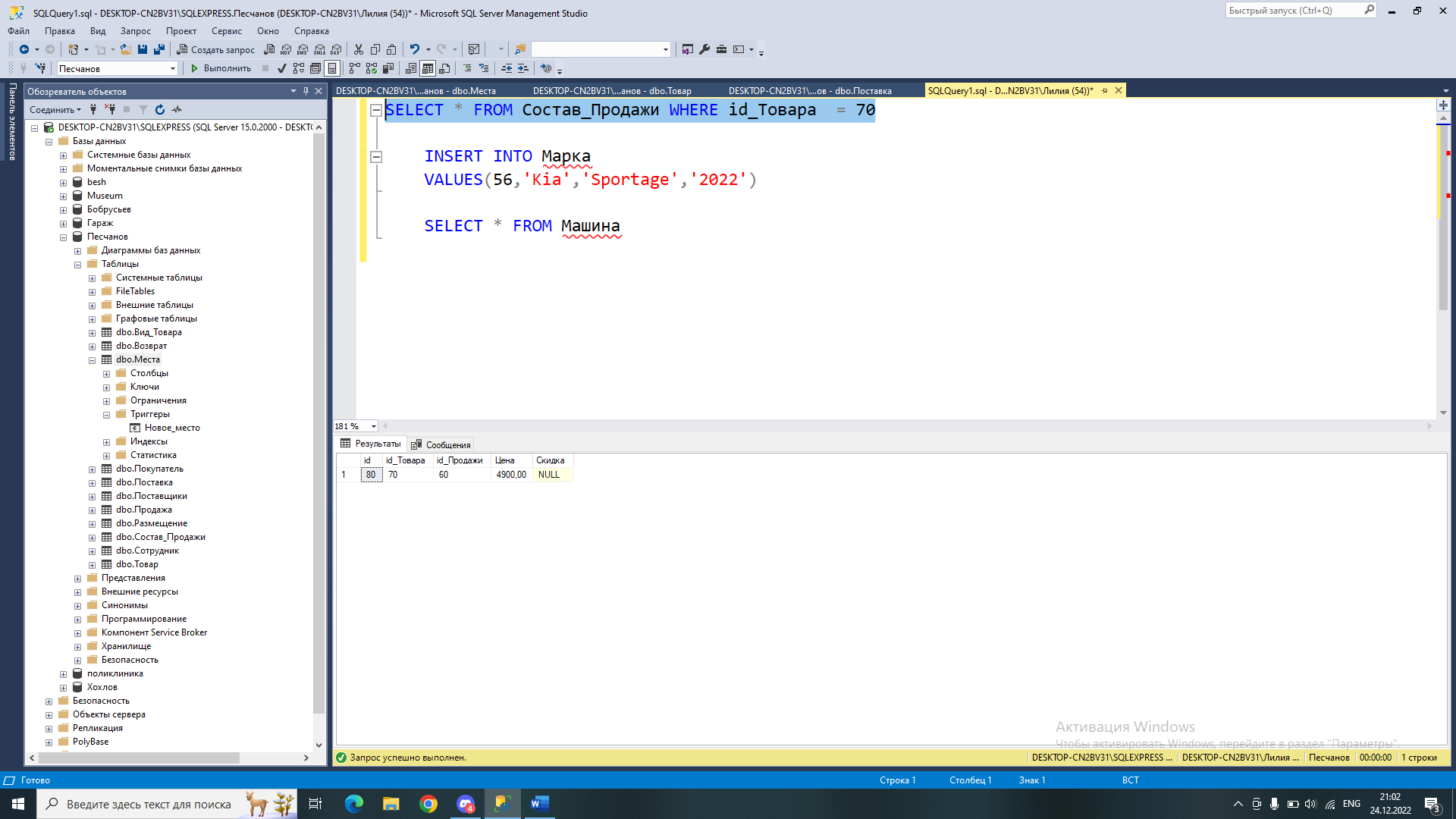
GO

SELECT \* FROM Состав\_Продажи WHERE id\_Товара = 70

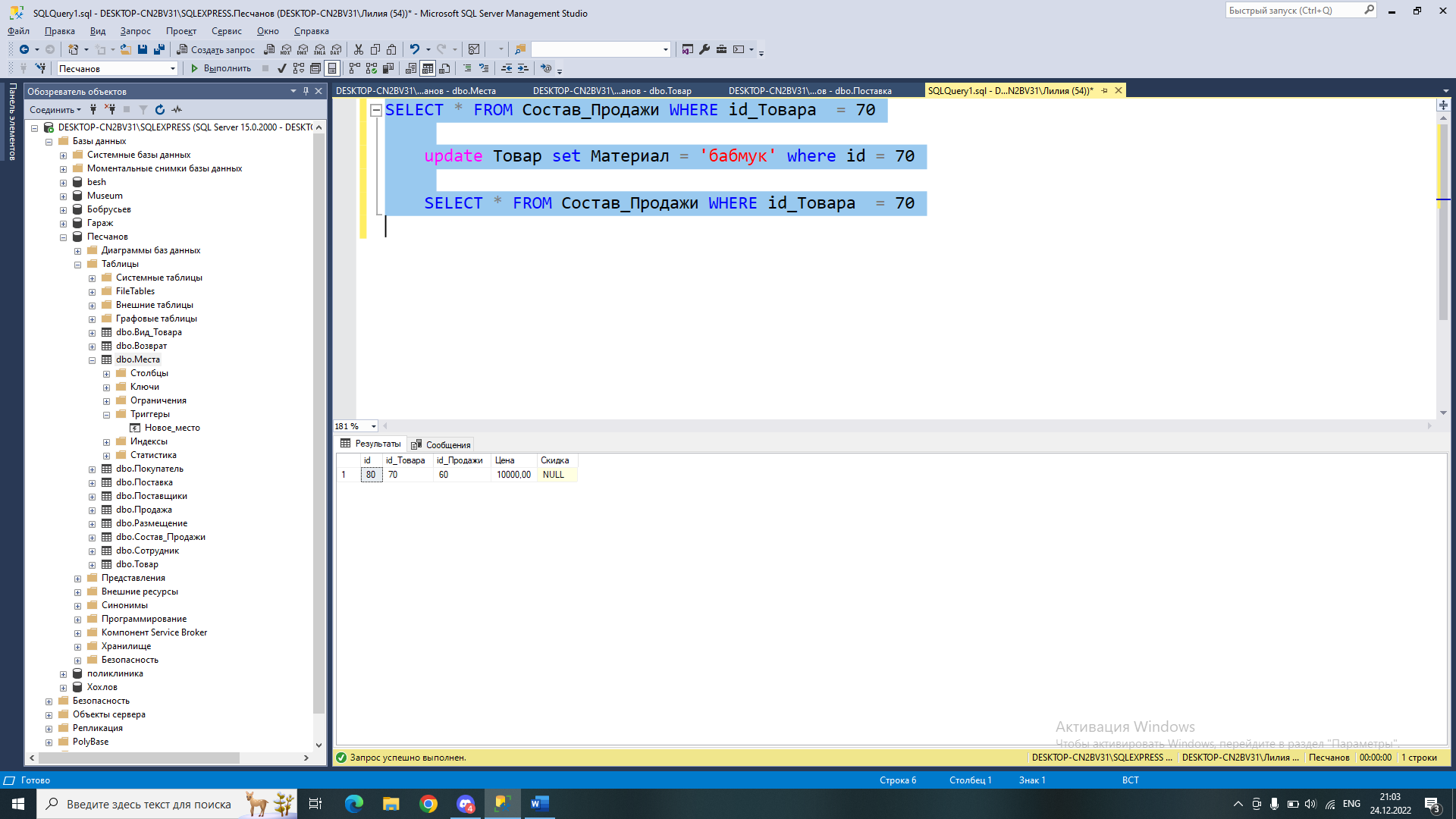
update Товар set Материал = 'бабмук' where id = 70

SELECT \* FROM Состав\_Продажи WHERE id\_Товара = 70

СОСТАВ ПРОДАЖИ **ДО**



СОСТАВ ПРОДАЖИ ПОСЛЕ



Триггер 5.

Назначение: при добавлении размещения добавляется товар

CREATE TRIGGER При\_добавлении\_товара

ON товар

INSTEAD OF INSERT

AS

DECLARE @ID INT

SELECT @ID = (SELECT [ID] FROM inserted)

insert Размещение values (555,@ID,8,39,'15.03.2022','19.08.2028')

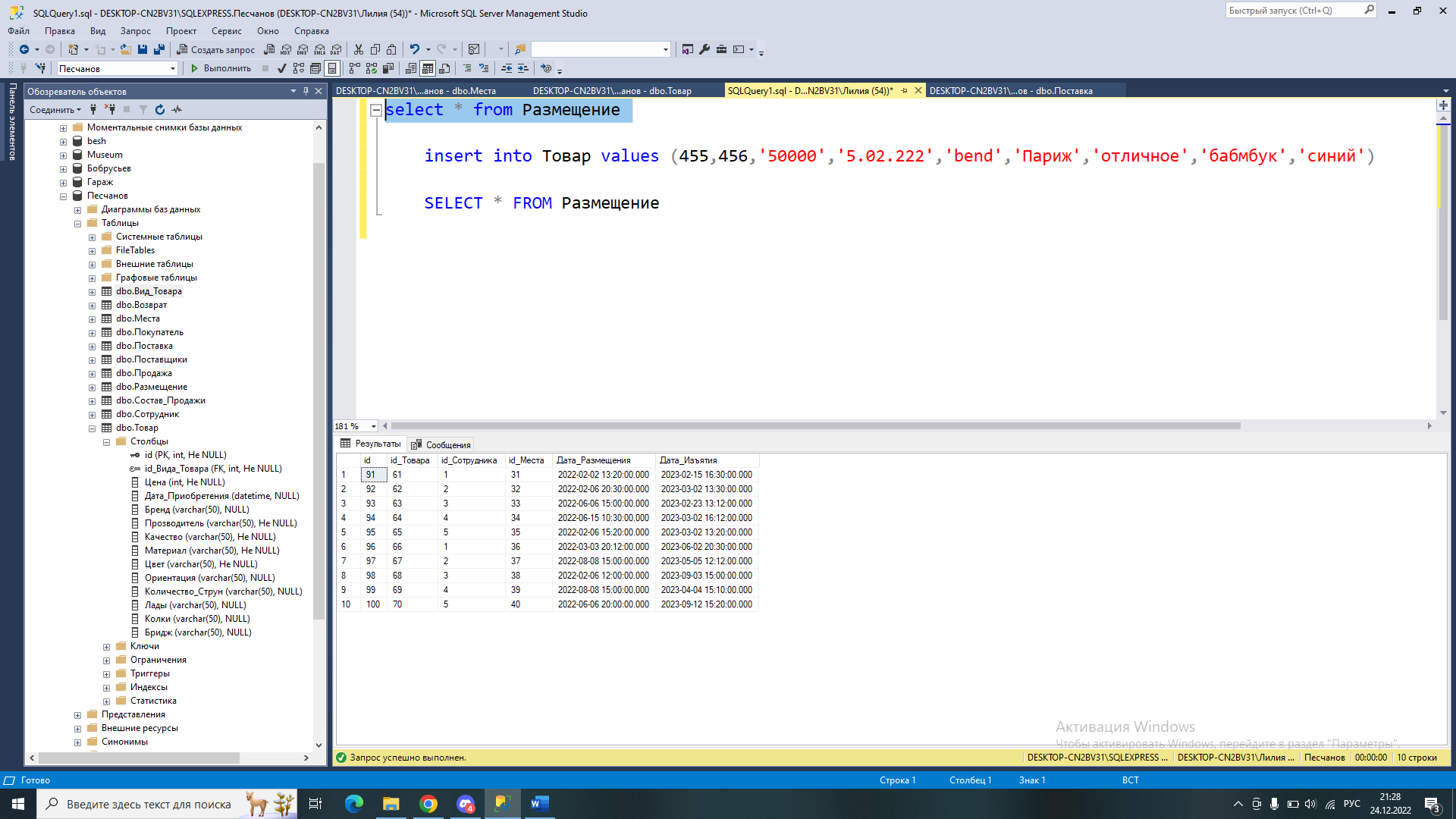
GO

select \* from Размещение

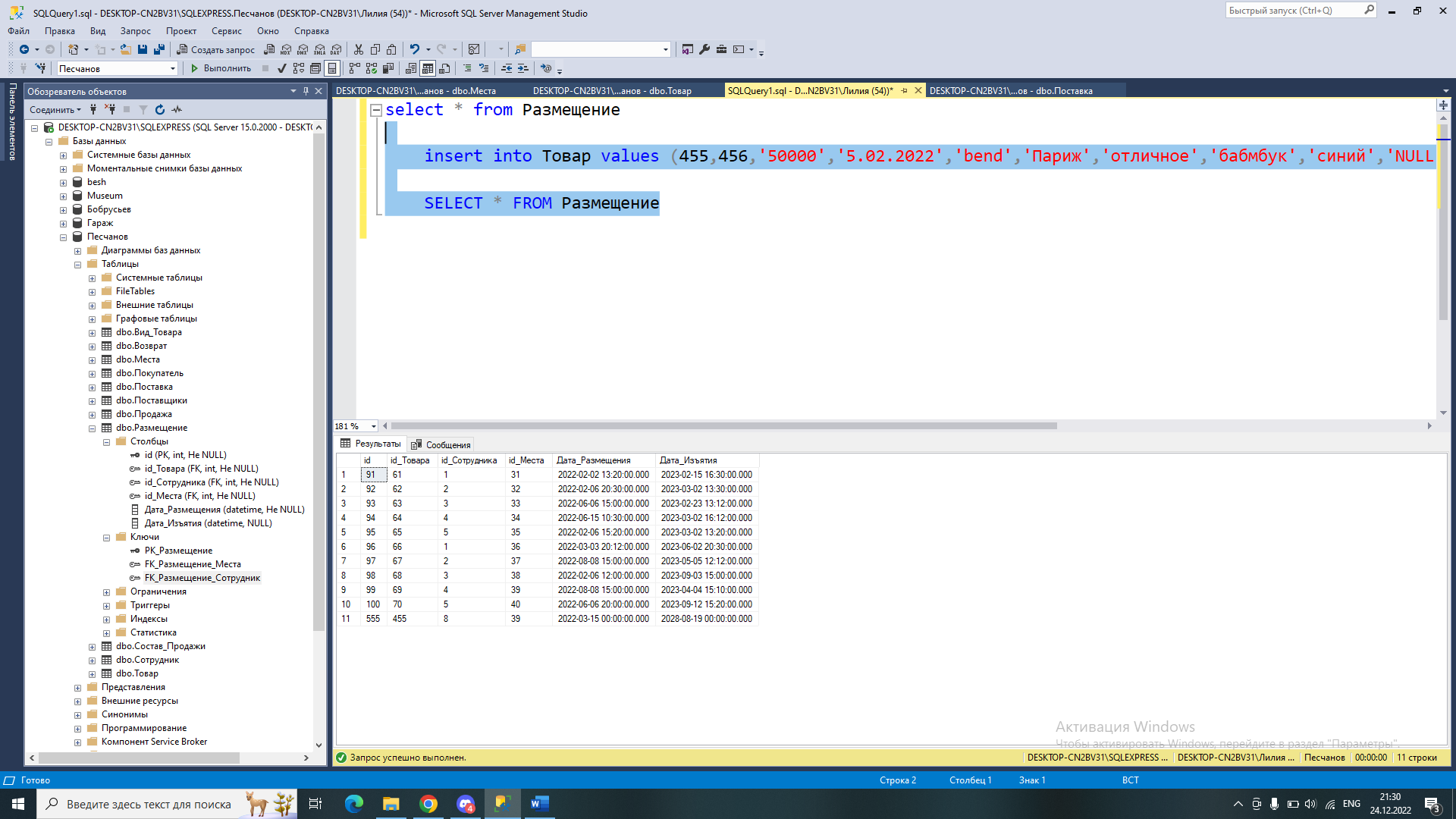
insert into Товар values (455,456,'50000','5.02.222','bend','Париж','отличное','бабмбук','синий')

SELECT \* FROM Размещение

РАЗМЕЩЕНИЕ **ДО**



РАЗМЕЩЕНИЕ **ПОСЛЕ**



Триггер 6.

Назначение: при добавлении поставщика добавляется поставка

CREATE TRIGGER При\_добавлении\_поставщика

ON поставщики

INSTEAD OF INSERT

AS

DECLARE @ID INT

SELECT @ID = (SELECT id FROM inserted)

insert into Поставка values(132,@ID,6,70,'15.09.2022','49000')

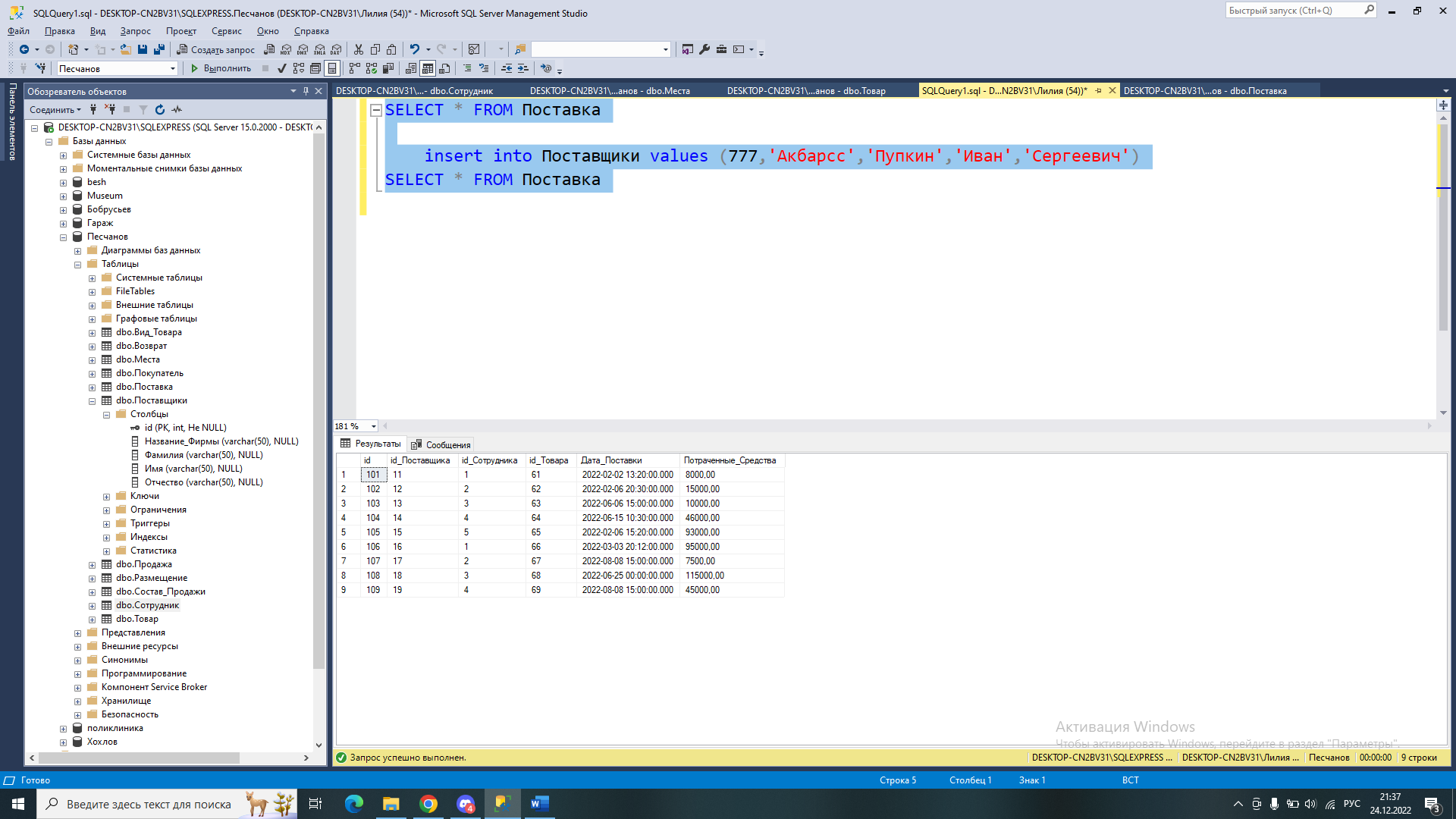
GO

SELECT \* FROM Поставка

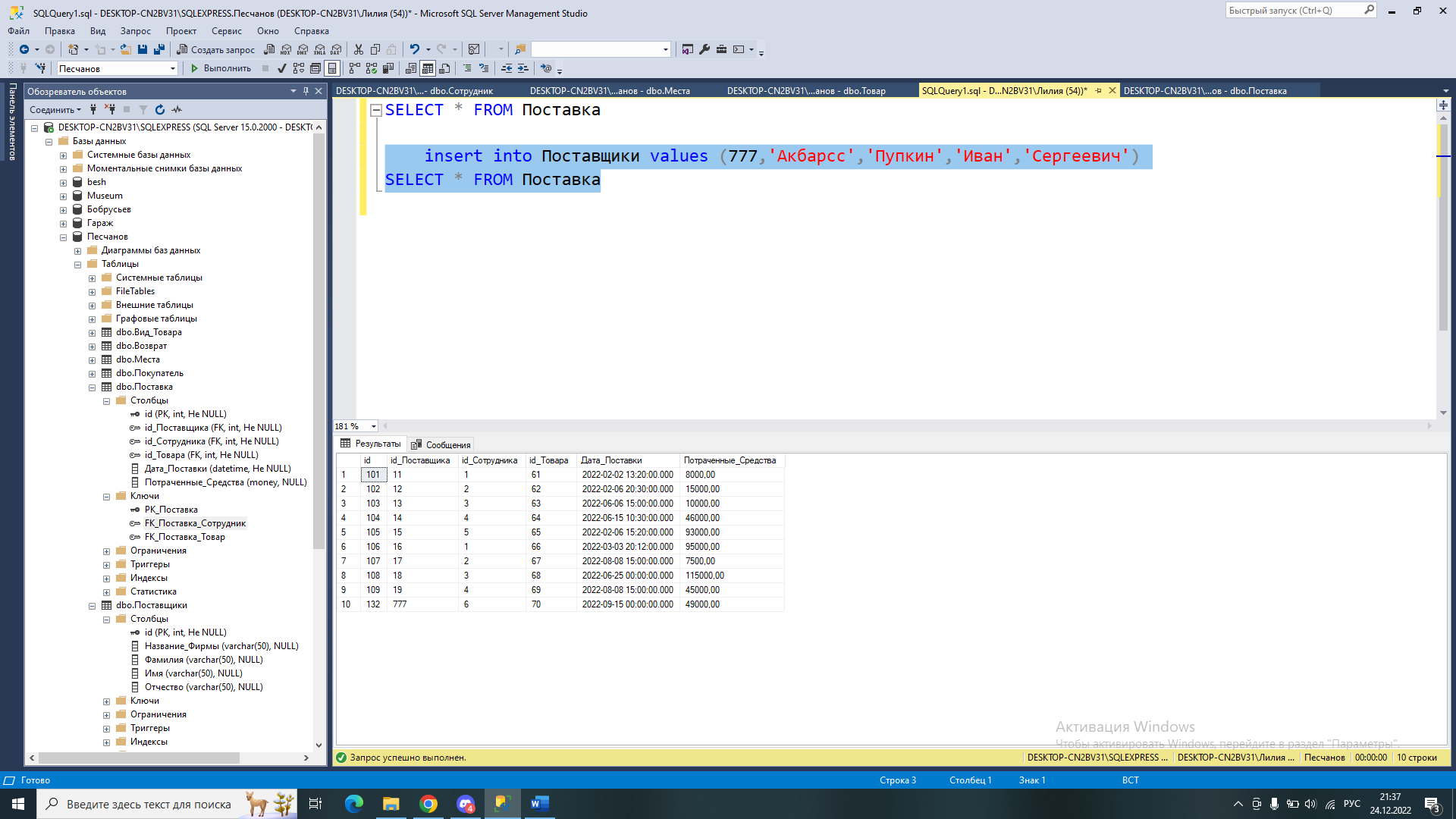
insert into Поставщики values (777,'Акбарсс','Пупкин','Иван','Сергеевич')

SELECT \* FROM Поставка

ПОСТАВКА **ДО**



ПОСТАВКА **ПОСЛЕ**



Триггер 7.

Назначение: при добавлении покупателя добавляется продажа

CREATE TRIGGER При\_добавлении\_покупателя

ON покупатель

AFTER INSERT

AS

DECLARE @ID INT

SELECT @ID = (SELECT id FROM inserted)

insert into Продажа values (899,2,@ID,'25.06.2023','NULL')

GO

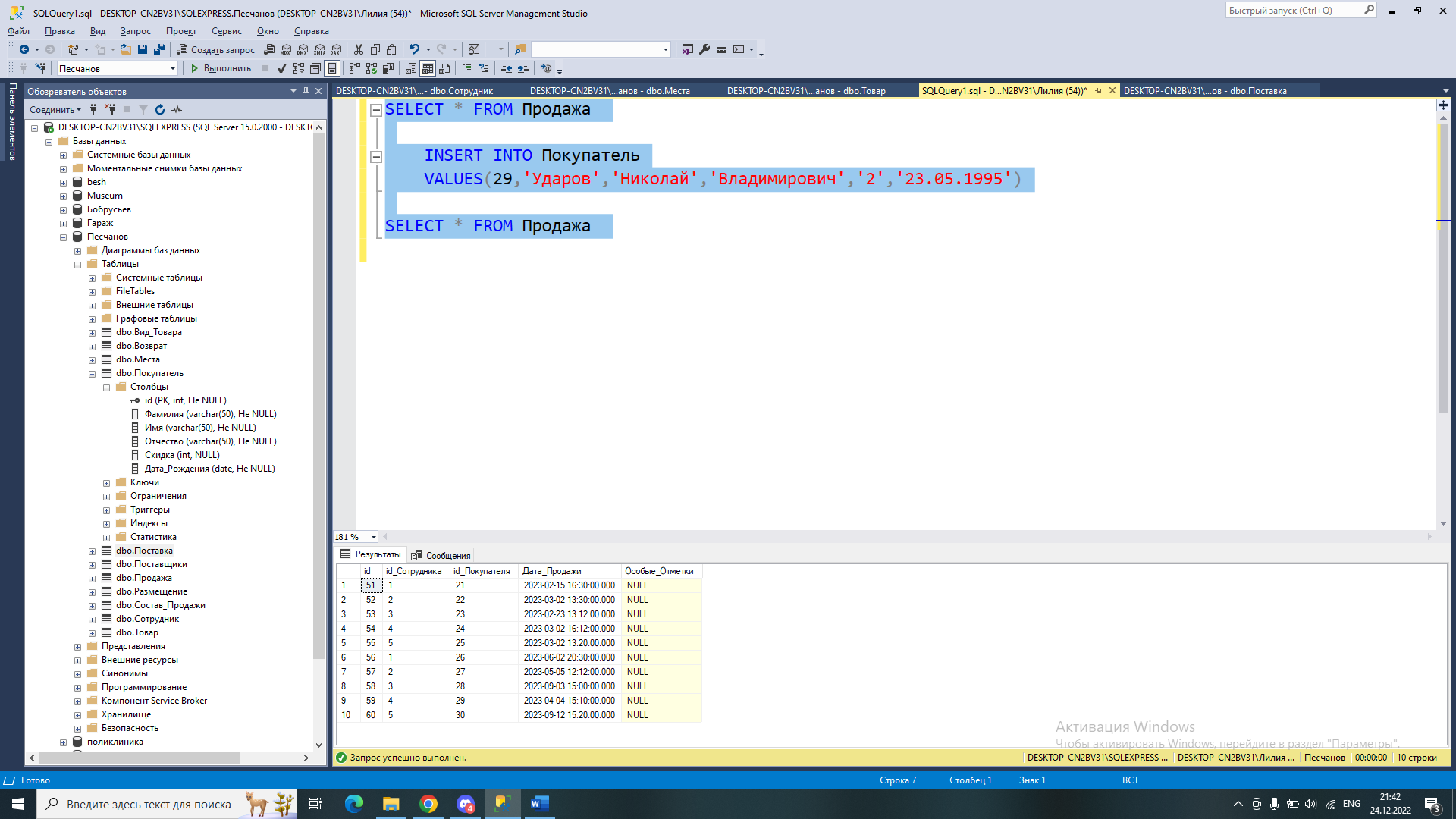
SELECT \* FROM Продажа

INSERT INTO Покупатель

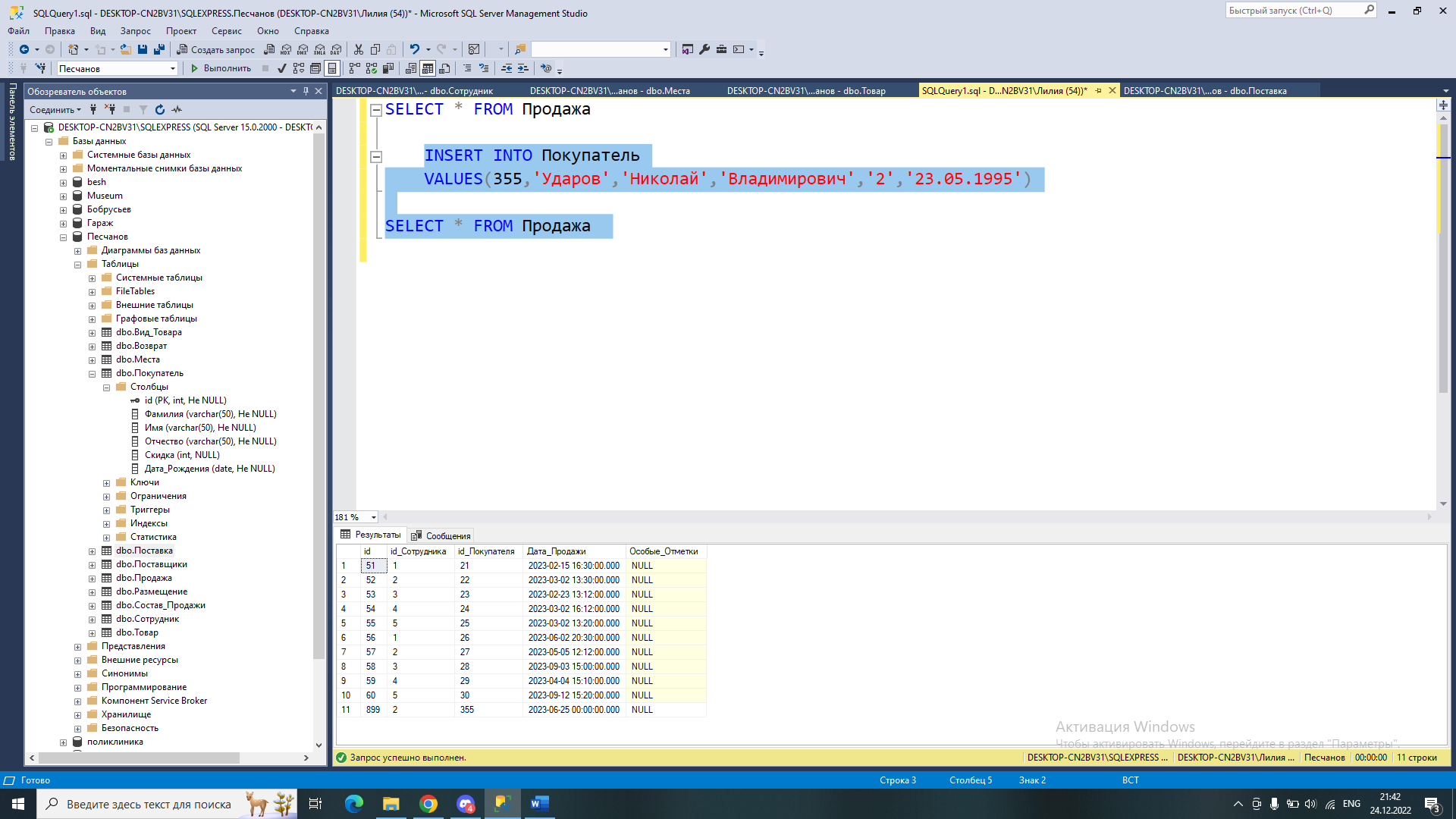
VALUES(355,'Ударов','Николай','Владимирович','2','23.05.1995')

SELECT \* FROM Продажа

ПРОДАЖА ДО:



ПРОДАЖА ПОСЛЕ:



Триггер 8.

Назначение: при удалении поставщика удаляется поставка

CREATE TRIGGER При\_удалении\_поставщика

ON поставщики

INSTEAD OF DELETE

AS

DECLARE @ID INT

SELECT @ID = (SELECT [ID] FROM deleted)

delete from Поставка where id\_Поставщика = @ID

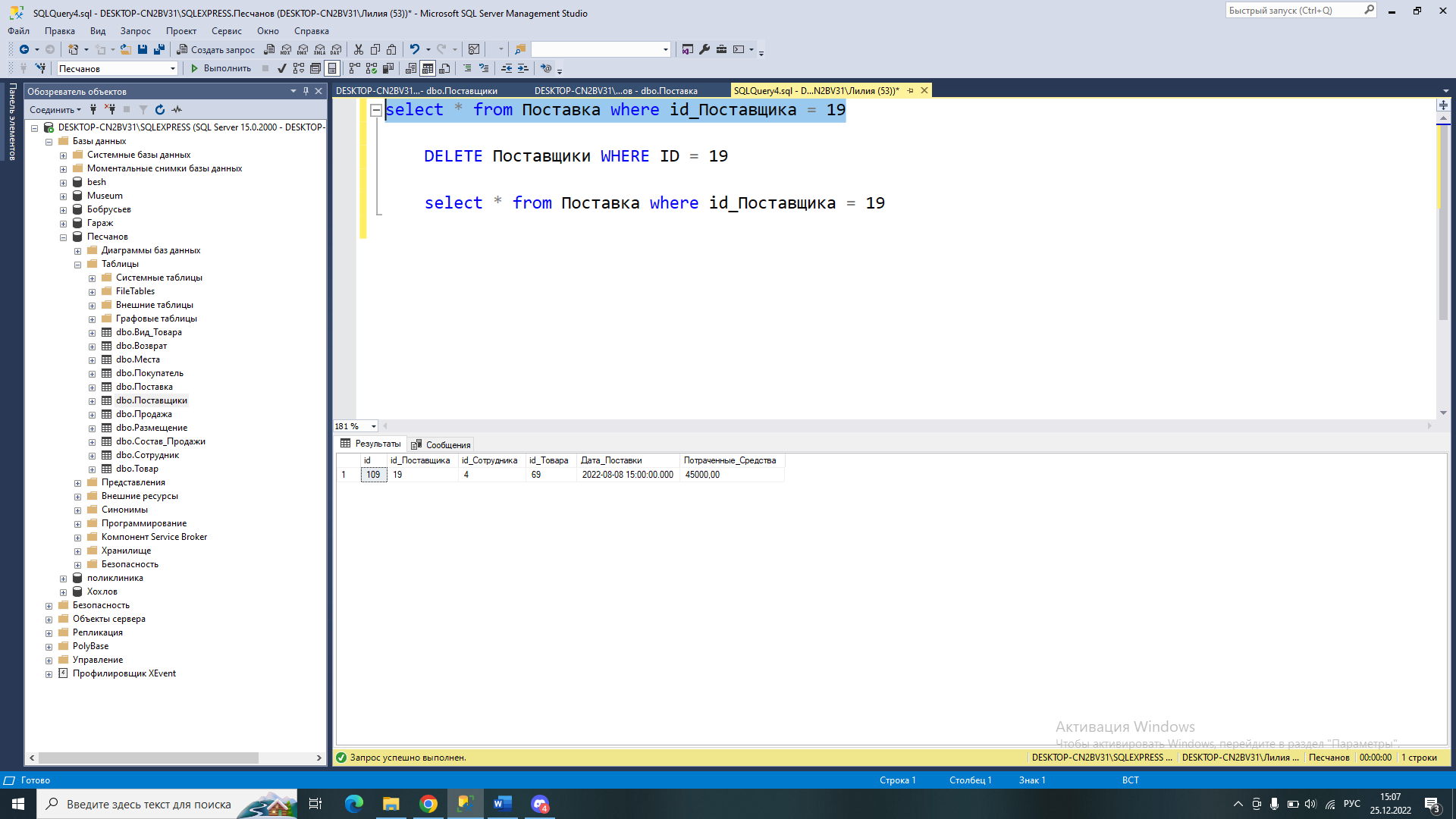
GO

select \* from Поставка where id\_Поставщика = 19

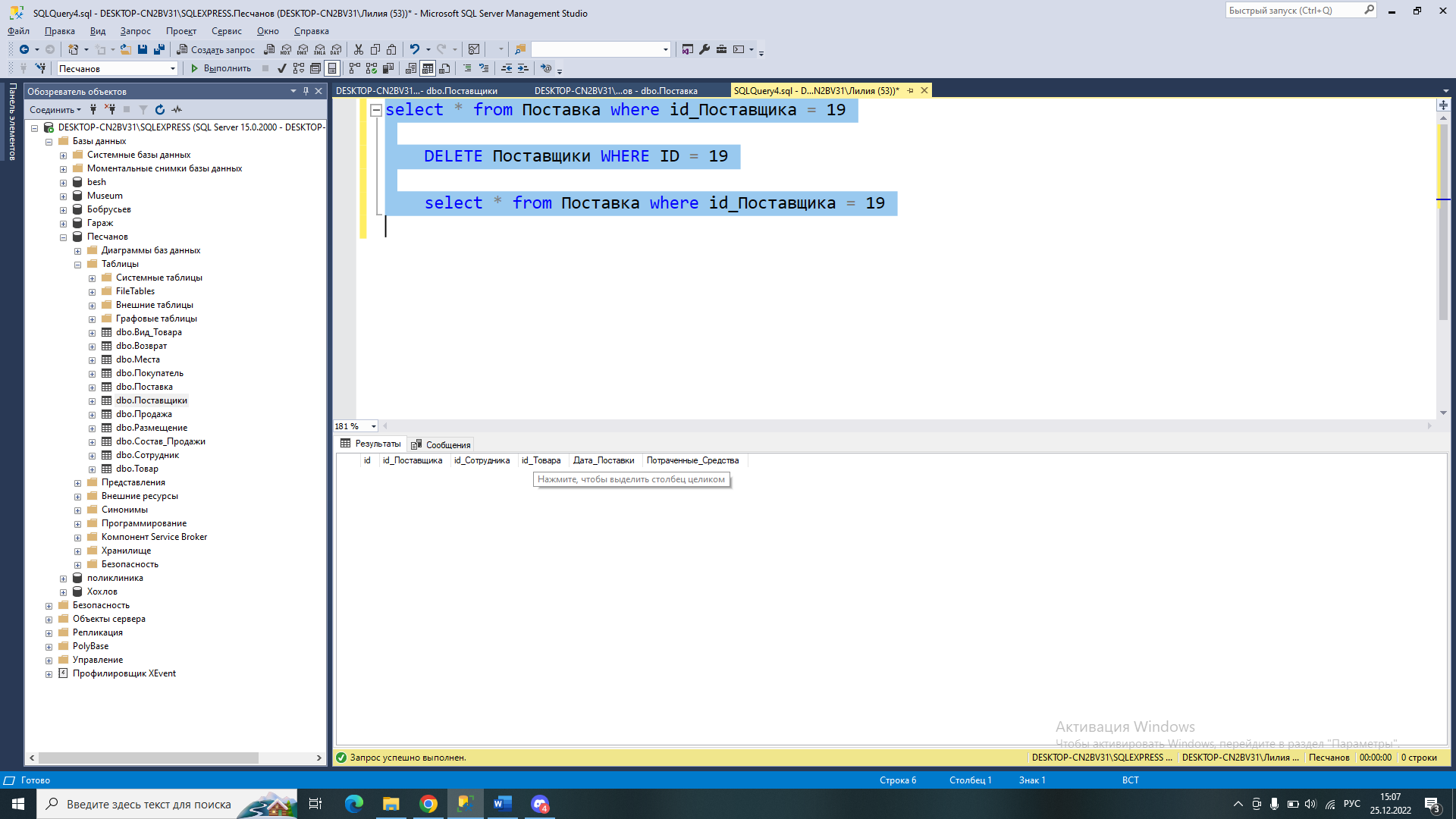
DELETE Поставщики WHERE ID = 19

select \* from Поставка where id\_Поставщика = 19

ПОСТАВКА ДО



ПОСТАВКА ПОСЛЕ



Триггер 9.

Назначение: при удалении товара удаляется вид товара

CREATE TRIGGER При\_удалении\_товара

ON товар

AFTER DELETE

AS

DECLARE @id INT

SELECT @id = (SELECT id\_Вида\_Товара FROM deleted)

delete from Вид\_Товара where id = @id

GO

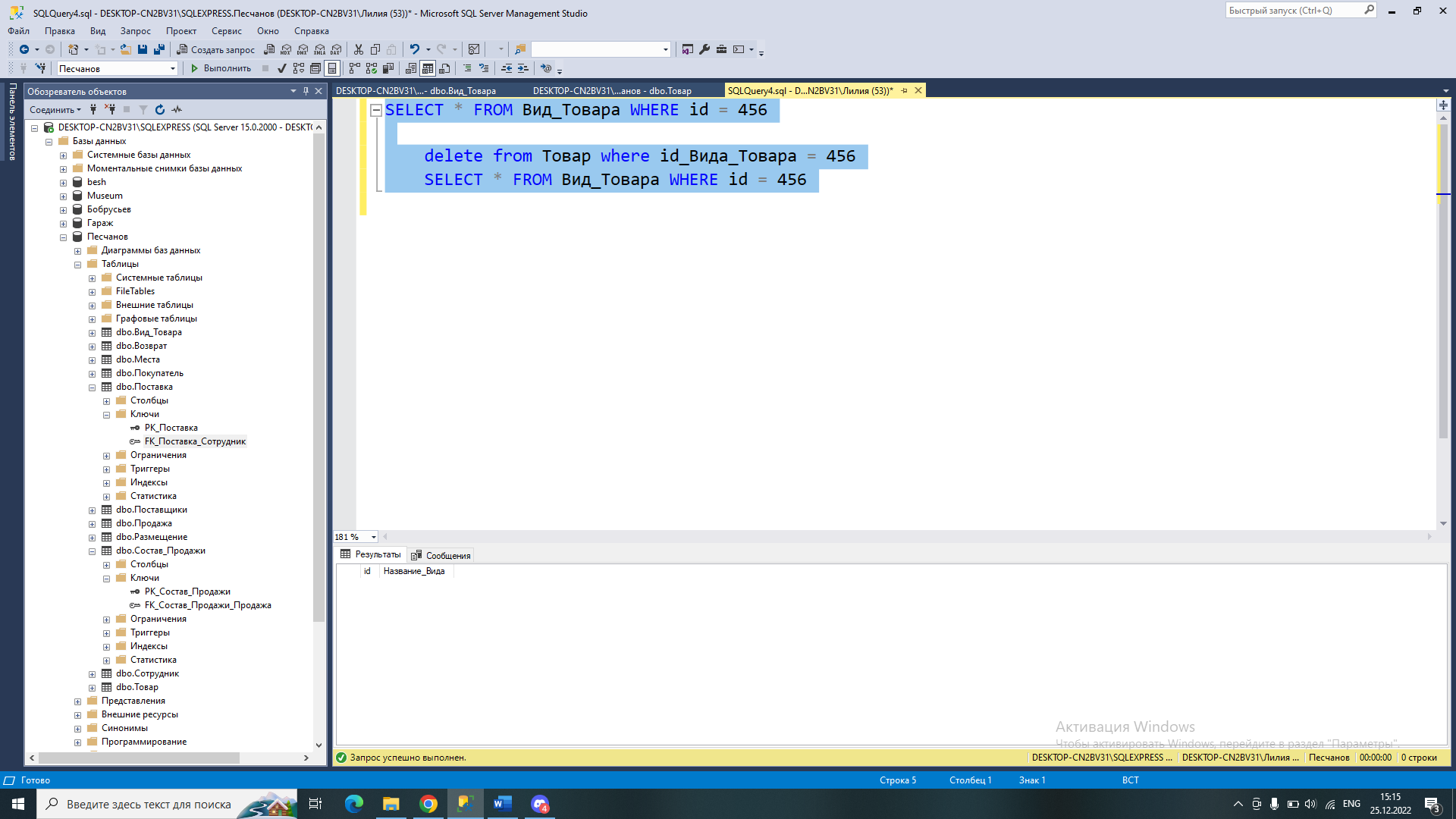
SELECT \* FROM Вид\_Товара WHERE id = 456

delete from Товар where id\_Вида\_Товара = 456

SELECT \* FROM Вид\_Товара WHERE id = 456

ВИД ТОВАРА **ДО И ПОСЛЕ**:





Триггер 10.

Назначение: при удалении места удаляется размещение

CREATE TRIGGER При\_удалении\_места

ON места

FOR DELETE

AS

DECLARE @ID INT

SELECT @ID = (SELECT [ID] FROM deleted)

delete from Размещение where id\_Места = @ID

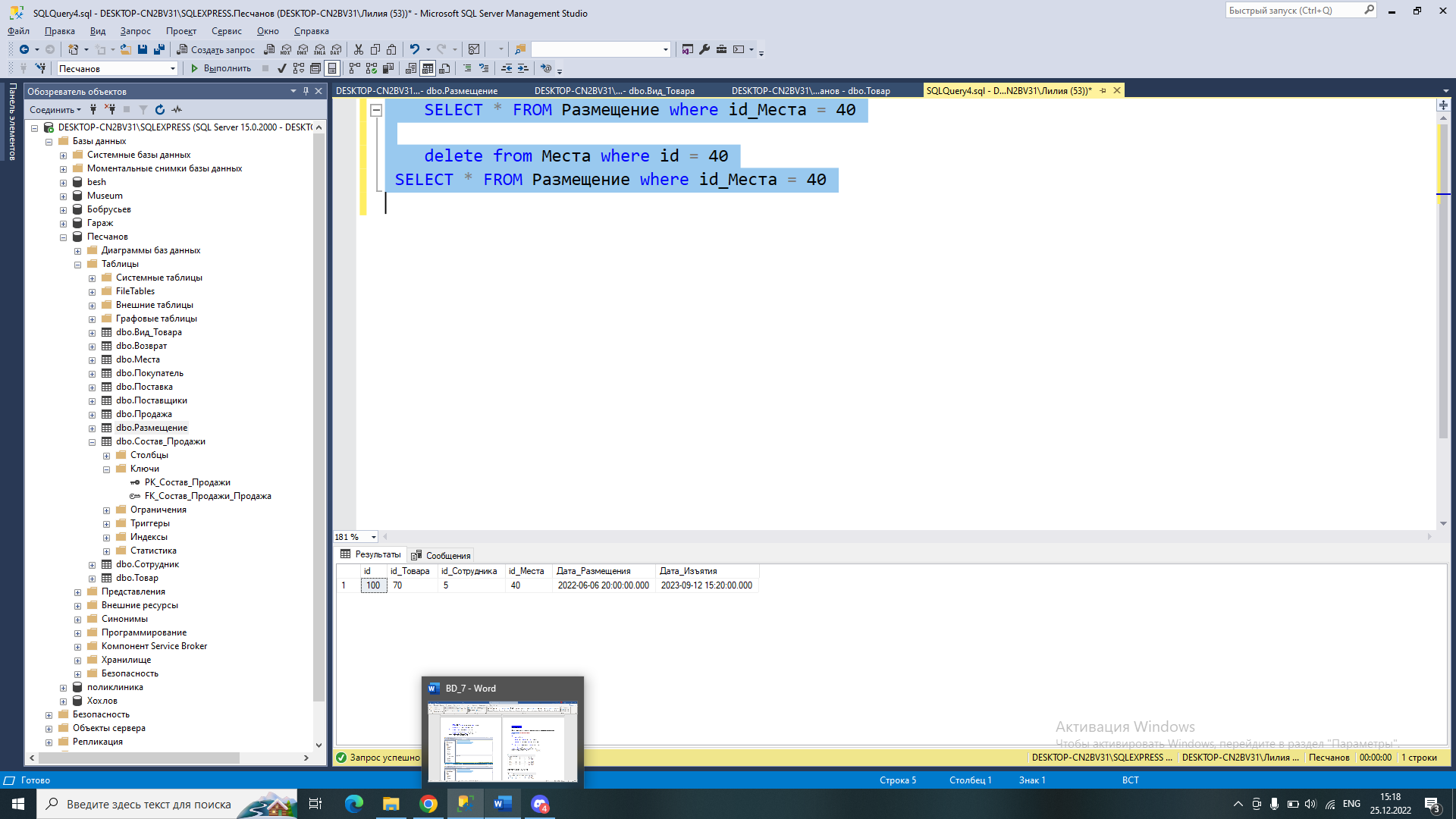
GO

SELECT \* FROM Размещение where id\_Места = 40

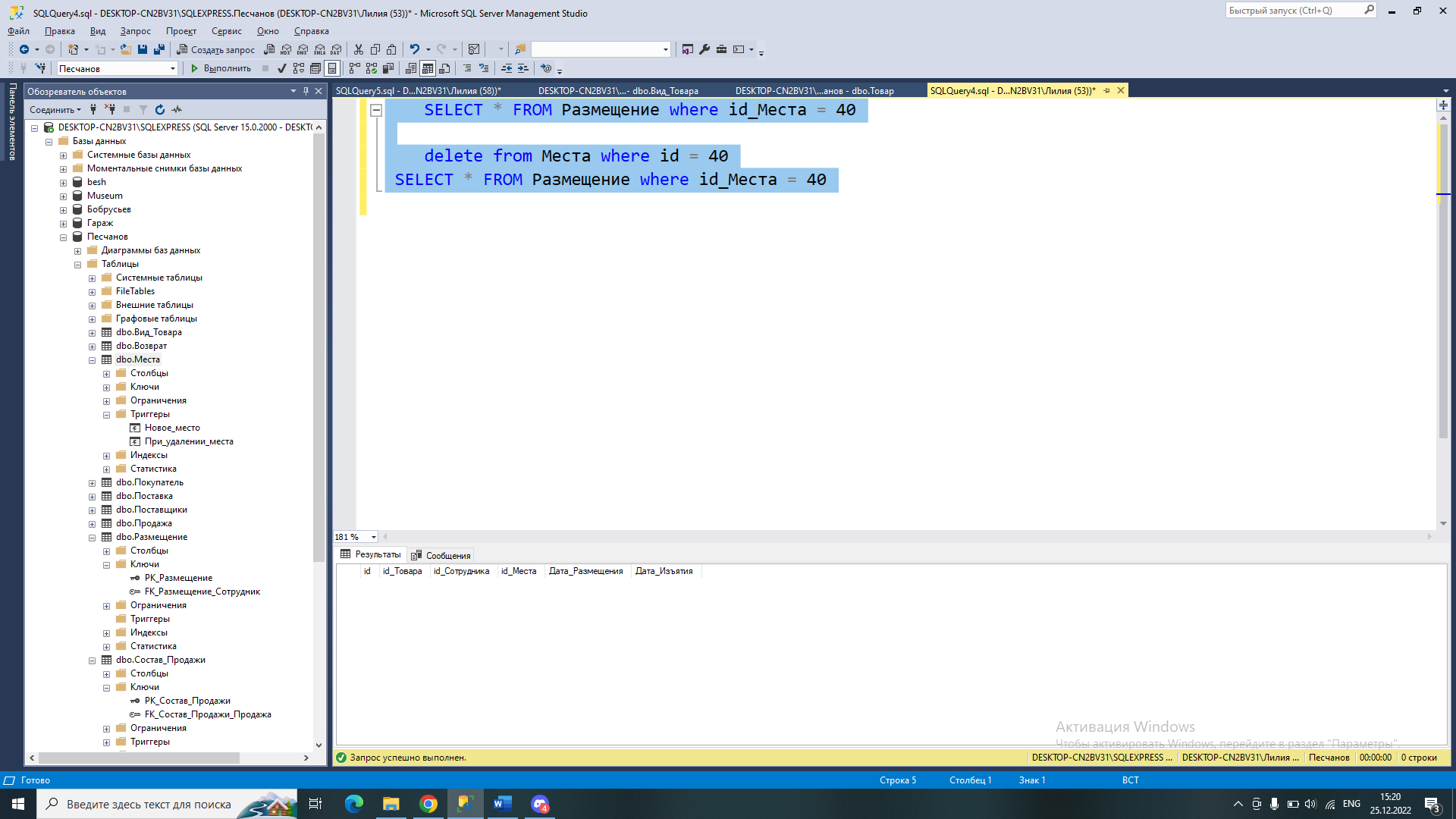
delete from Места where id = 40

SELECT \* FROM Размещение where id\_Места = 40

РАЗМЕЩЕНИЕ ДО



РАЗМЕЩЕНИЕ ПОСЛЕ



Триггер 11.

Назначение: при удалении покупателя удаляется продажа

CREATE TRIGGER При\_удалении\_покупателя

ON покупатель

AFTER DELETE

AS

DECLARE @ID INT

SELECT @ID = (SELECT id FROM deleted)

delete from Продажа where id\_Покупателя = @ID

GO

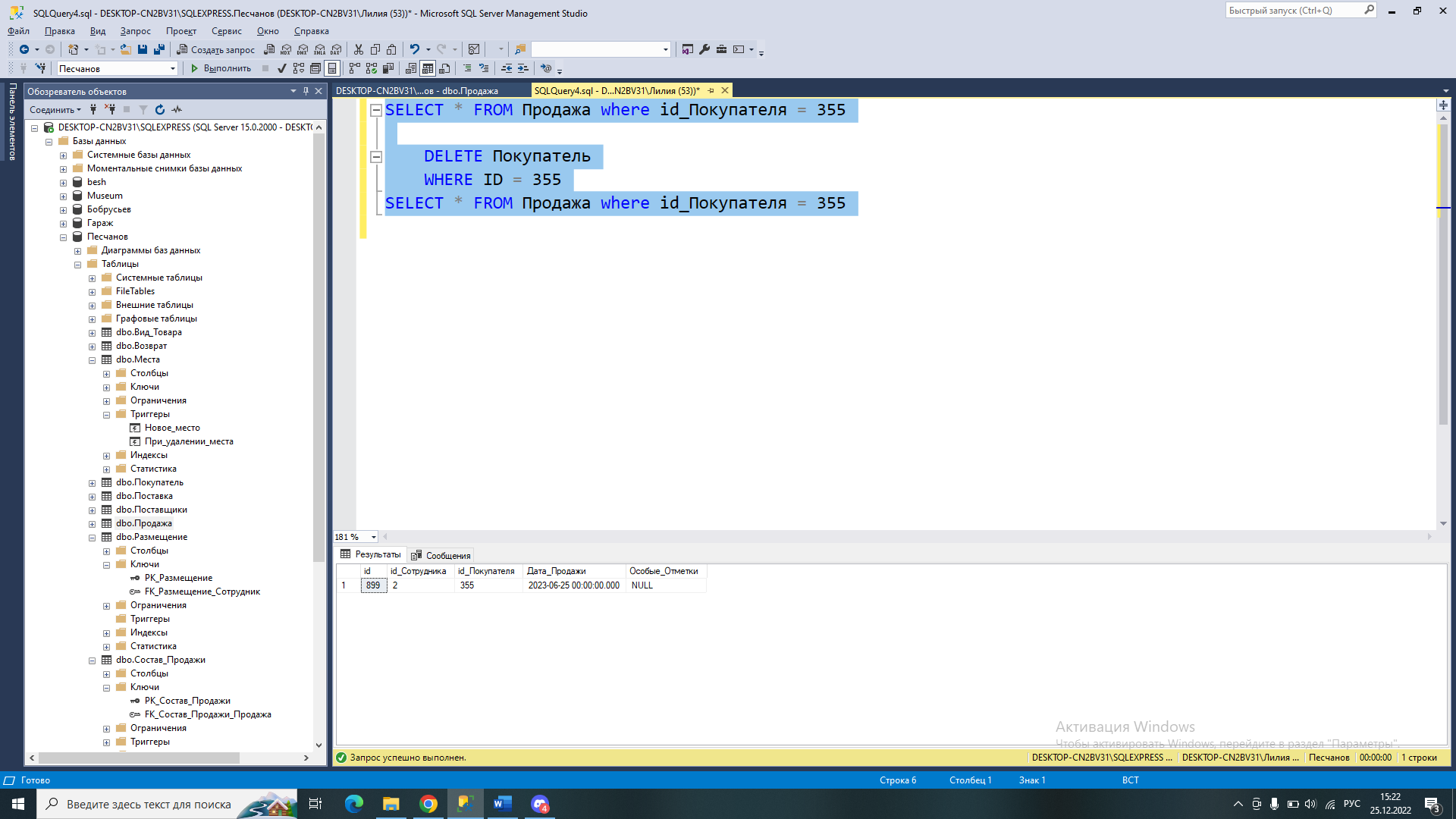
SELECT \* FROM Продажа where id\_Покупателя = 355

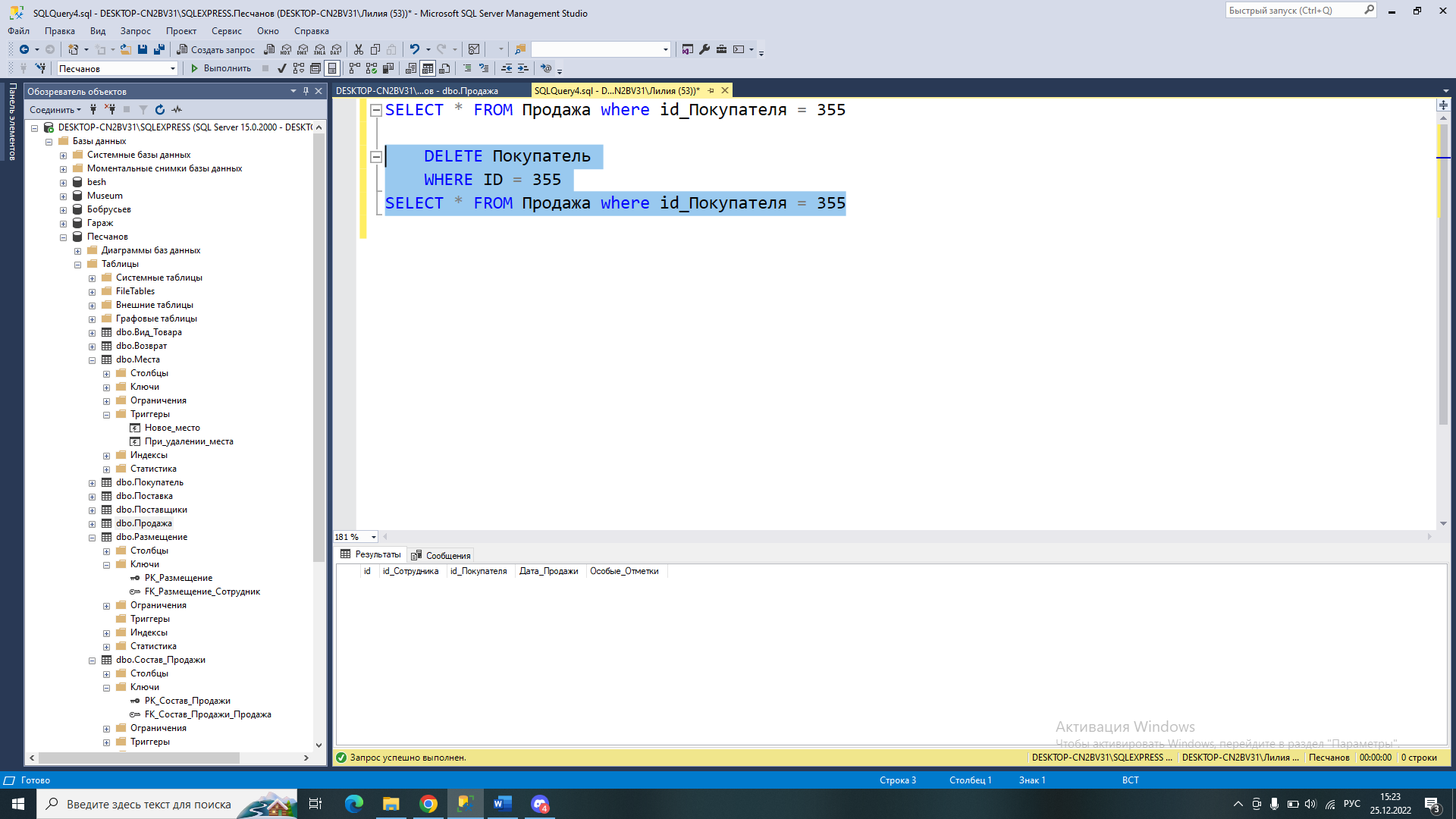
DELETE Покупатель

WHERE ID = 355

SELECT \* FROM Продажа where id\_Покупателя = 355

ПРОДАЖА ДО И ПОСЛЕ





Функция 1.

Назначение: название товара, которая меньше средней цены

CREATE FUNCTION МеньшеСрЦены (@data date)

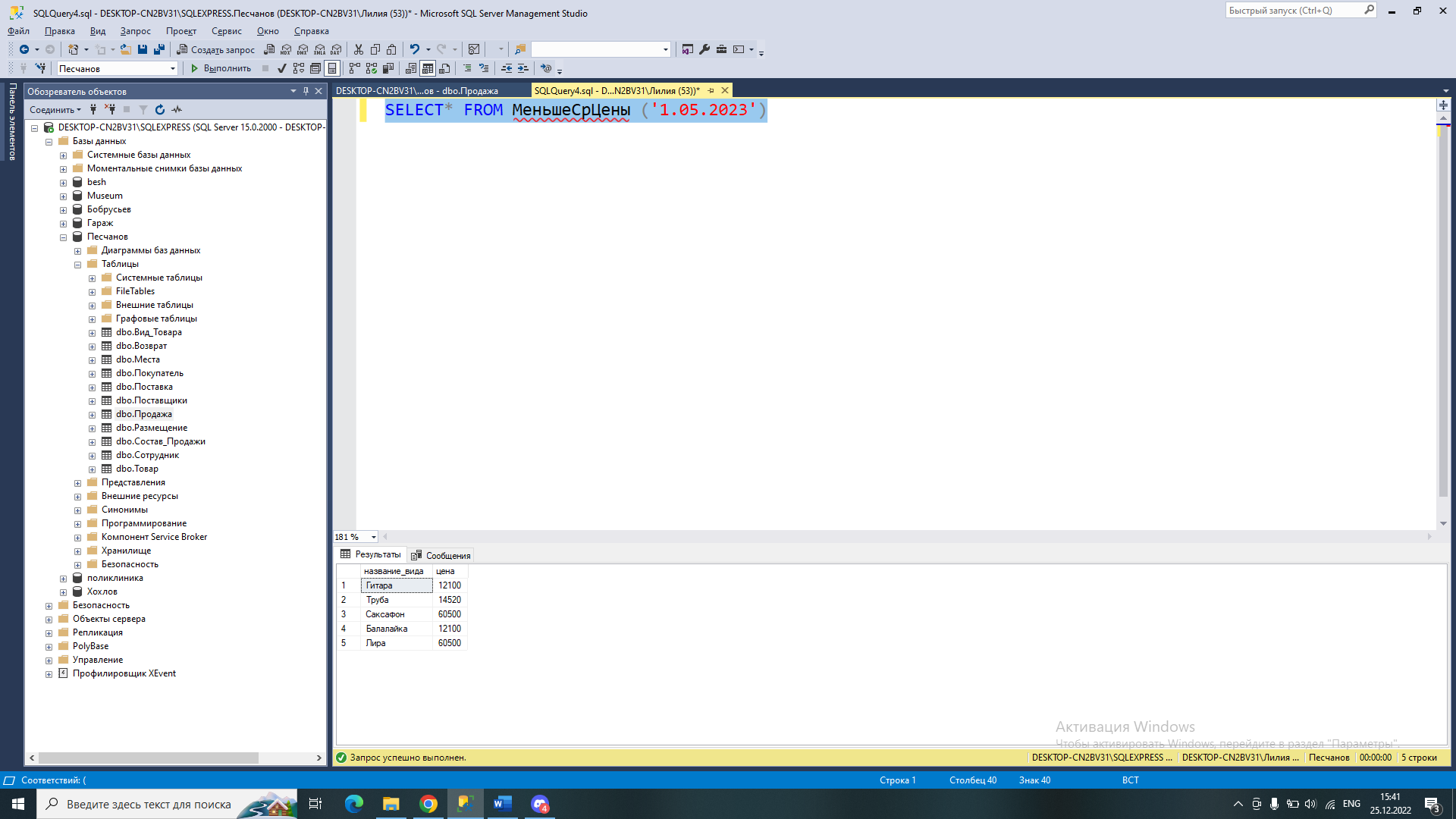
RETURNS TABLE

AS RETURN (SELECT название\_вида, цена

FROM Товар join Вид\_Товара on Вид\_Товара.id = Товар.id\_Вида\_Товара

WHERE Цена <= (SELECT avg(Цена) FROM Товар )and Дата\_Приобретения < @data)

SELECT\* FROM МеньшеСрЦены ('1.05.2023')



Функция 2.

Назначение: узнать название товара , на которое потратили больше 50000 тысяч

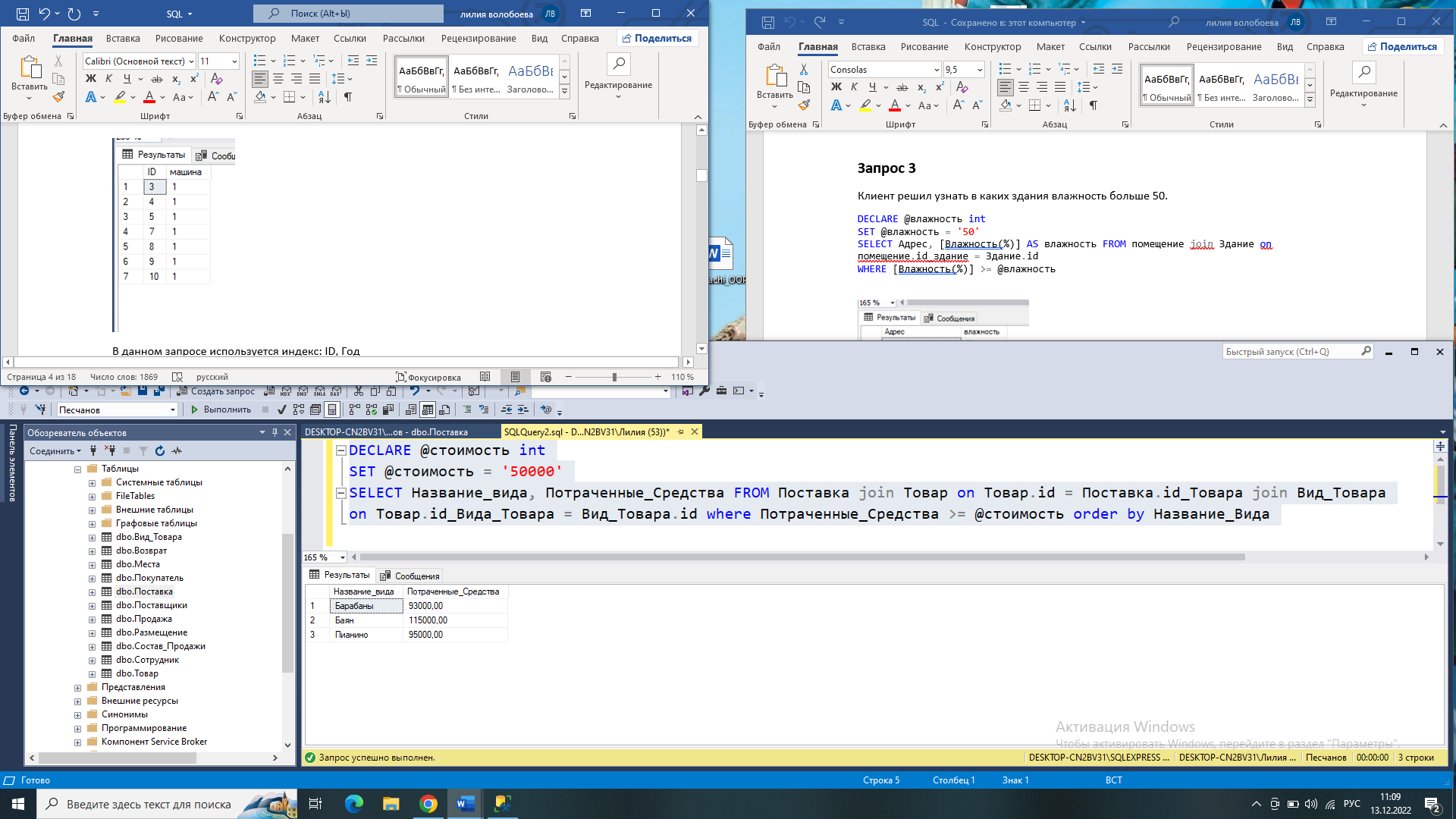
CREATE FUNCTION Потратилибольше50000тысяч (@стоимость int)

RETURNS TABLE

AS RETURN (SELECT Название\_вида, Потраченные\_Средства FROM Поставка join Товар on Товар.id = Поставка.id\_Товара join Вид\_Товара

on Товар.id\_Вида\_Товара = Вид\_Товара.id where Потраченные\_Средства >= @стоимость)

SELECT\* FROM Потратилибольше50000тысяч ('50000')



Функция 3.

Назначение: вывести вид товара, бренд, материал и цвет товара

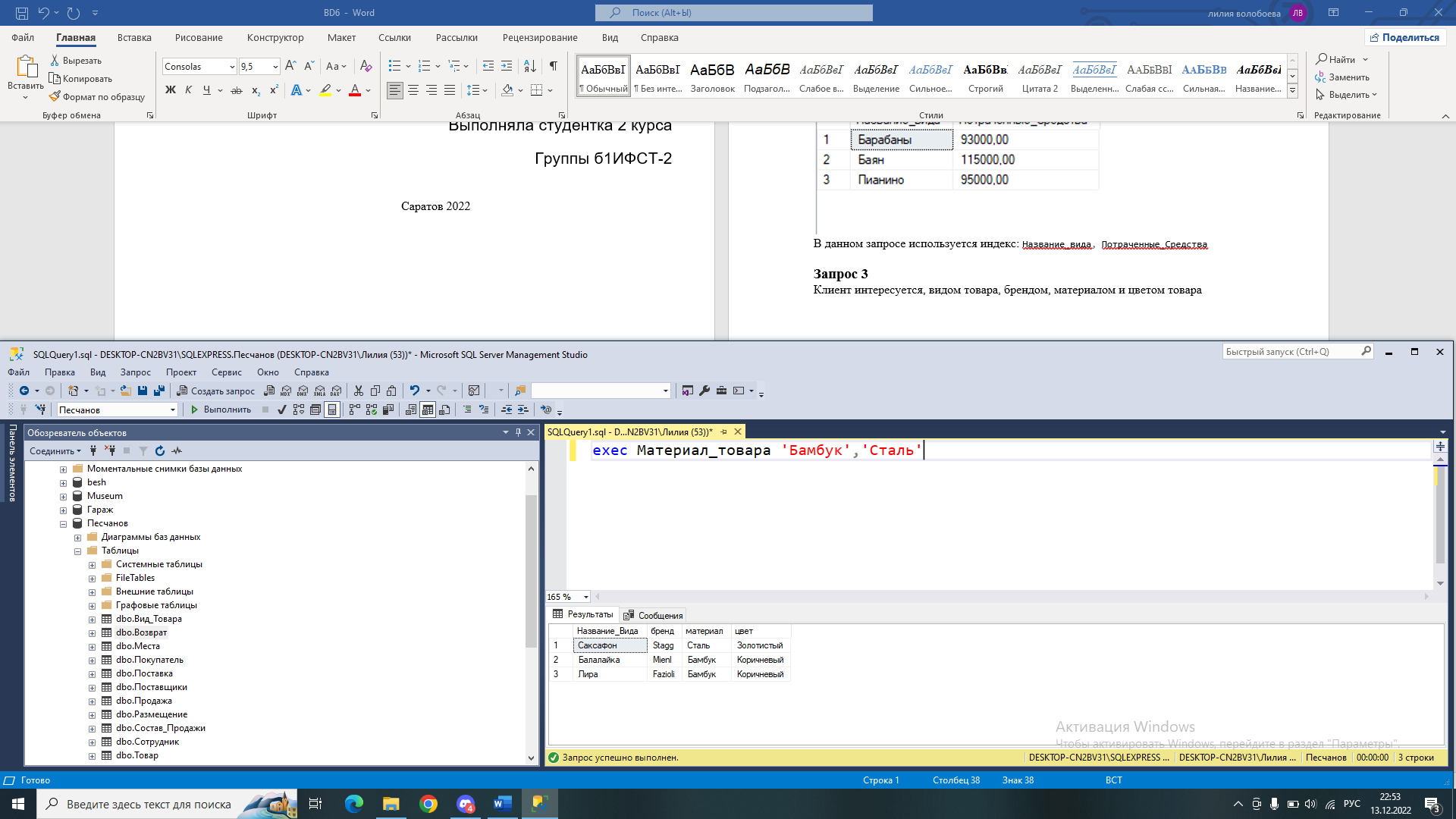
CREATE FUNCTION ВидБрендМатериалЦвет (@материал varchar(50), @материалл varchar(50)

RETURNS TABLE

AS RETURN ( select Название\_Вида, бренд, материал, цвет from Вид\_Товара join Товар on Товар.id\_Вида\_Товара = Вид\_Товара.id

where Материал = @материал or Материал =@материалл)

SELECT\* FROM ВидБрендМатериалЦвет ('Бамбук','Сталь')



Функция 4.

Назначение: Общая стоимость от покупки

CREATE FUNCTION ОбщаяСтоимость (@id1 int,@id2 int)

RETURNS MONEY

AS

BEGIN

DECLARE @цена money

SELECT @цена = (select SUM(цена) AS общая\_стоимость FROM Состав\_Продажи

where id between @id1 and @id2)

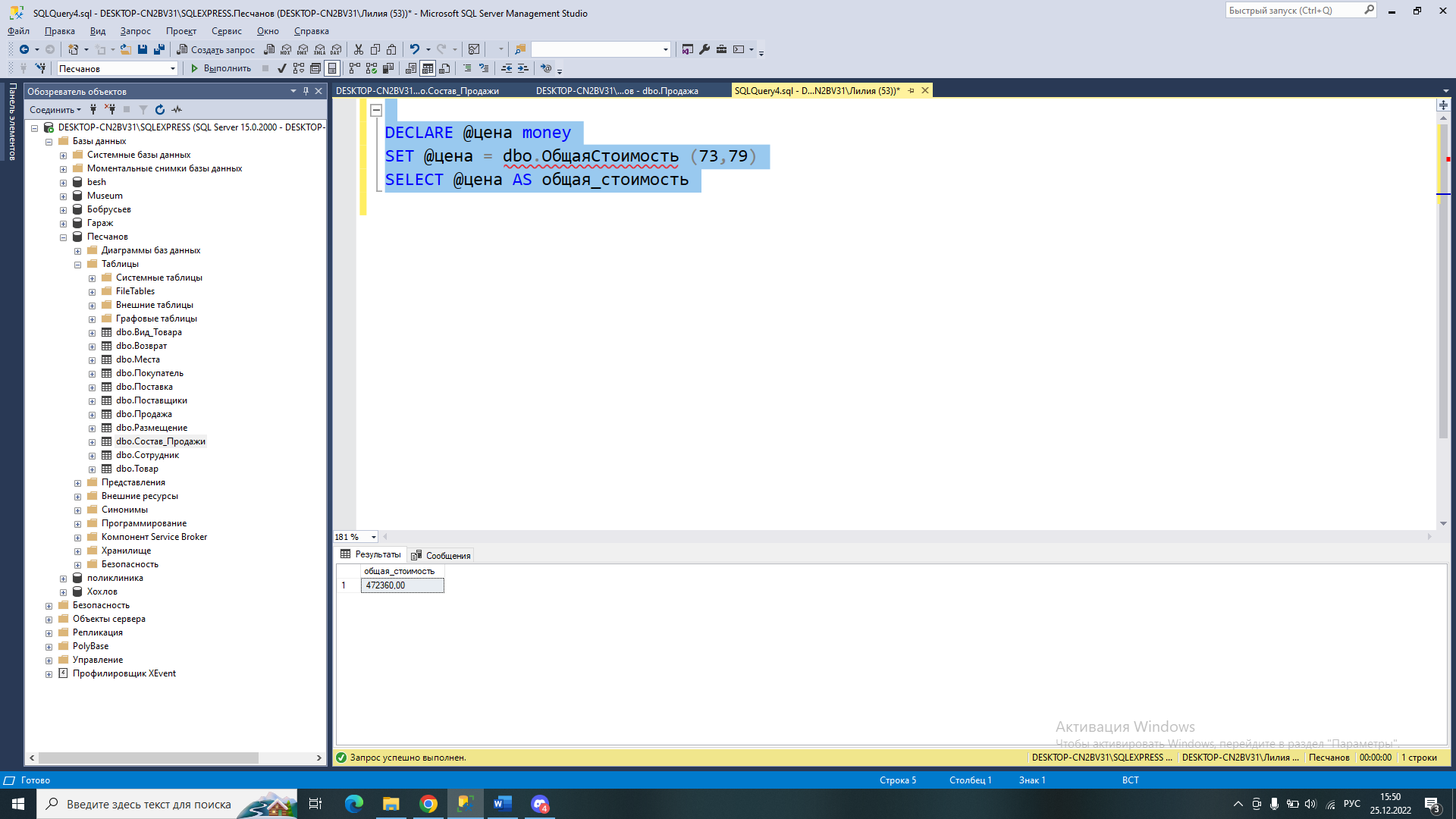
return @цена

END

DECLARE @цена money

SET @цена = dbo.ОбщаяСтоимость (73,79)

SELECT @цена AS общая\_стоимость



Функция 5.

Назначение: Вывести дешевый товар

CREATE FUNCTION ДешевыйТовар (@цена int )

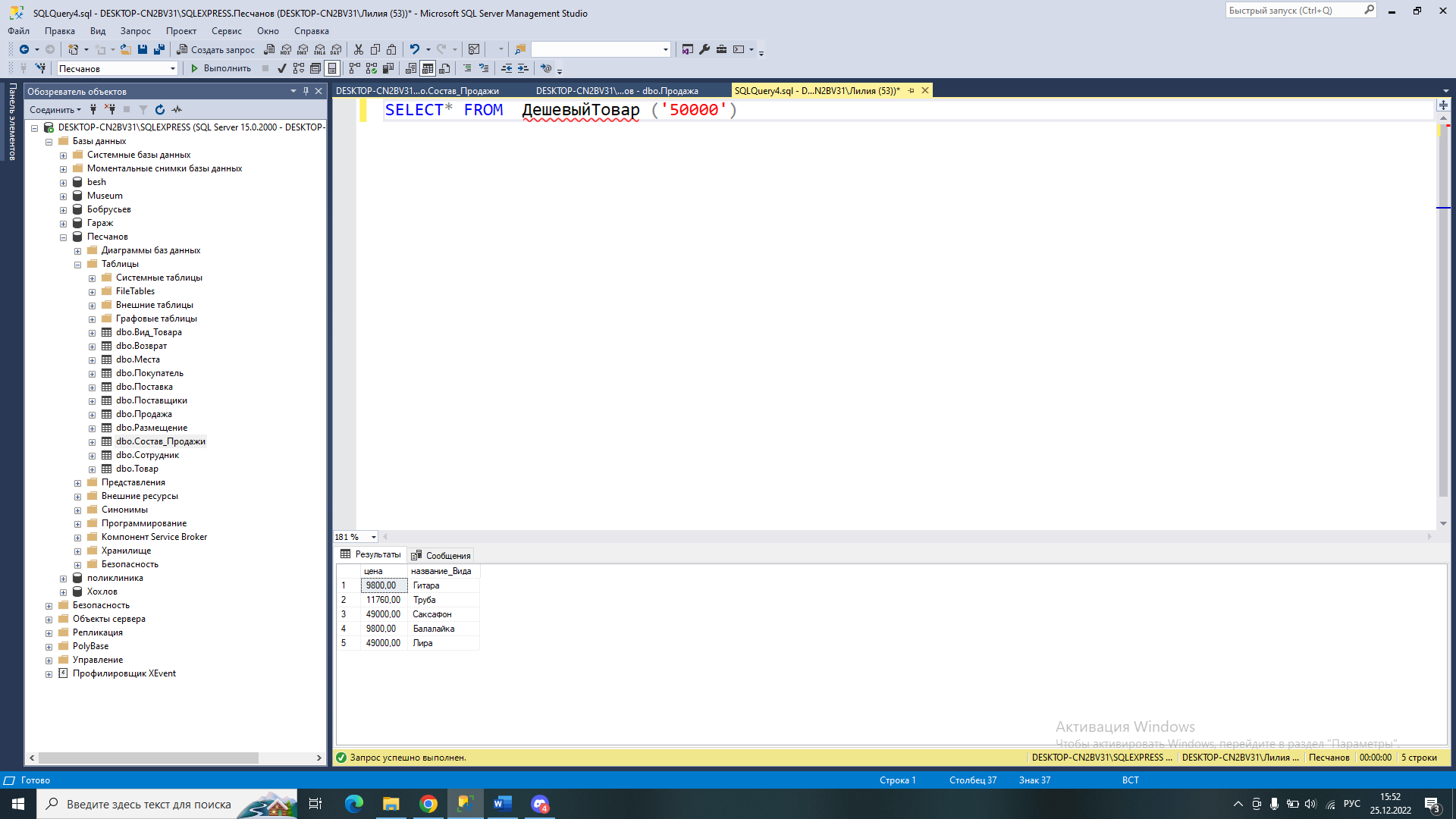
RETURNS TABLE

AS RETURN ( select Состав\_Продажи.цена,название\_Вида

from Состав\_Продажи join Товар on Товар.id = Состав\_Продажи.id\_Товара join Вид\_Товара on Вид\_Товара.id = Товар.id\_Вида\_Товара

where Состав\_Продажи.Цена <=@цена )

SELECT\* FROM ДешевыйТовар ('50000')



Функция 6.

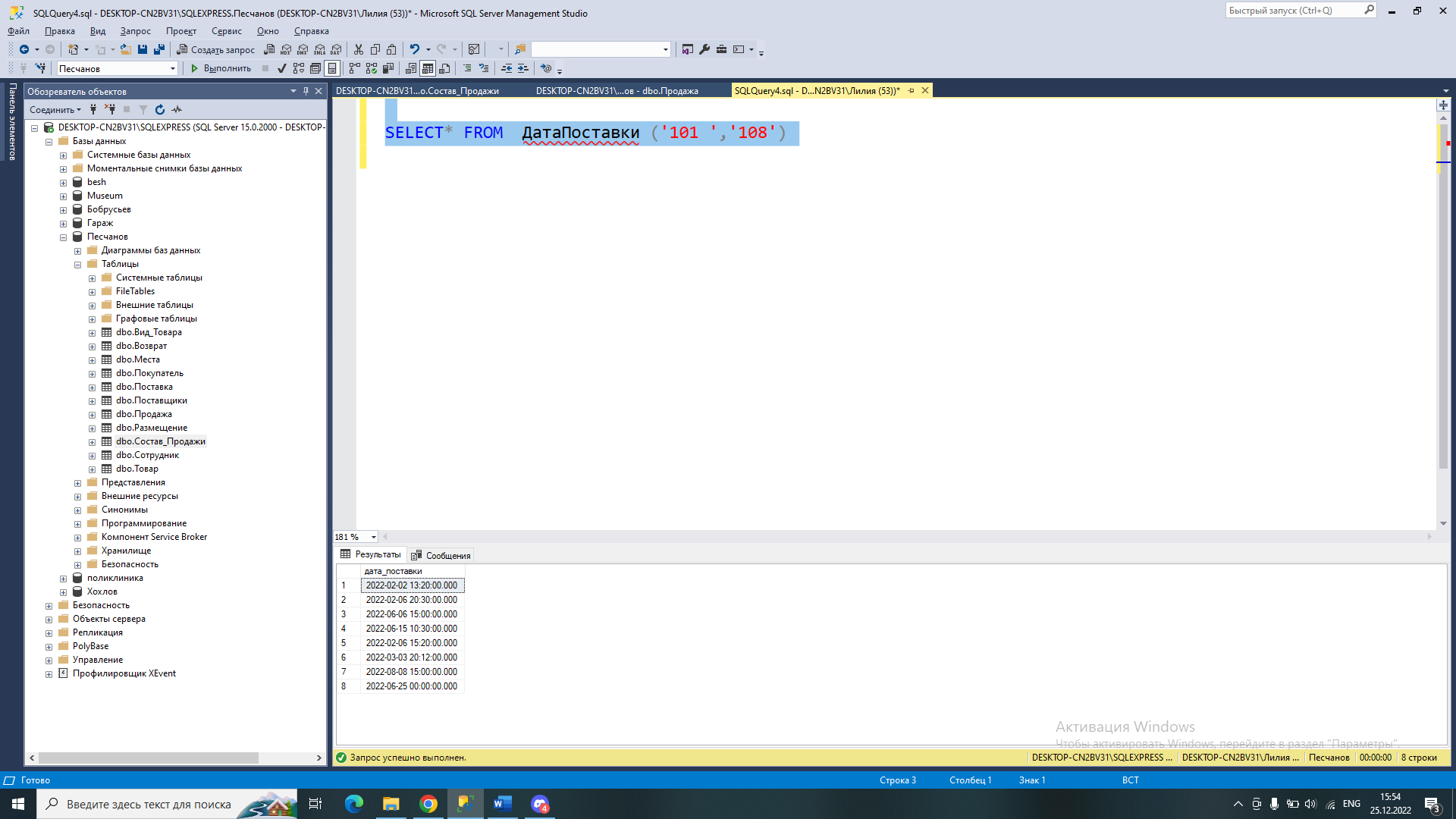
Назначение: узнать дату поставки

CREATE FUNCTION ДатаПоставки ( @id1 int,@id2 int)

RETURNS TABLE

AS RETURN ( select дата\_поставки from поставка where id between @id1 and @id2)

SELECT\* FROM ДатаПоставки ('101 ','108')



Функция 7.

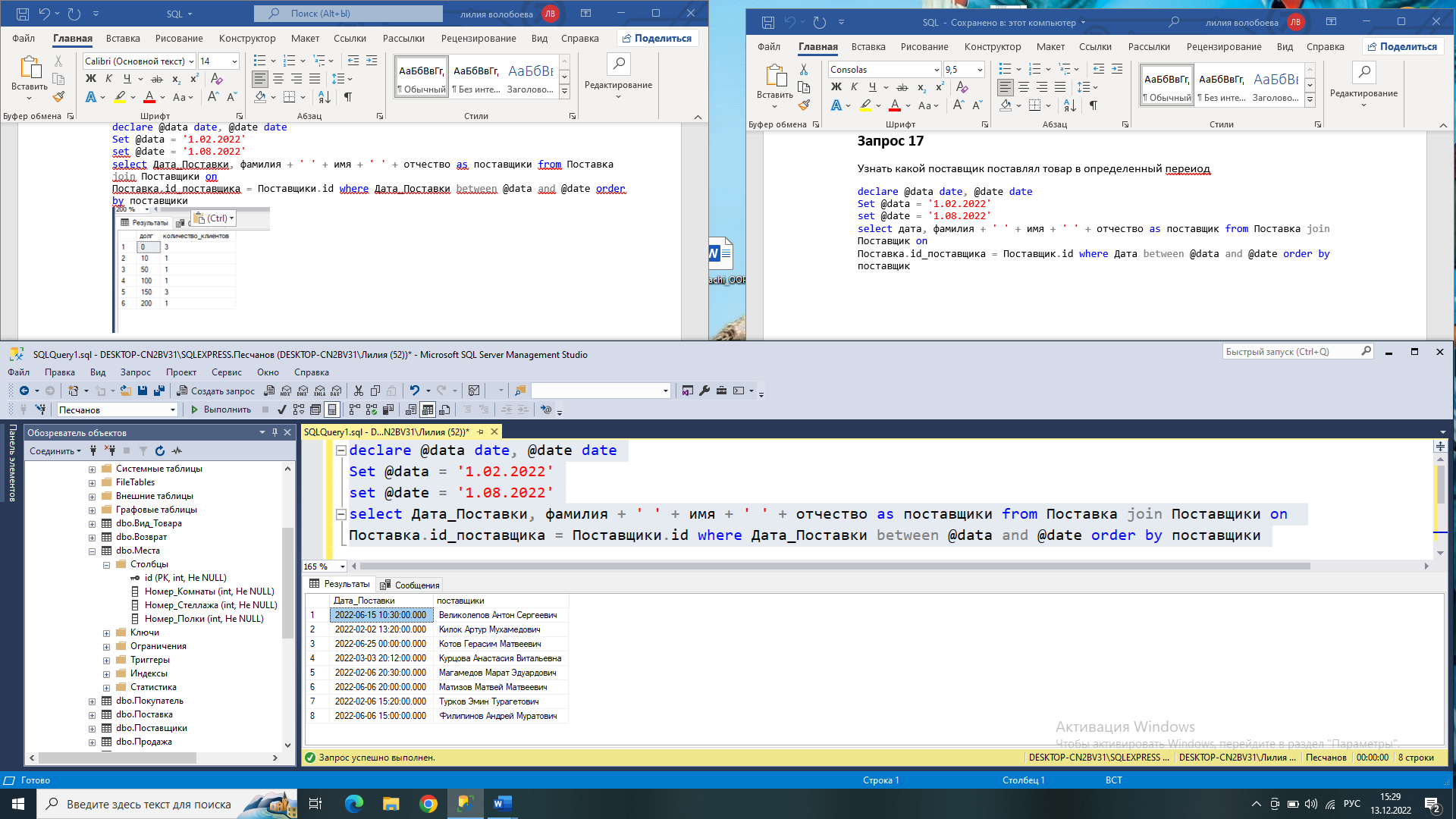
Назначение: Узнать поставщика и дату поставки

CREATE FUNCTION Поставки (@data date, @date date)

RETURNS TABLE

AS RETURN ( select Дата\_Поставки, фамилия + ' ' + имя + ' ' + отчество as поставщики from Поставка join Поставщики on Поставка.id\_поставщика = Поставщики.id where Дата\_Поставки between @data and @date)

SELECT\* FROM Поставки ('1.02.2022','1.08.2022')



Функция 8.

Назначение: вывести образование сотрудника по id

CREATE FUNCTION Образзование\_сотрудника (@id INT)

RETURNS CHAR(12)

AS

BEGIN

DECLARE @образование CHAR(12)

SELECT @образование = (SELECT образование FROM Сотрудник

WHERE [ID] = @id)

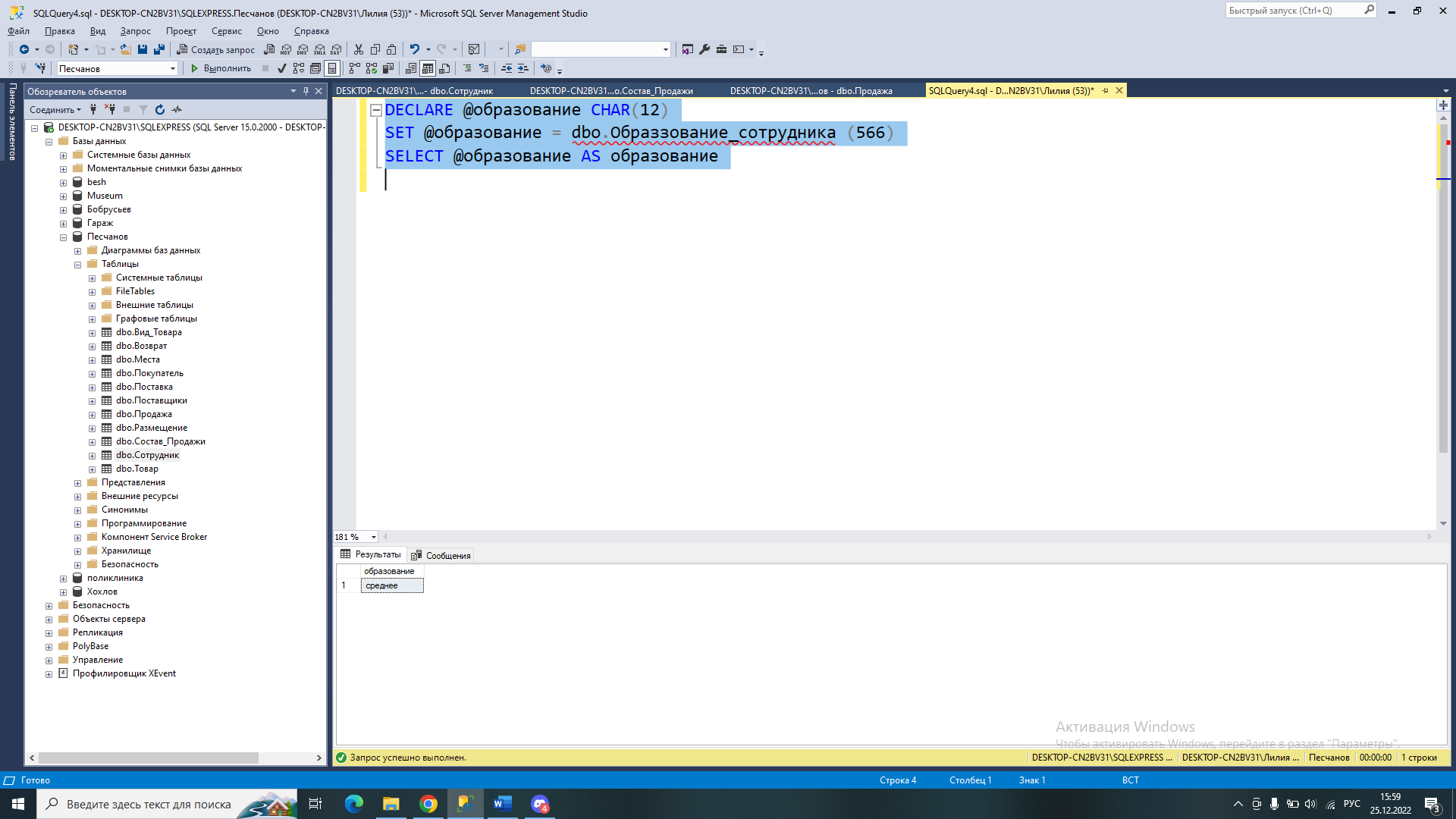
RETURN @образование

END;

DECLARE @образование CHAR(12)

SET @образование = dbo.Образзование\_сотрудника (566)

SELECT @образование AS образование



Функция 9.

Назначение: узнать дату продажи и покупателя

CREATE FUNCTION ПродажаПокупатель (@id1 int, @id2 int)

RETURNS TABLE

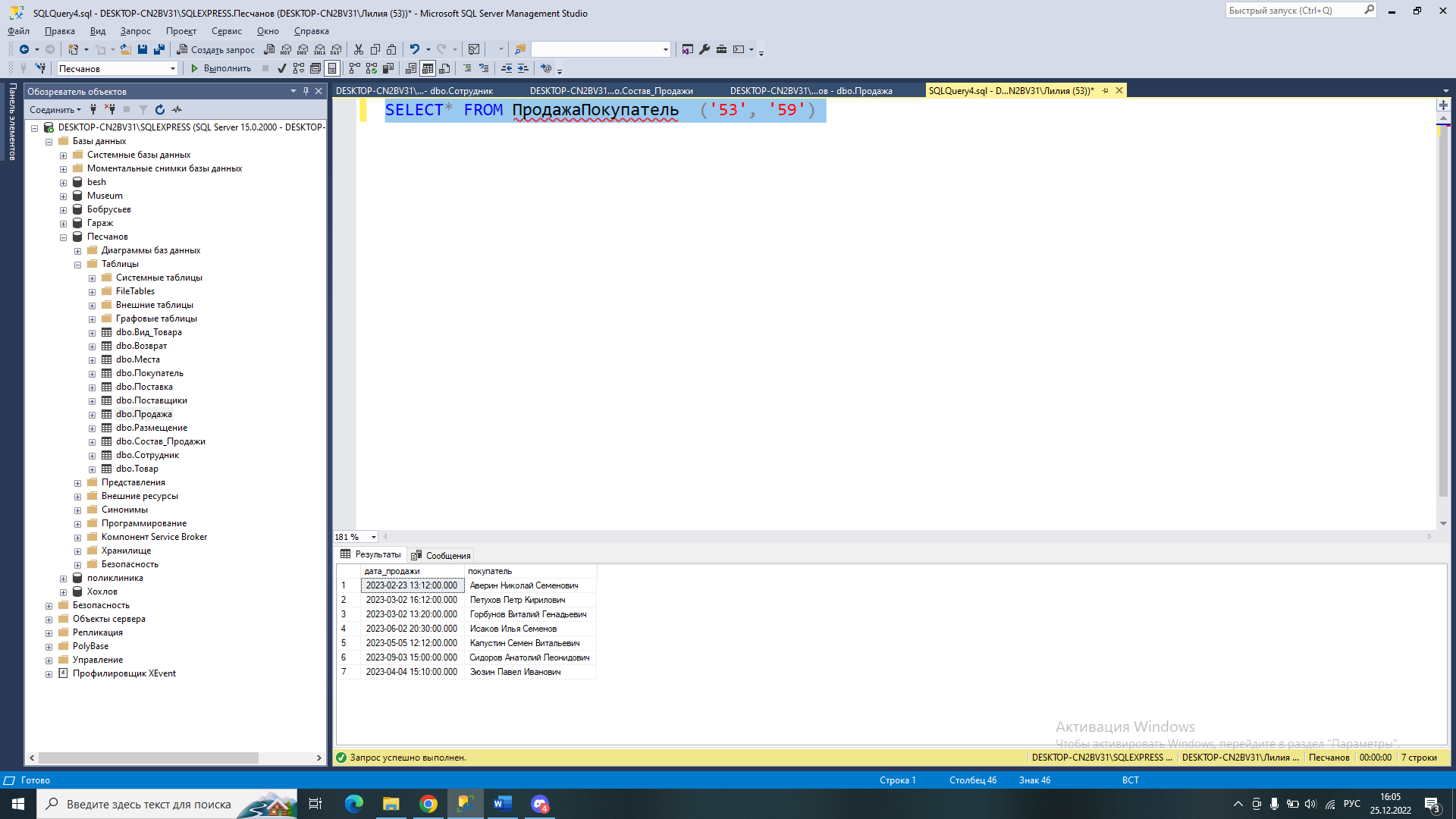
AS

RETURN

(select дата\_продажи,фамилия + ' ' + имя + ' ' + отчество as покупатель from Продажа join Покупатель

on Покупатель.id = Продажа.id\_Покупателя where продажа.id between @id1 and @id2)

SELECT\* FROM ПродажаПокупатель ('53', '59')



Функция 10.

Назначение: узнать скидку покупателя по ФИО.

CREATE FUNCTION скидка\_покупателя ( @Фамилия NVARCHAR(100),@Имя NVARCHAR(100), @Отчество NVARCHAR(50))

RETURNS INT

AS

BEGIN

DECLARE @Скидка INT

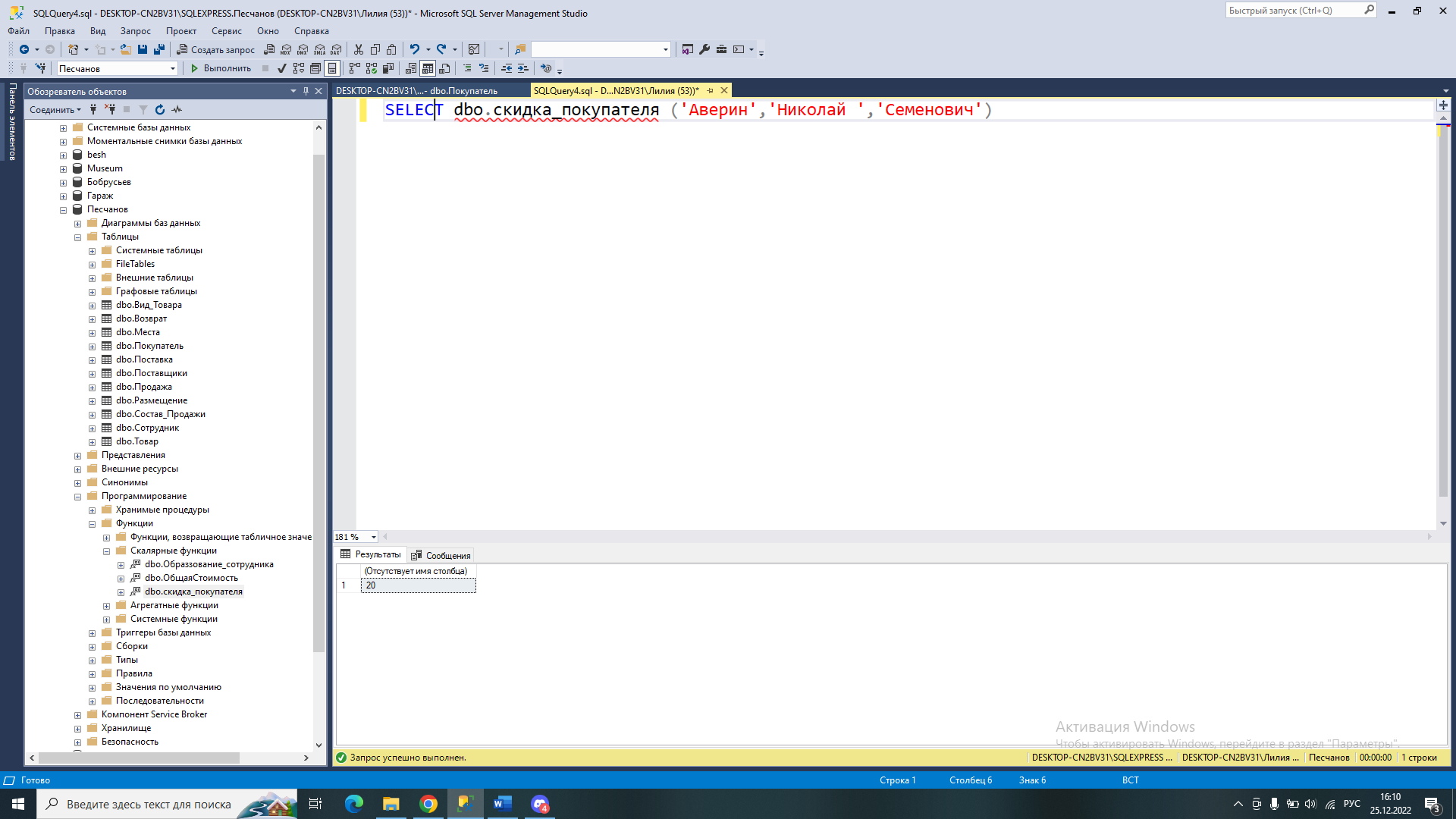
SELECT @Скидка = (SELECT скидка FROM Покупатель

WHERE Фамилия = @Фамилия AND Имя = @Имя AND Отчество = @Отчество)

RETURN @Скидка

END

SELECT dbo.скидка\_покупателя ('Аверин','Николай ','Семенович')



Функция 11.

Назначение: поиск сотрудника по должности

CREATE FUNCTION ДолжностьСотрудника (@должность varchar(50))

RETURNS TABLE

AS

RETURN

(select фамилия + ' ' + имя + ' ' + отчество as покупатель, Должность from Сотрудник

where Должность = @должность)

SELECT\* FROM ДолжностьСотрудника ('охранник')

