Учреждение образования

«Белорусский государственный технологический университет»

**Лабораторная работа №5**

Использование подзапросов

Выполнила:

Студентка 2 курса 1 группы ФИТ

Ксёнжик Ольга Александровна

**2022 г.**

1. На основе таблиц заключение контракта, фирмы и контактное лицо сформировать список номеров контракта, которые работают с фирмой, которая сотрудничает с контактным лицом. Примечание: использовать в секции WHERE предикат IN c некоррелированным подзапросом к таблице PROFESSION.

USE K\_MyBase2

SELECT Номер\_контракта

FROM Заключение\_контракта

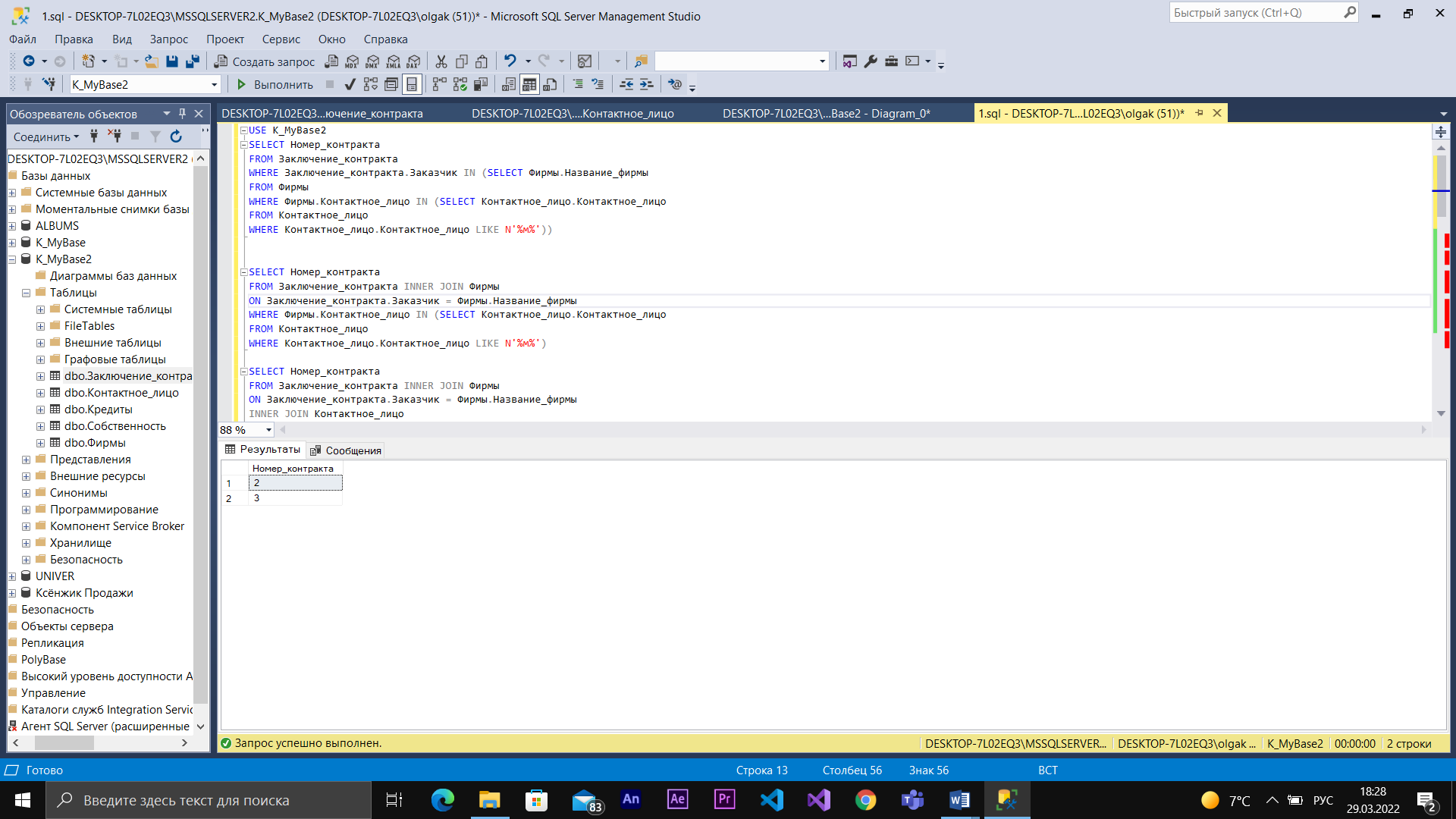
WHERE Заключение\_контракта.Заказчик IN (SELECT Фирмы.Название\_фирмы

FROM Фирмы

WHERE Фирмы.Контактное\_лицо IN (SELECT Контактное\_лицо.Контактное\_лицо

FROM Контактное\_лицо

WHERE Контактное\_лицо.Контактное\_лицо LIKE N'%м%'))



1. Переписать запрос пункта 1 таким образом, чтобы тот же подзапрос был записан в конструкции INNER JOIN секции FROM внешнего запроса. При этом результат выполнения запроса должен быть аналогичным результату исходного запроса.

SELECT Номер\_контракта

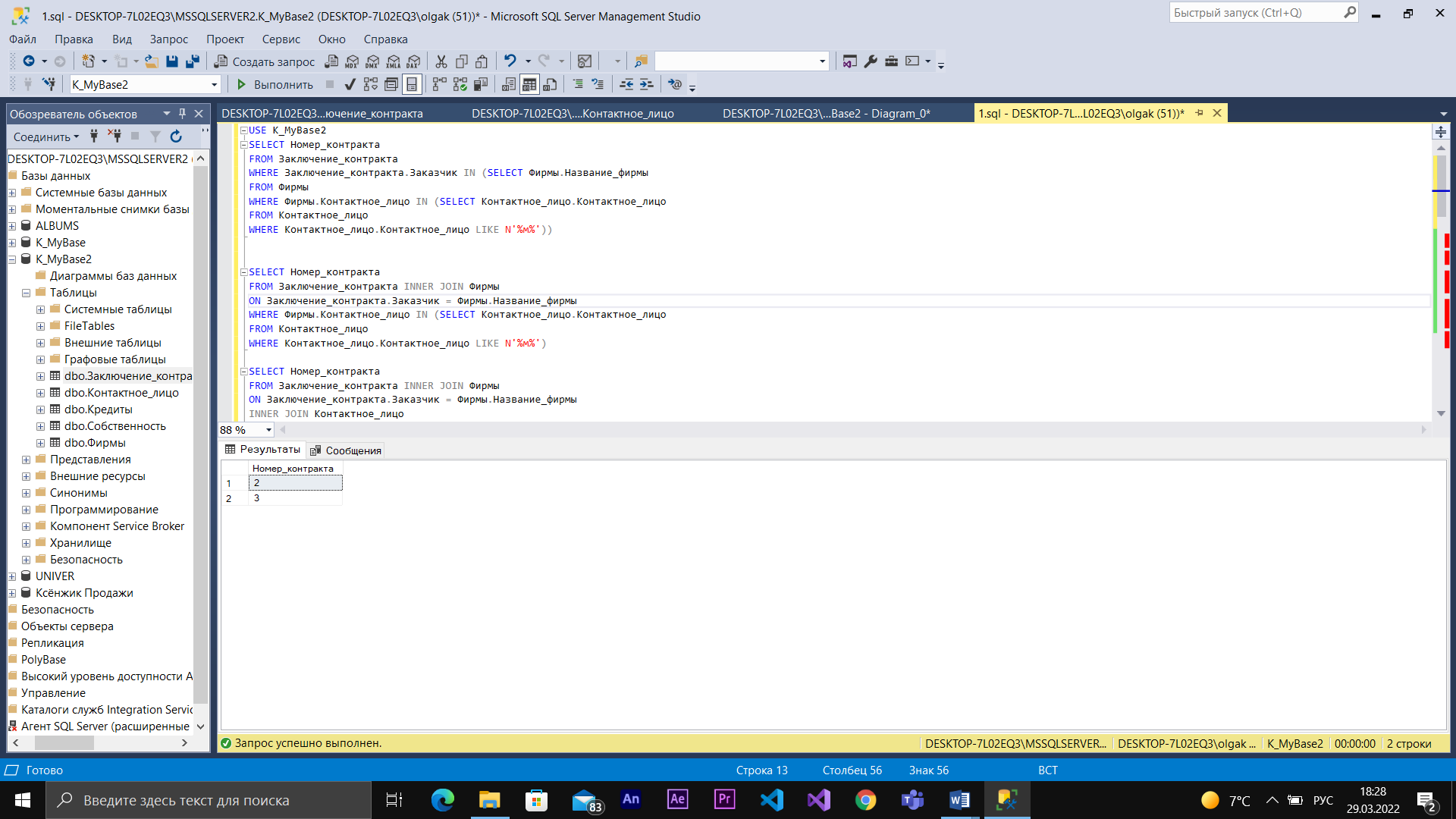
FROM Заключение\_контракта INNER JOIN Фирмы

ON Заключение\_контракта.Заказчик = Фирмы.Название\_фирмы

WHERE Фирмы.Контактное\_лицо IN (SELECT Контактное\_лицо.Контактное\_лицо

FROM Контактное\_лицо

WHERE Контактное\_лицо.Контактное\_лицо LIKE N'%м%')



1. Переписать запрос, реализующий 1 пункт без использования подзапроса. Примечание: использовать соединение INNER JOIN трех таблиц.

SELECT Номер\_контракта

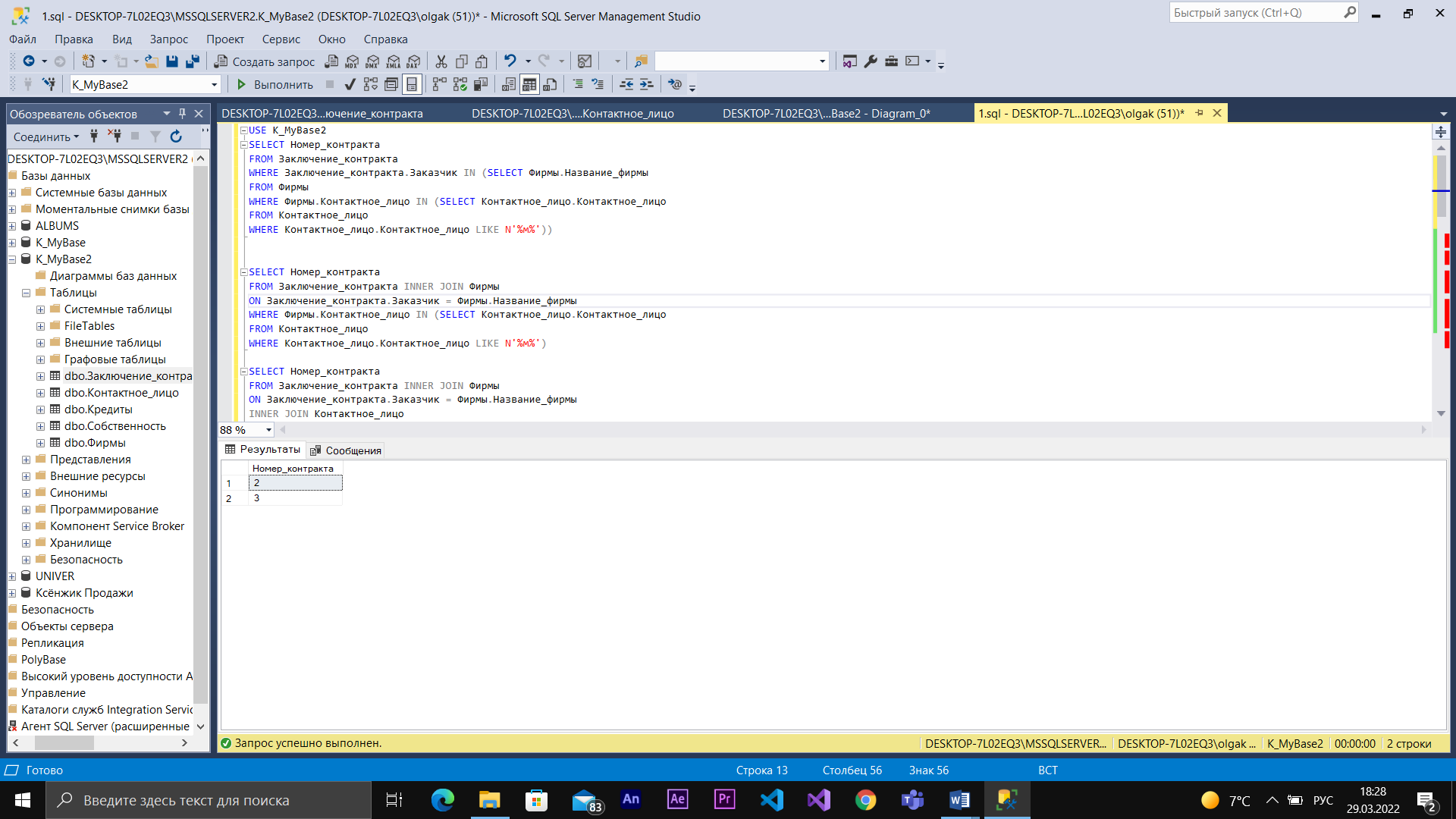
FROM Заключение\_контракта INNER JOIN Фирмы

ON Заключение\_контракта.Заказчик = Фирмы.Название\_фирмы

INNER JOIN Контактное\_лицо

ON Контактное\_лицо.Контактное\_лицо = ФИРМЫ.Контактное\_лицо

WHERE Контактное\_лицо.Контактное\_лицо LIKE N'%м%'



1. На основе таблицы AUDITORIUM сформировать список аудиторий самых больших вместимостей (столбец AUDI-TORIUM\_CAPACITY) для каждого типа аудитории (AUDITORIUM\_TYPE). При этом результат следует отсортировать в порядке убывания вместимости. Примечание: использовать коррелируемый подзапрос c секциями TOP и ORDER BY.

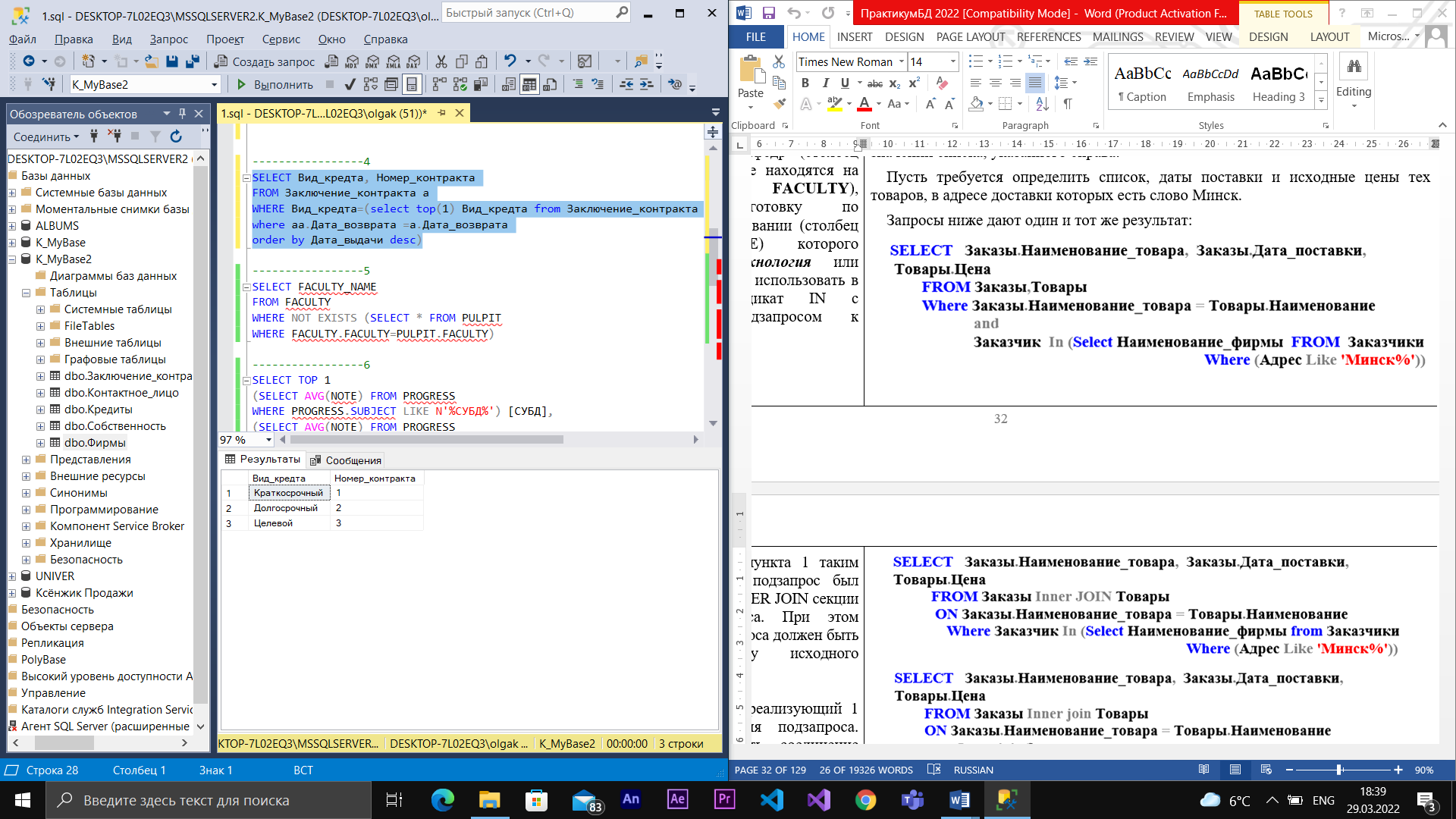
SELECT Вид\_кредта, Номер\_контракта

FROM Заключение\_контракта a

WHERE Вид\_кредта=(select top(1) Вид\_кредта from Заключение\_контракта aa

where aa.Дата\_возврата =a.Дата\_возврата

order by Дата\_выдачи desc)



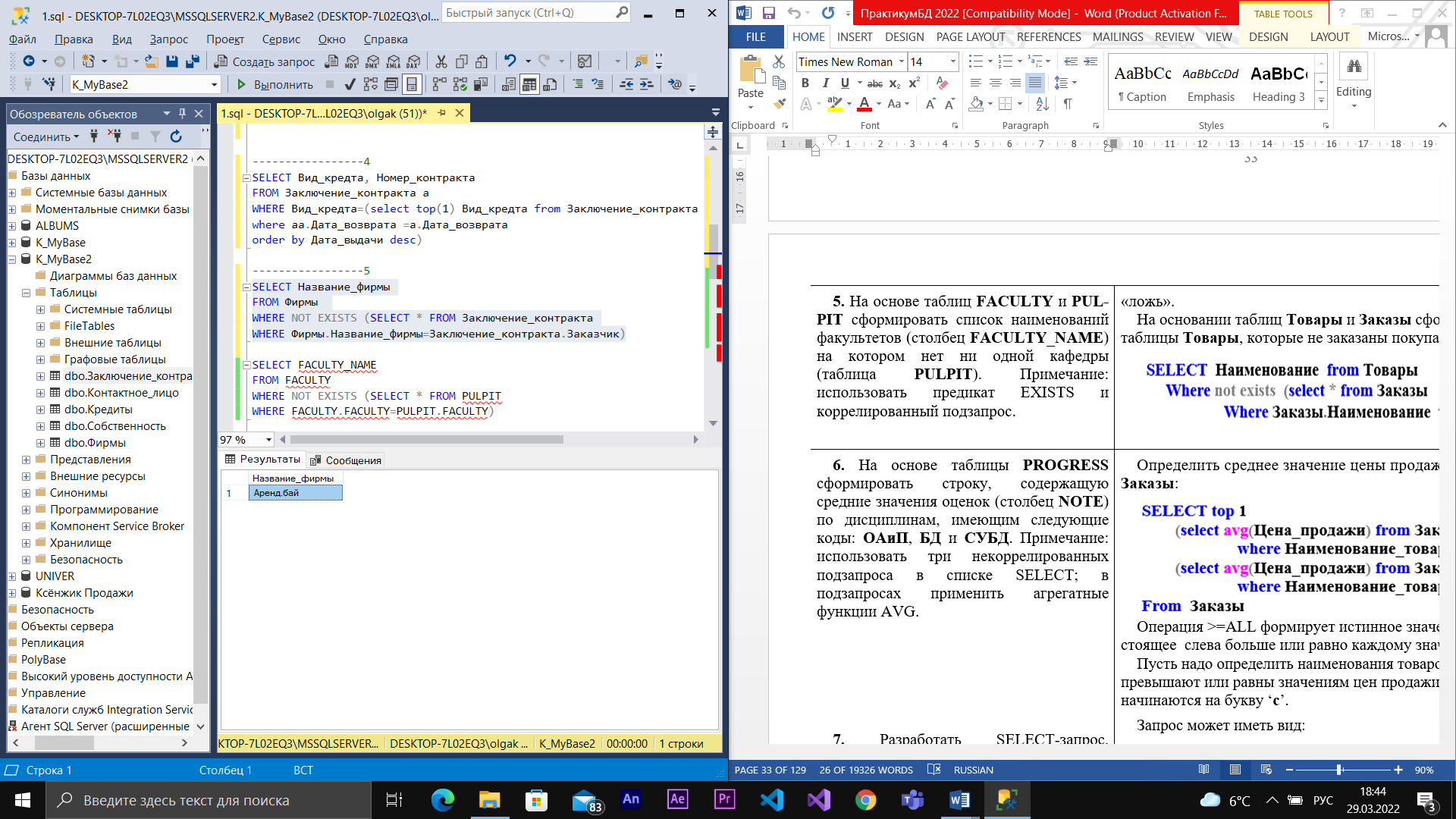
1. На основе таблиц FACULTY и PULPIT сформировать список наименований факультетов (столбец FACULTY\_NAME) на котором нет ни одной кафедры (таблица PULPIT). Примечание: использовать предикат EXISTS и коррелированный подзапрос.

SELECT Название\_фирмы

FROM Фирмы

WHERE NOT EXISTS (SELECT \* FROM Заключение\_контракта

WHERE Фирмы.Название\_фирмы=Заключение\_контракта.Заказчик)



1. На основе таблицы PROGRESS сформировать строку, содержащую средние значения оценок (столбец NOTE) по дисциплинам, имеющим следующие коды: ОАиП, БД и СУБД. Примечание: использовать три некоррелированных подзапроса в списке SELECT; в подзапросах применить агрегатные функции AVG.

SELECT TOP 1

(SELECT AVG(Цена\_собственности) FROM Собственность

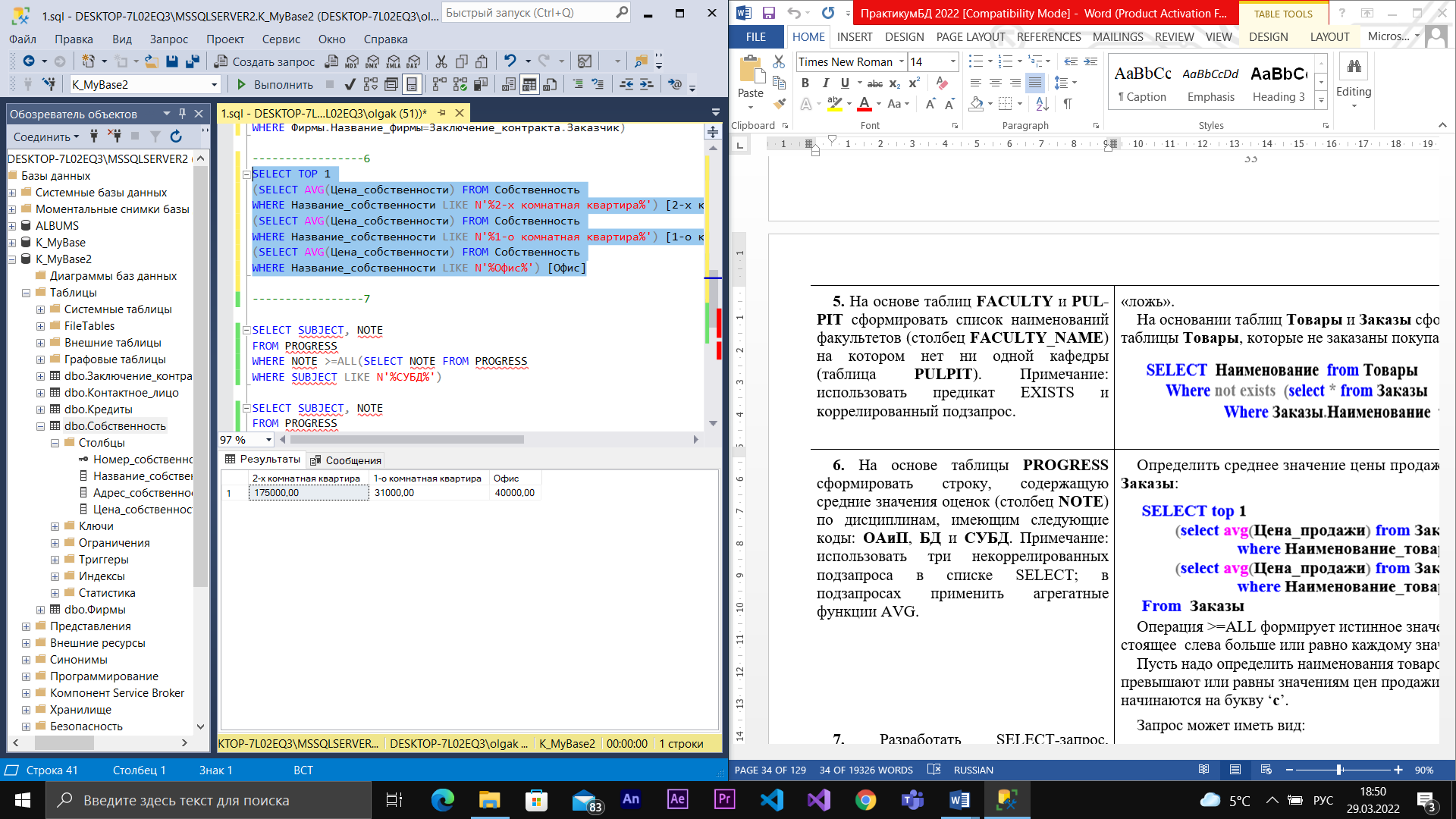
WHERE Название\_собственности LIKE N'%2-х комнатная квартира%') [2-х комнатная квартира],

(SELECT AVG(Цена\_собственности) FROM Собственность

WHERE Название\_собственности LIKE N'%1-о комнатная квартира%') [1-о комнатная квартира],

(SELECT AVG(Цена\_собственности) FROM Собственность

WHERE Название\_собственности LIKE N'%Офис%') [Офис]



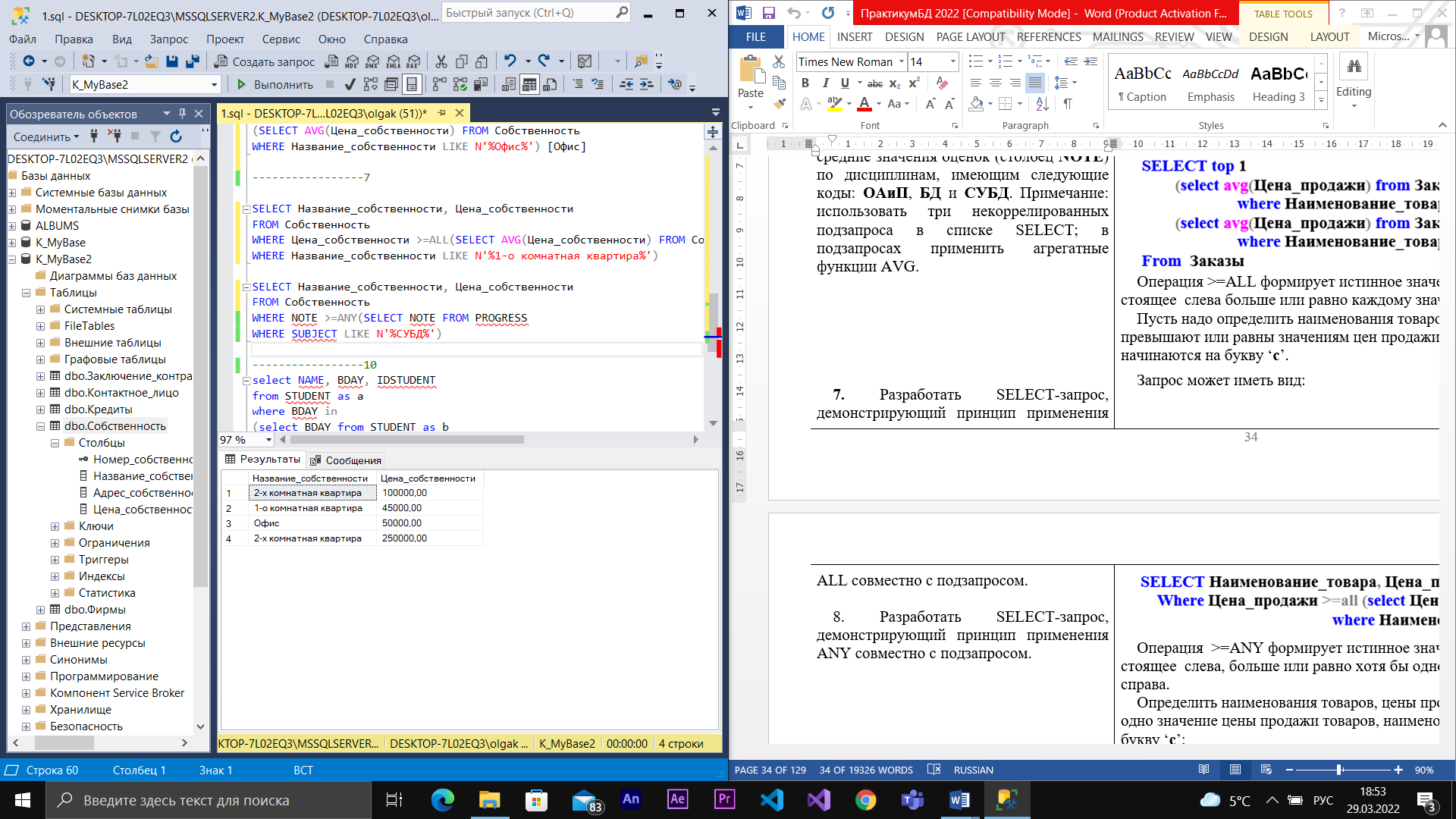
1. Разработать SELECT-запрос, демонстрирующий принцип применения ALL совместно с подзапросом.

SELECT Название\_собственности, Цена\_собственности

FROM Собственность

WHERE Цена\_собственности >=ALL(SELECT Цена\_собственности FROM Собственность

WHERE Название\_собственности LIKE N'%1-о комнатная квартира%')



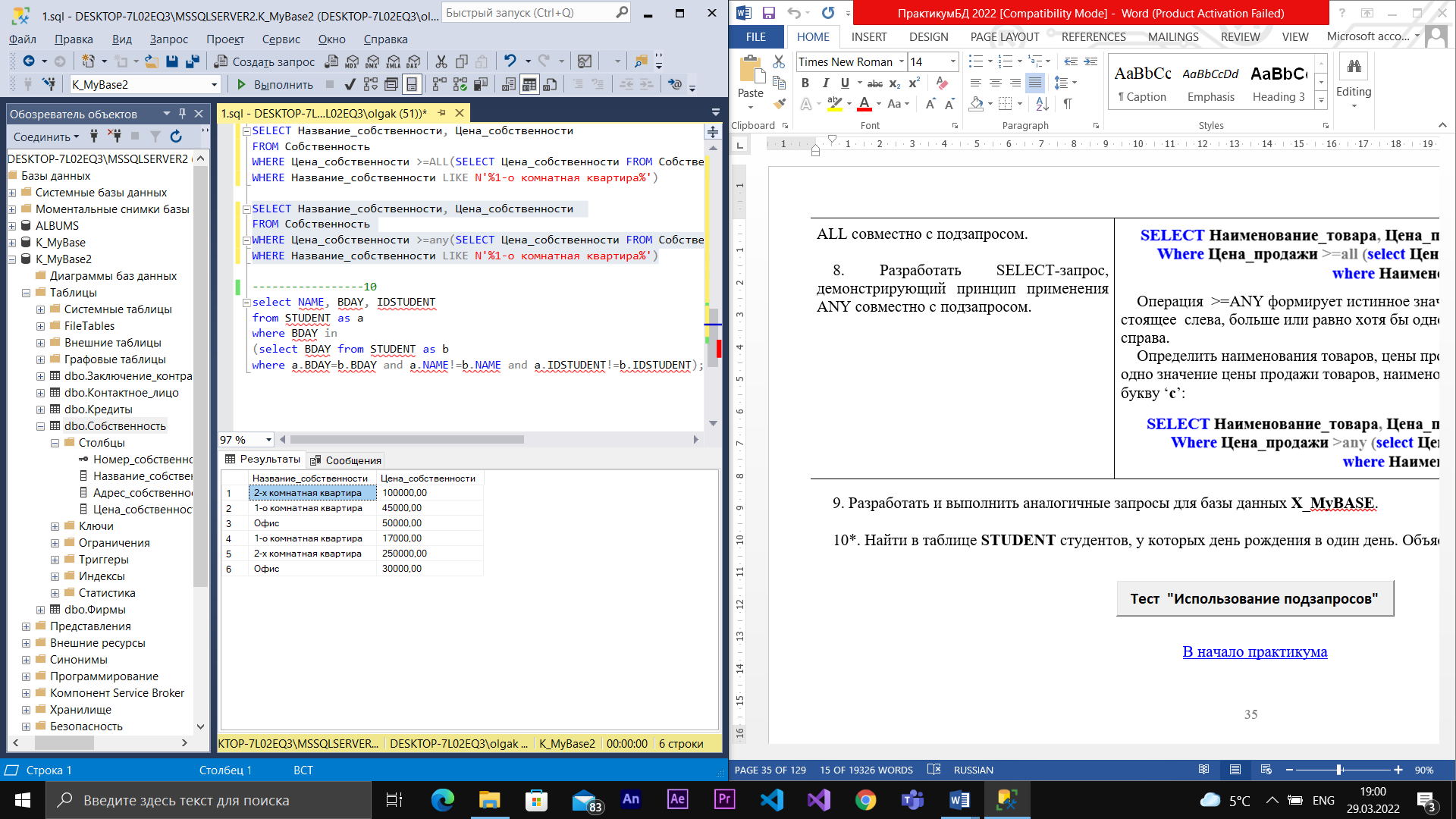
1. Разработать SELECT-запрос, демонстрирующий принцип применения ANY совместно с подзапросом

SELECT Название\_собственности, Цена\_собственности

FROM Собственность

WHERE Цена\_собственности >=any(SELECT Цена\_собственности FROM Собственность

WHERE Название\_собственности LIKE N'%1-о комнатная квартира%')



1. Найти в таблице **STUDENT** студентов, у которых день рождения в один день. Объяснить решение.

use UNIVER

select NAME, BDAY, IDSTUDENT

from STUDENT as a

where BDAY in

(select BDAY from STUDENT as b

where a.BDAY=b.BDAY and a.NAME!=b.NAME and a.IDSTUDENT!=b.IDSTUDENT);

