**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Модели данных и системы управления базами данных**

Итоговый отчет по индивидуальному заданию

**Выполнила:**

Зуйкевич Лидия

4 курс 7 группа

**Преподаватель:**

Мороз А. С.

Минск, 2023

**Оглавление**

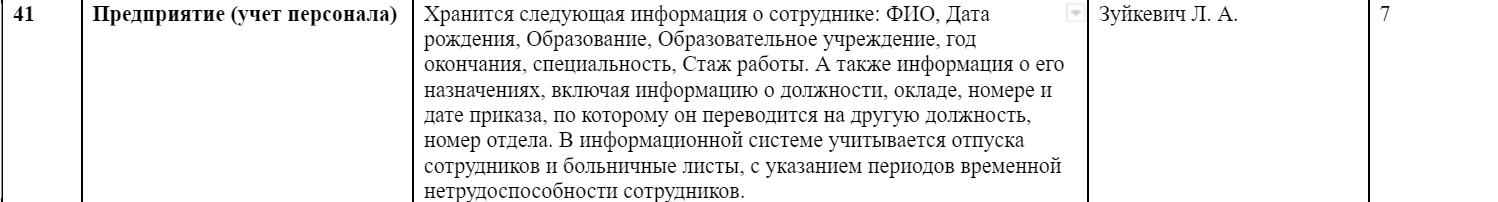
[Лабораторная работа №2. Концептуальное и логическое проектирование реляционной базы данных 3](#_Toc153659600)

[Дополнение по лабораторным 2 – 4 11](#_Toc153659601)

[Дополнение к лабораторным работам 5 – 6 26](#_Toc153659602)

[Дополнение к лабораторным работам 7 - 9 33](#_Toc153659603)

# **Лабораторная работа №2. Концептуальное и логическое проектирование реляционной базы данных**



**Требования**

1) БД должна хранить информацию о сотрудниках (ФИО, дата рождения, образование, образовательного учреждения, год окончания, специальность, стаж работы)

2) Должна храниться информация о назначениях сотрудника (должность, оклад, номер и дата приказа, номер отдела)

3) Должна храниться информация об отпусках сотрудников (номер приказа, период временной нетрудоспособности, выплаты (если есть))

4) Должна храниться информация о больничных листах сотрудников (номер справки, период временной нетрудоспособности, выплаты (если есть))

5) Возможность просмотра всех назначений конкретного сотрудника

6) Возможность просмотра информации обо всех отпусках сотрудника

7) Возможность просмотра информации обо всех больничных листах сотрудника

8) Возможность просмотреть всех сотрудников определенного отдела

9) Возможность просмотреть всех сотрудников, обучавшихся в определенном УО, добавить информацию о повышении квалификации

**Выбор сущностей и атрибутов**

**Сотрудник**

ID сотрудника

Фамилия

Имя

Отчество

Дата рождения

Стаж работы

**Учреждение образования**

ID УО

Название

Рейтинг УО (от него так же будет зависеть рейтинг сотрудника)

**Сотрудник – Учреждение образования**

ID сотрудника

ID УО

Образование

Специальность

Год окончания

**Назначение**

Номер приказа

ID Сотрудника

Должность

Оклад

Номер отдела

Дата перевода

Узнать нынешнюю должность/оклад сотрудника: смотрим по ID и берем по максимальной дате.

**Отпуск**

Номер приказа

ID Сотрудника

Период временной нетрудоспособности

Выплаты (необязательный атрибут)?

**Больничный лист**

Номер справки

ID Сотрудника

Период временной нетрудоспособности

Выплаты (необязательный атрибут)?

**Связи**

**Сотрудник - Учреждение образования** (M:N – в одном УО могли учиться разные сотрудники и 1 сотрудник мог закончить несколько УО + повышение квалификации бывает в ВУЗах, например). Связь что-то вроде «закончил/обучался», атрибуты связи: год окончания, специальность

**Сотрудник – назначение** (1:N - один сотрудник может иметь несколько назначений)

**Сотрудник – отпуск** (1:N - один сотрудник может уходить в отпуск несколько раз)

**Сотрудник – больничный лист** (1:N - у одного сотрудника может быть несколько больничных листов)

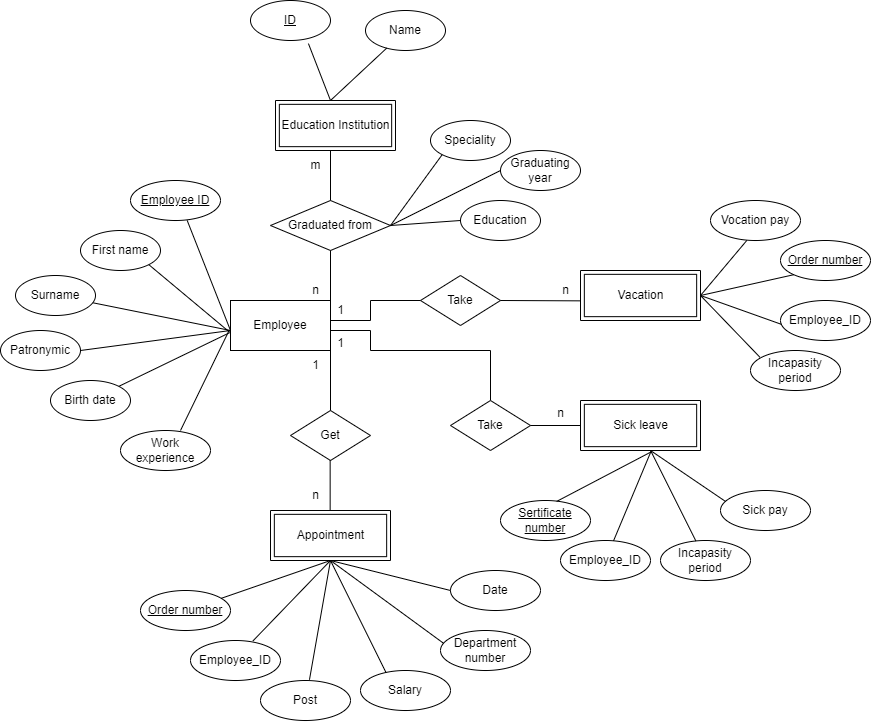


Рис. 1 - ER-диаграмма **до нормализации** (нотация Чена)

**Нормализация отношений**

1 НФ

Разбили составной атрибут ФИО на 3 простых, вынесли многозначный атрибут Учреждение образования в отдельные отношения.

Период нетрудоспособности разбивается на 2 поля: дата начала и дата конца.

2 НФ

БД находится в первой нормальной форме, во всех таблицах есть ключи. У нас только 1 составной ключ – в таблице Сотрудник – УО, все неключевые столбцы в этой таблице зависят от полного ключа.

3 НФ

Атрибуты в каждом отношении зависят от первичного ключа и не зависят от других атрибутов. Для этого были убраны атрибуты, хранящие информацию о выплатах, в сущностях Отпуск и Больничный лист, поскольку выплаты зависят от оклада сотрудника (соответственно, некоторым образом связаны с его ID) и периода нетрудоспособности: выплаты рассчитываются как некоторый процент от оклада, процент зависит от длительности отпуска и некоторых свойств сотрудника (наличие детей и т.д.).

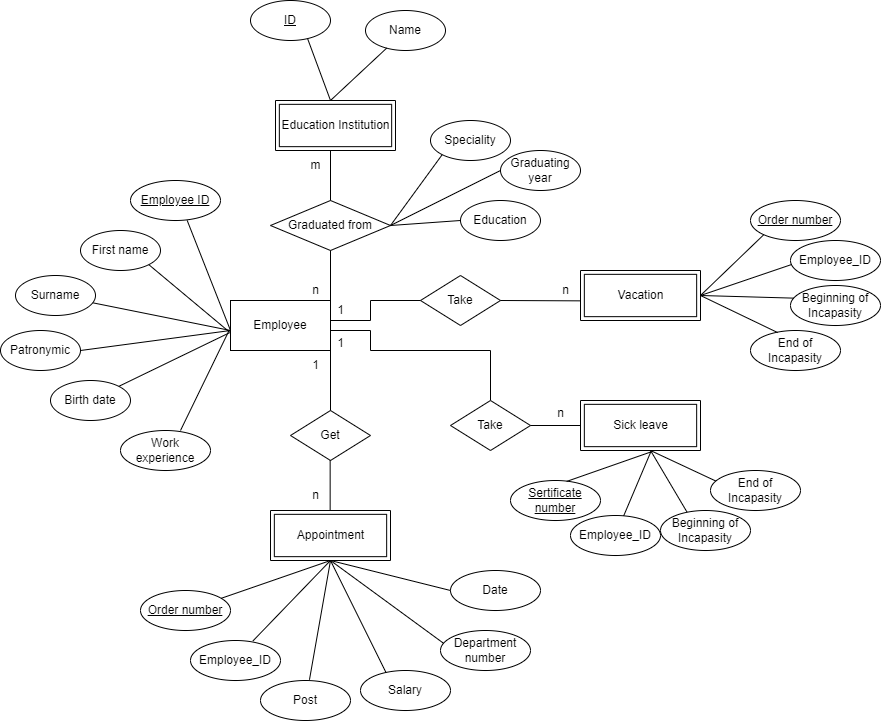


Рис. 2 - ER-диаграмма в нотации Чена **после нормализации**

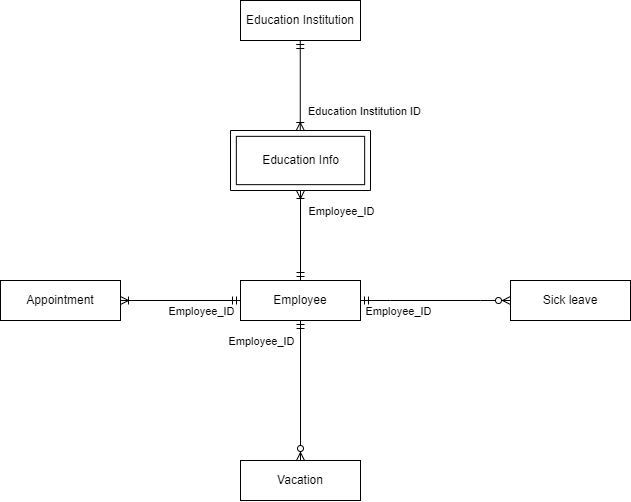


Рис. 3 – логическая модель реляционной БД

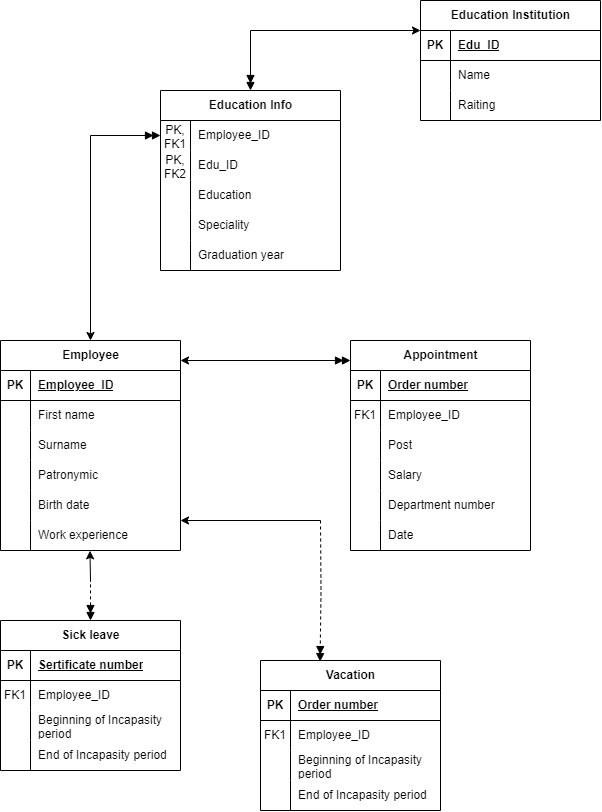


Рис. 4 – логическая модель БД

**Сотрудник (Employee)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| ID сотрудника | EMPL\_ID | INT | IDENTITY  (1,1) | Обязательное поле | Первичный ключ |
| Фамилия | SURNAME | NVARCHAR (20) |  | Обязательное поле |  |
| Имя | NAME | NVARCHAR (20) |  | Обязательное поле |  |
| Отчество | PATRONIMIC | NVARCHAR (20) | NULL | Необязательное поле |  |
| Дата рождения | B\_DATE | DATE |  | Обязательное поле |  |
| Стаж работы (в мес.) | WORK\_EXP | SMALLINT | 0 | Обязательное поле |  |

**Учреждение образования (Education Institution)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| ID УО | EDU\_ID | INT | IDENTITY (1,1) | Обязательное поле | Первичный ключ |
| Название | EDU\_NAME | NVARCHAR (100) |  | Обязательное поле |  |
| Рейтинг УО | EDU\_RATING |  |  | Обязательное поле, целые значения от 1 до 10 |  |

**Сотрудник – Учреждение образования (Education)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| ID сотрудника | EMPL\_ID | INT |  | Обязательное поле | Составной первичный ключ, внешний ключ к УО |
| ID УО | EDU\_ID | INT |  | Обязательное поле | Составной первичный ключ, внешний ключ к Сотрудник |
| Образование | EDUCATION | VARCHAR (60) |  | Обязательное поле |  |
| Специальность | SPECIALITY | VARCHAR (60) |  | Обязательное поле |  |
| Год окончания | GRAD\_YEAR | SMALLINT |  | Обязательное поле |  |

**Назначение (Appointment)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| Номер приказа | ORDER\_NUM | INT | IDENTITY (1,1) | Обязательное поле | Первичный ключ |
| ID Сотрудника | EMPL\_ID | INT |  | Обязательное поле |  |
| Должность | POST | VARCHAR(40) |  | Обязательное поле |  |
| Оклад | SALARY | DECIMAL (10, 2) |  | Обязательное поле |  |
| Номер отдела | DEPARTMENT | SMALLINT |  | Обязательное поле |  |
| Дата перевода | APP\_DATE | DATE |  | Обязательное поле |  |

**Отпуск (Vacation)**

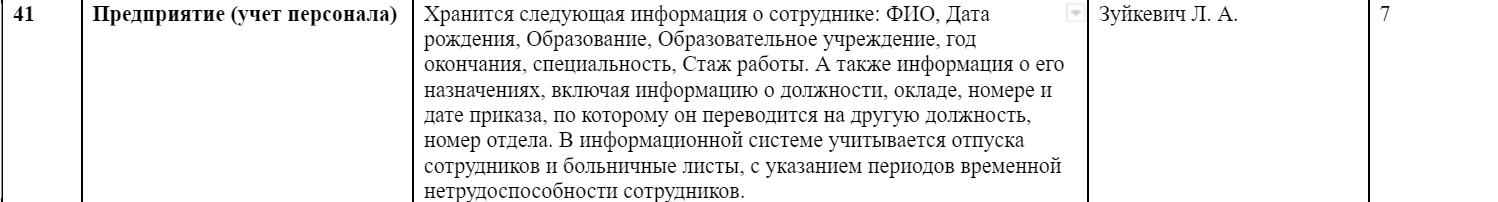
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| Номер приказа | ORDER\_NUM | INT | IDENTITY (1,1) | Обязательное поле | Первичный ключ |
| ID Сотрудника | EMPL\_ID | INT |  | Обязательное поле |  |
| Начало периода временной нетрудоспособности | INC\_BEGINNING | DATE |  | Обязательное поле |  |
| Конец периода временной нетрудоспособности | INC\_END | DATE |  | Обязательное поле |  |

**Больничный лист (Sick leave)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название атрибута | Имя поля | Тип, размер | Значение по умолчанию | Ограничения | Ключ или индекс |
| Номер справки | SERTIFICATE\_NUM | INT | IDENTITY (1,1) | Обязательное поле | Первичный ключ |
| ID Сотрудника | EMPL\_ID | INT |  | Обязательное поле |  |
| Начало периода временной нетрудоспособности | INC\_BEGINNING | DATE |  | Обязательное поле |  |
| Конец периода временной нетрудоспособности | INC\_END | DATE |  | Обязательное поле |  |

# **Дополнение по лабораторным 2 – 4**

1. Из лабораторной работы №2 (индивидуальное задание) привести постановку задачи, описание предметной области



**Требования**

1) БД должна хранить информацию о сотрудниках (ФИО, дата рождения, образование, образовательное учреждение, год окончания УО, специальность, стаж работы)

2) Должна храниться информация о назначениях сотрудника (должность, оклад, номер и дата приказа, номер отдела)

3) Должна храниться информация об отпусках сотрудников (номер приказа, период временной нетрудоспособности, выплаты (если есть))

4) Должна храниться информация о больничных листах сотрудников (номер справки, период временной нетрудоспособности, выплаты (если есть))

5) Возможность просмотра всех назначений конкретного сотрудника

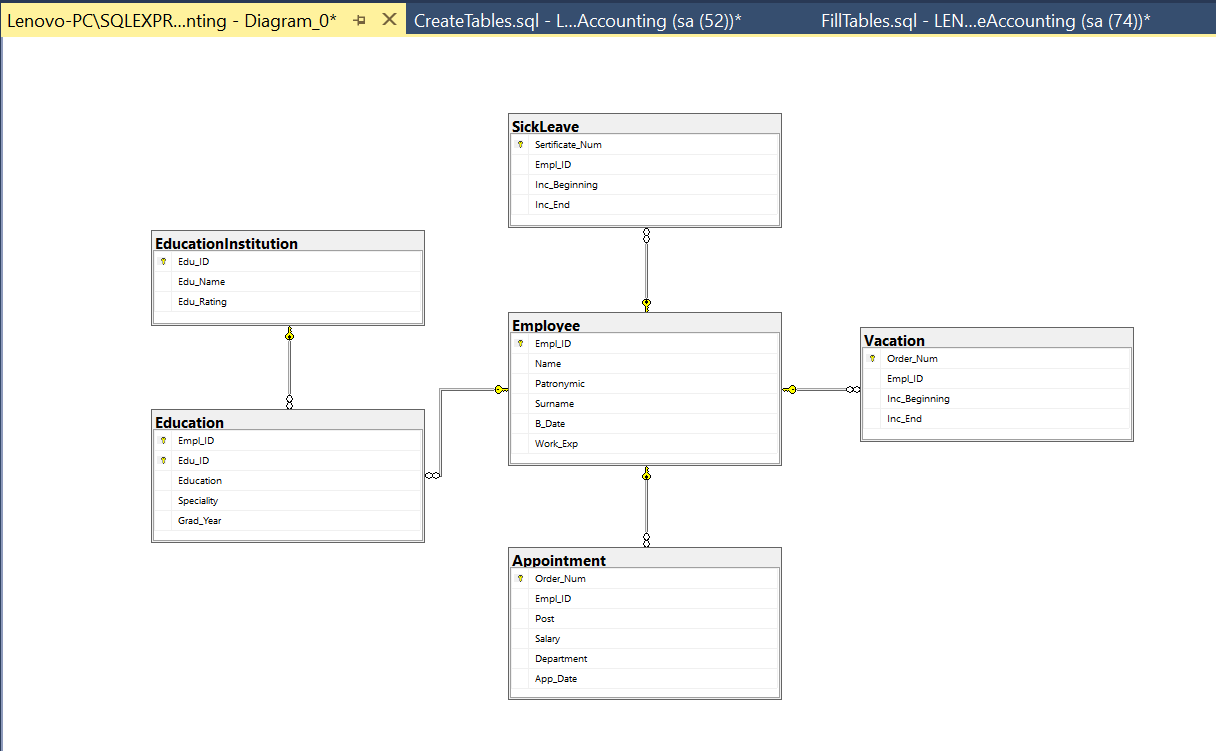
6) Возможность просмотра информации обо всех отпусках сотрудника

7) Возможность просмотра информации обо всех больничных листах сотрудника

8) Возможность просмотреть всех сотрудников определенного отдела

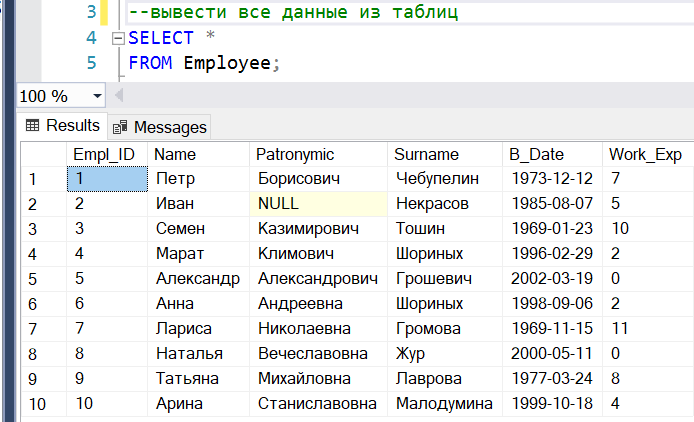
9) Возможность просмотреть всех сотрудников, обучавшихся в определенном УО, добавить информацию о повышении квалификации

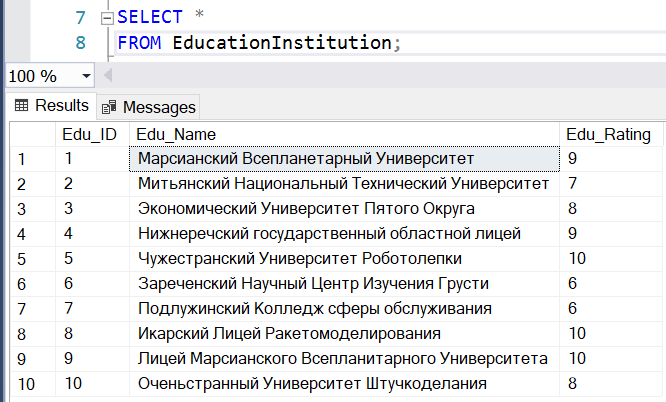
2. С помощью SQL Management Studio (SSM) отобразить структуры таблиц, ранее вами созданные, данные в таблицах, диаграмму баз данных

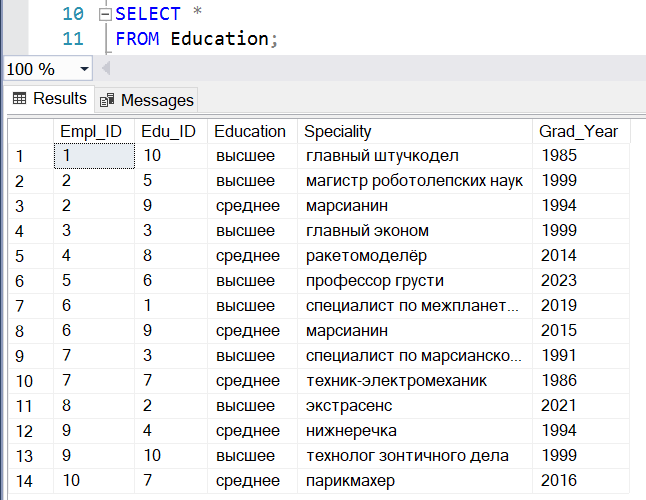


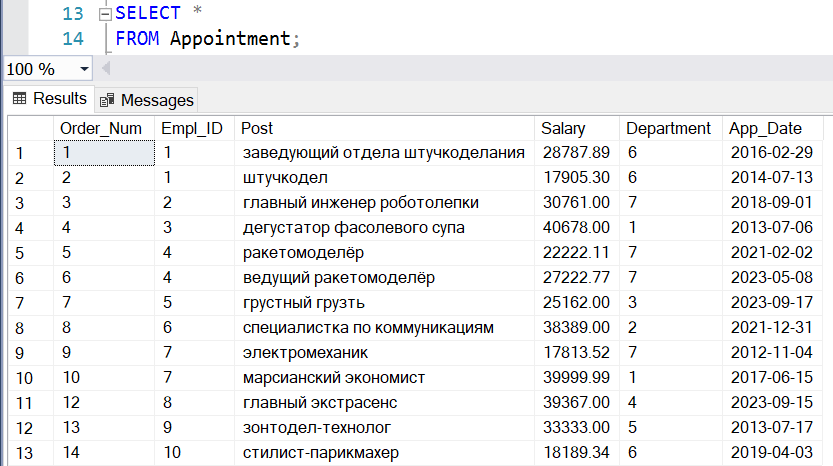
3. С помощью языка T-SQL выполнить следующие запросы

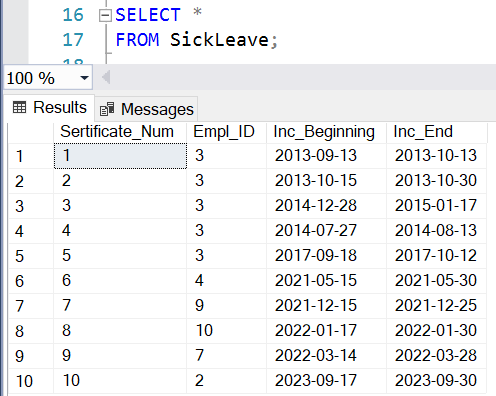
* вывести все данные из таблиц

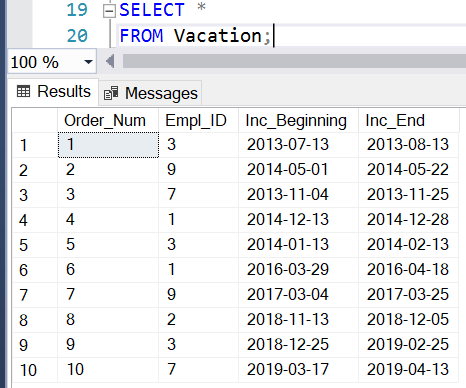




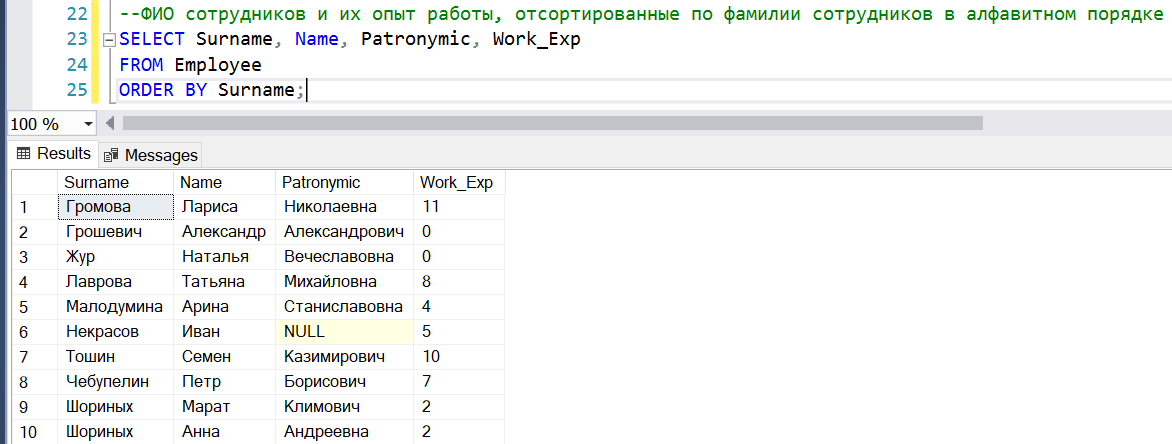




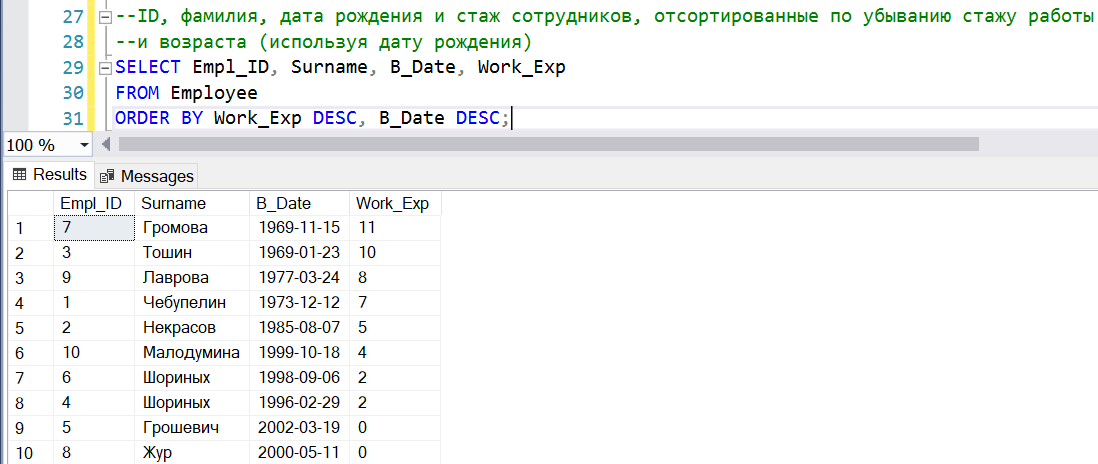




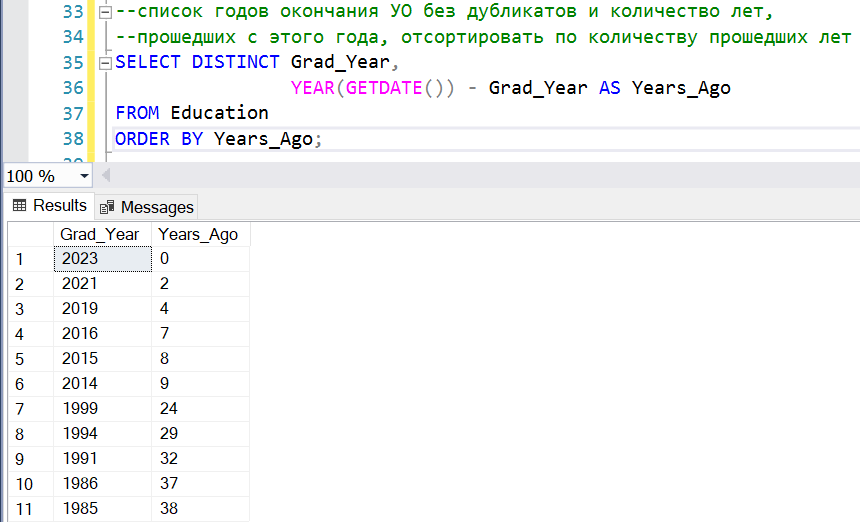
* вывести ФИО сотрудников и их опыт работы, отсортировав по фамилии сотрудников в алфавитном порядке



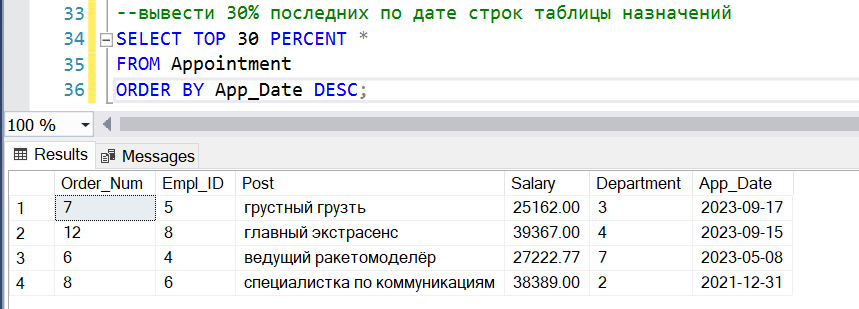
* вывести ID, фамилию, дату рождения и стаж сотрудников, отсортировав по убыванию стажа работы и возраста (используя дату рождения)



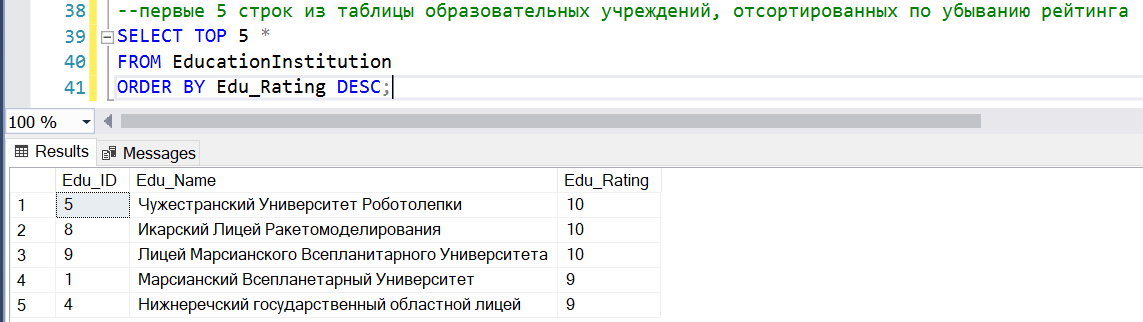
* вывести список годов окончания УО без дубликатов и количество лет, прошедших с года окончания УО, отсортировав по количеству прошедших лет



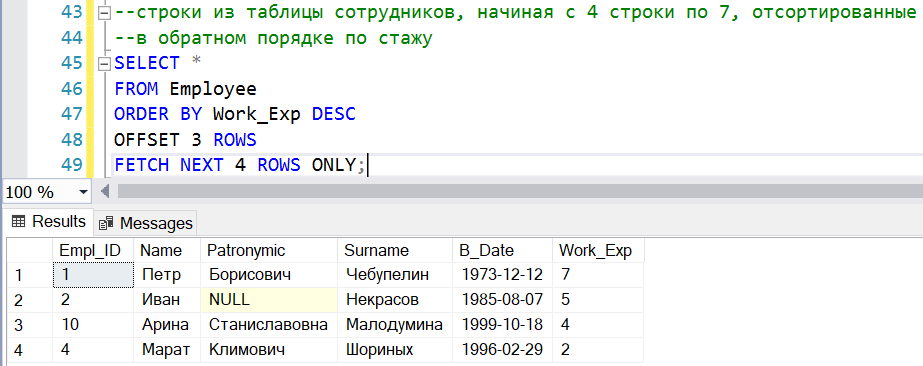
* вывести 30% последних по дате строк таблицы назначений



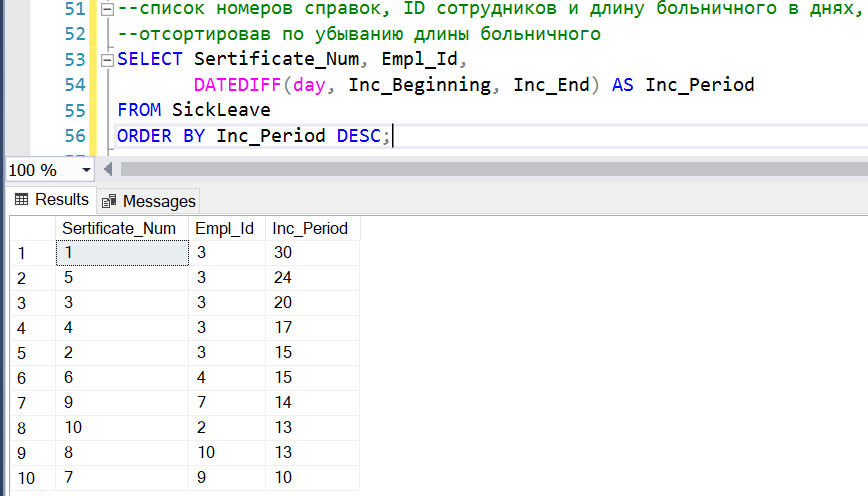
* вывести первые 5 строк из таблицы образовательных учреждений, отсортированных по убыванию рейтинга



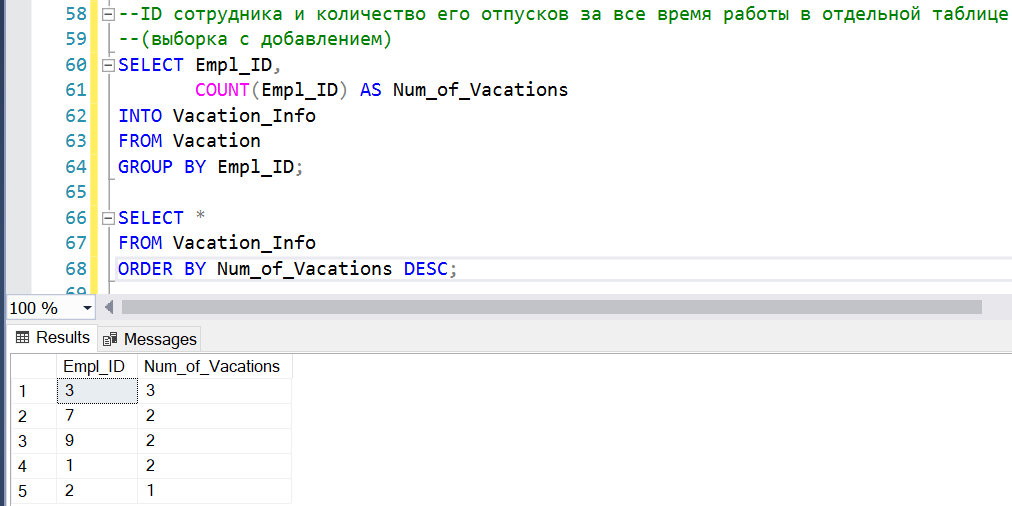
* вывести строки из таблицы сотрудников, начиная с 4 строки по 7, отсортированные в обратном порядке по стажу



* вывести список номеров справок, ID сотрудников и длину больничного в днях, отсортировав по убыванию длины больничного

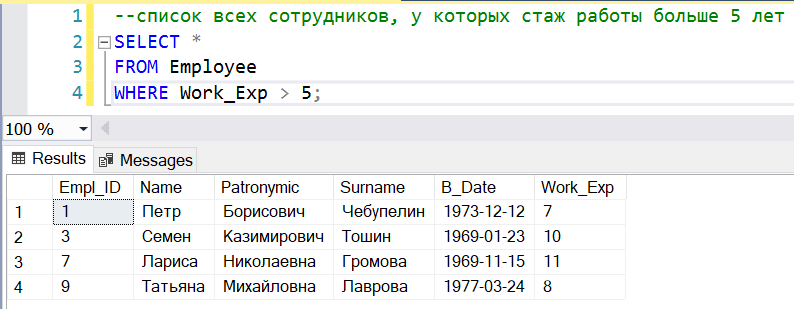


* вывести ID сотрудника и количество его отпусков за все время работы в отдельную таблицу с помощью SELECT INTO

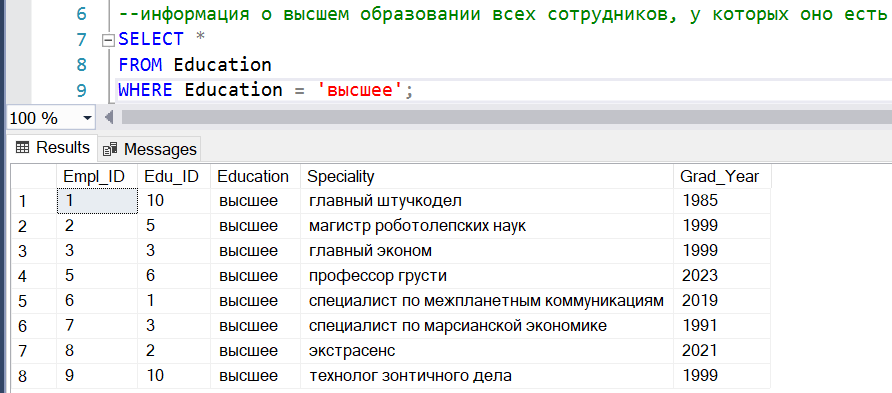


4. С помощью языка T-SQL написать и выполнить 7 запросов на фильтрацию данных с использованием в запросах операторов сравнения, логических операторов

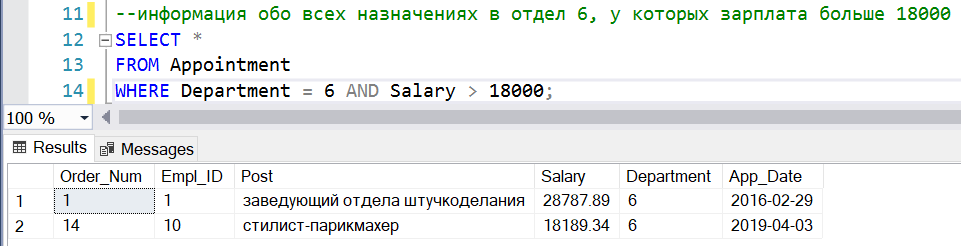
1) Вывести список всех сотрудников, у которых стаж работы больше 5 лет



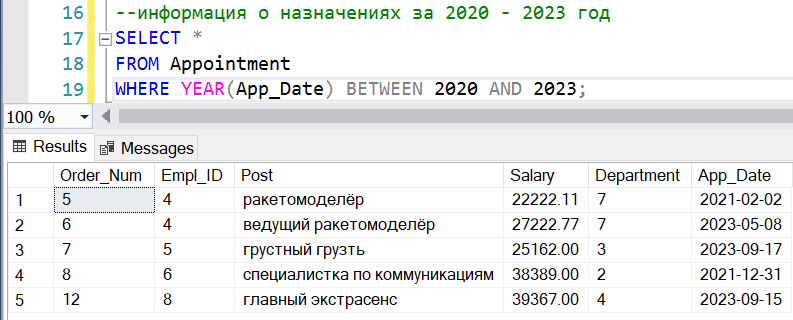
2) Вывести информацию о высшем образовании всех сотрудников, у которых оно есть



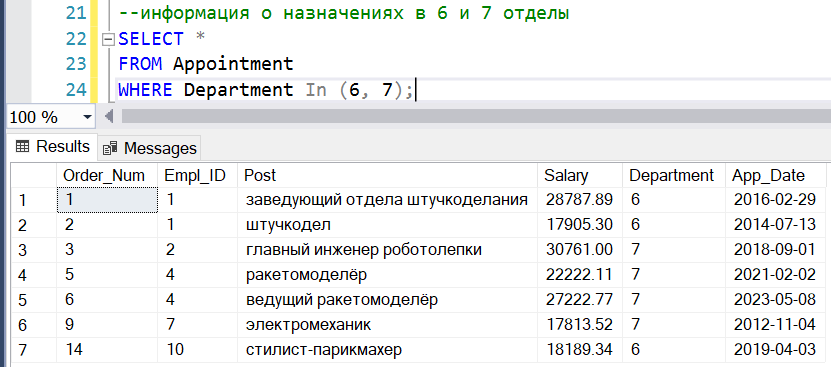
3) Вывести информацию обо всех назначениях в отдел 6, у которых зарплата больше 18000



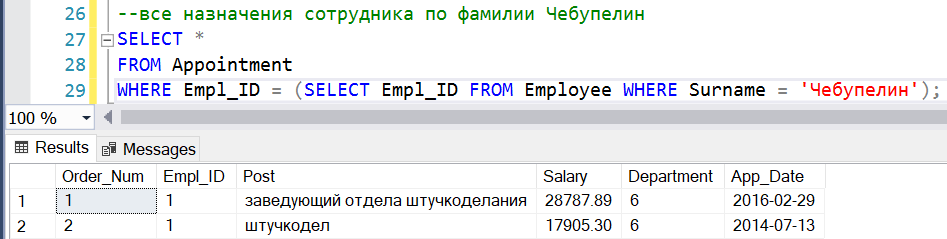
4) Вывести информацию о назначениях за 2020 - 2023 год



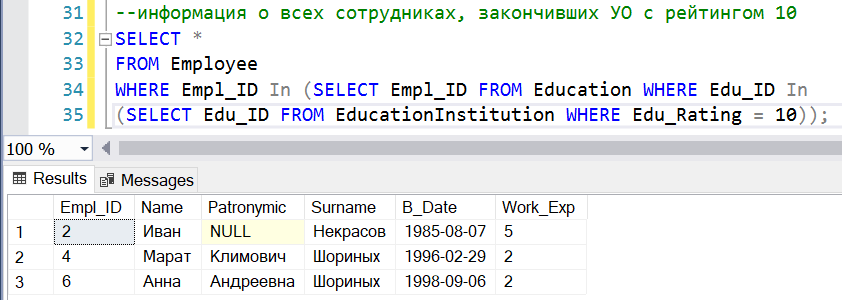
5) Вывести информацию о назначениях в 6 и 7 отделы



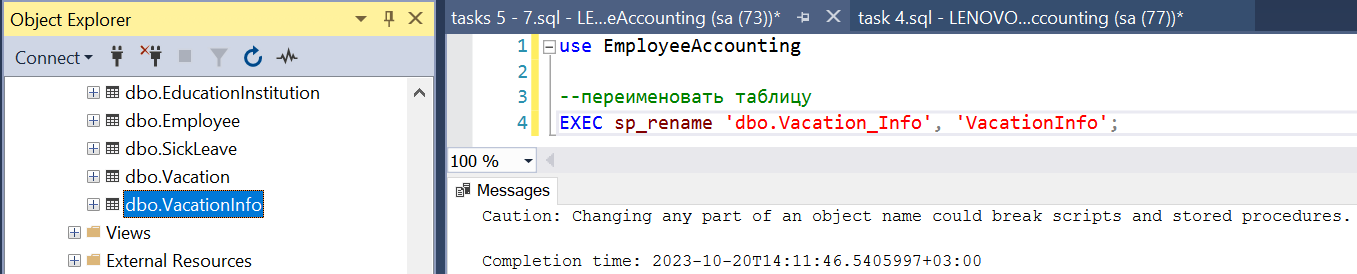
6) Вывести все назначения сотрудника по фамилии Чебупелин



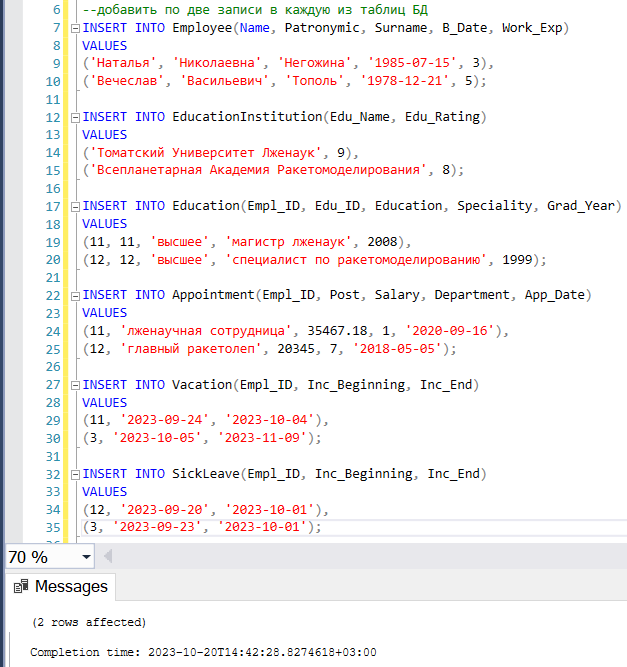
7) Вывести информацию о всех сотрудниках, закончивших УО с рейтингом 10

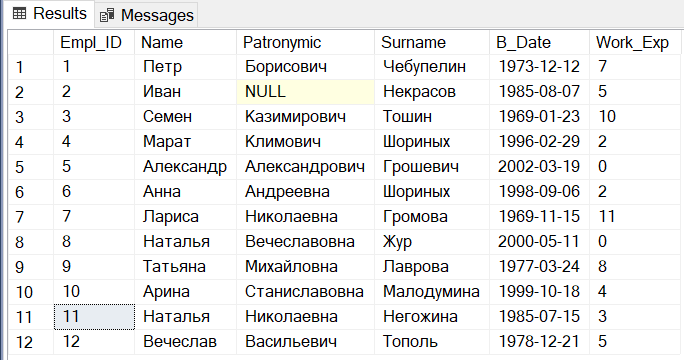


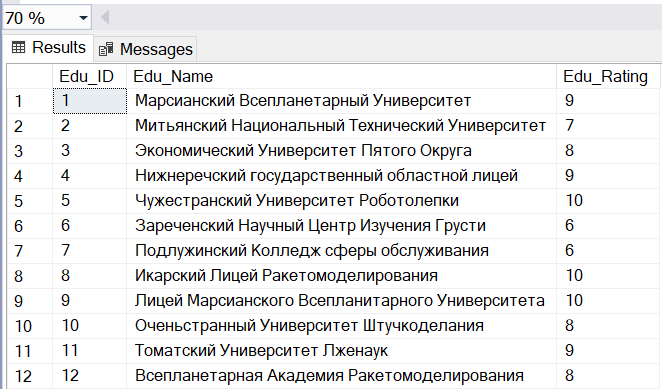
5. Переименуйте одну из таблиц вашей БД

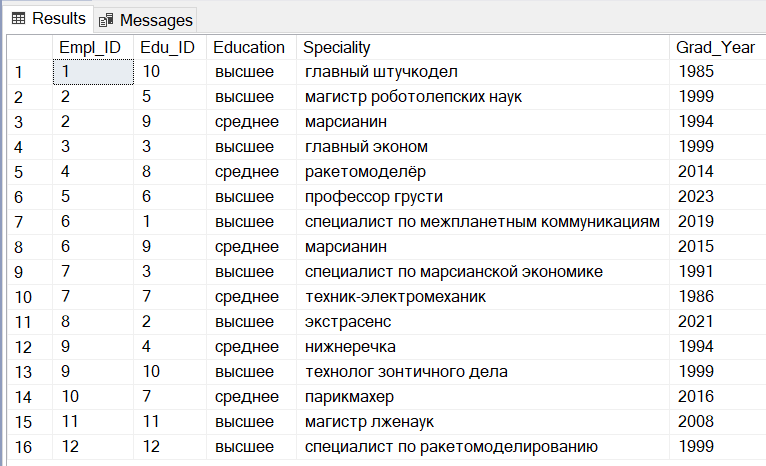


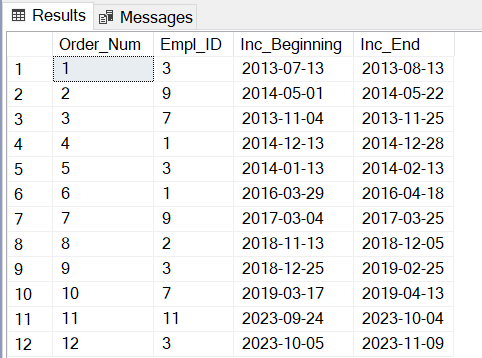
6. Добавьте по две записи в каждую из таблиц вашей БД

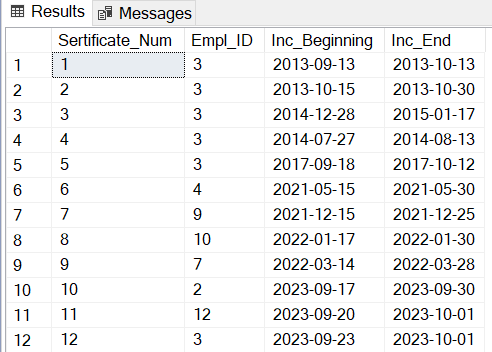






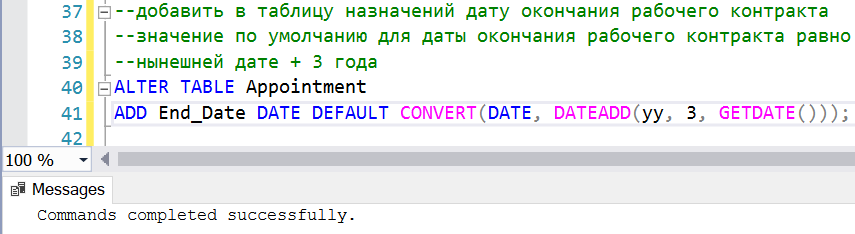




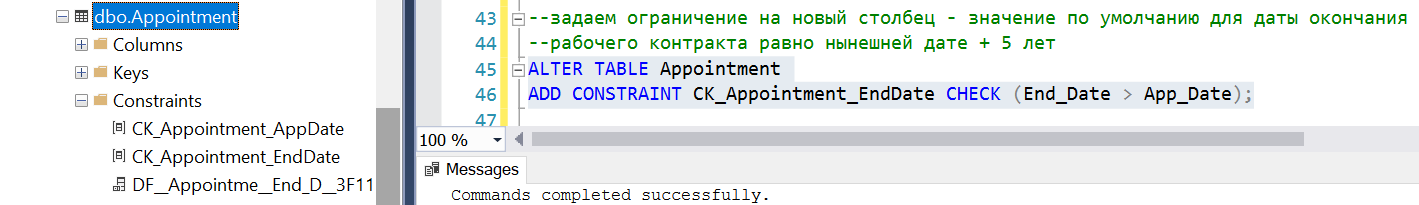


7. Измените одну из таблиц вашей БД:

• добавьте по смыслу один столбец (добавим в таблицу назначений дату окончания контракта)

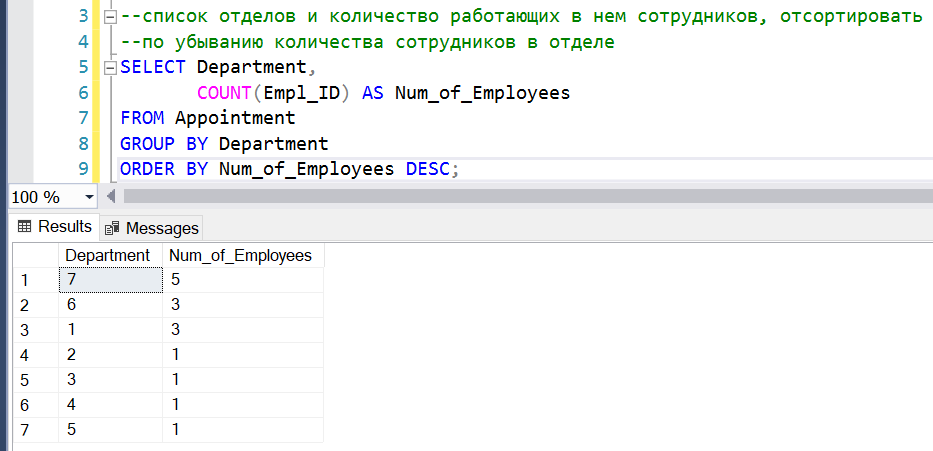


• добавьте ограничение с именем на новый введенный вами столбец (проверяем, чтобы дата окончания контракта была больше даты начала контракта)

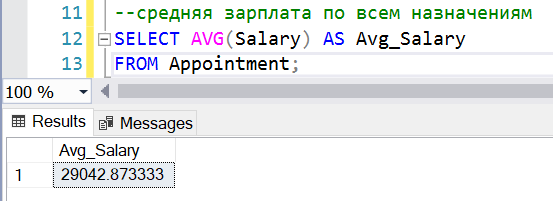


8. Напишите 5 запросов, используя агрегатные функции (SUM, MAX, MIN, AVG, COUNT)

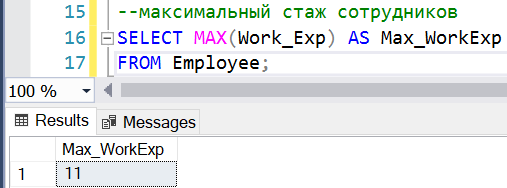
1) Вывести список отделов и количество работающих в нем сотрудников, отсортировать по убыванию количества сотрудников в отделе



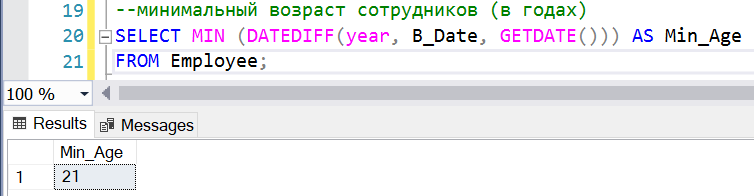
2) Вывести среднюю зарплату по всем назначениям



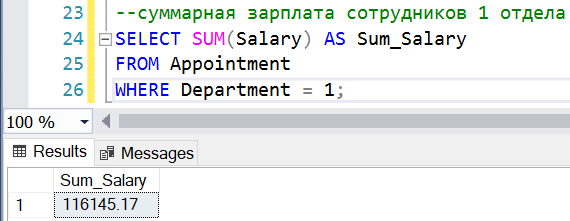
3) Вывести максимальный стаж сотрудников



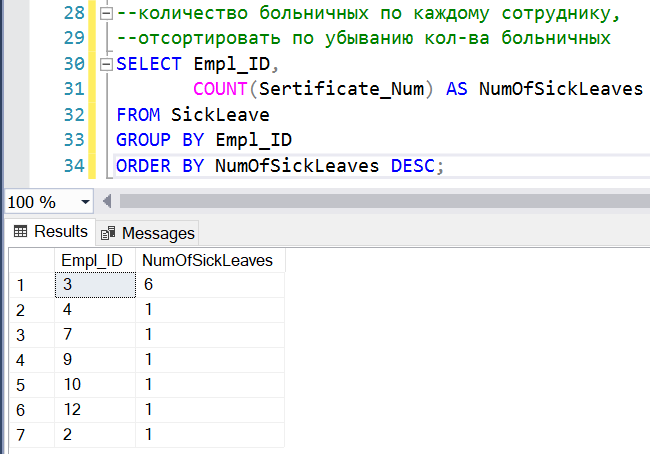
4) Вывести минимальный возраст сотрудников (в годах)



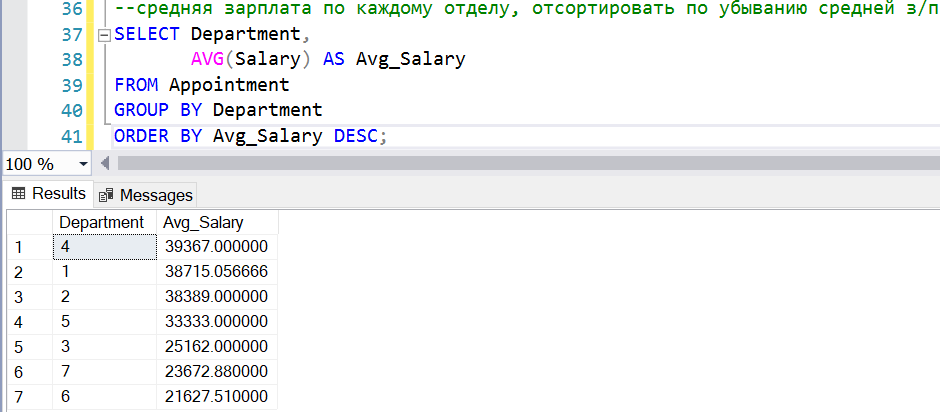
5) Вывести суммарную зарплату сотрудников 1 отдела



9. Написать 2 запроса на группировку данных, используя оператор GROUP BY1) Вывести количество больничных по каждому сотруднику, отсортировать по убыванию количества больничных

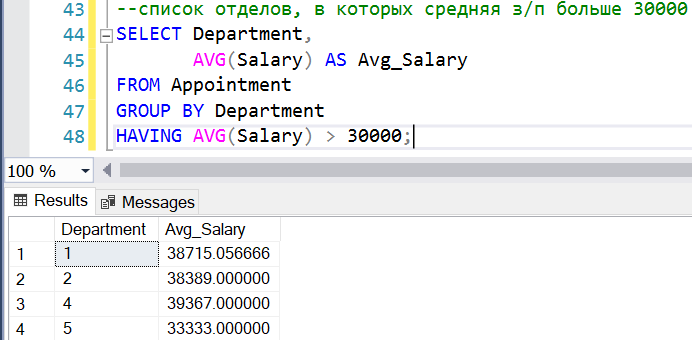


2) Вывести среднюю зарплату по каждому отделу, отсортировать по убыванию средней з/п

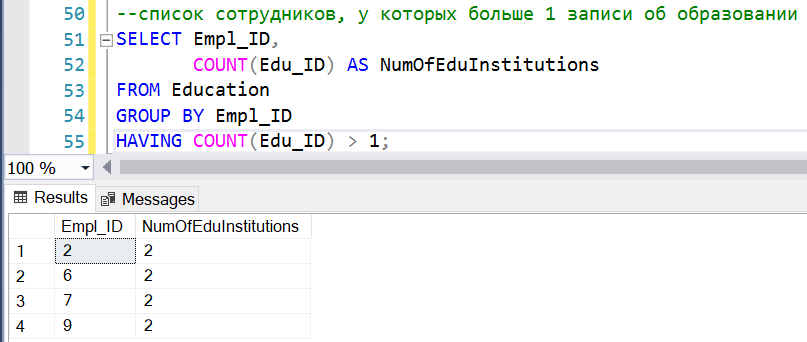


10. Написать 2 запроса на фильтрацию групп, используя HAVING.

1) Вывести список отделов, в которых средняя з/п больше 30000

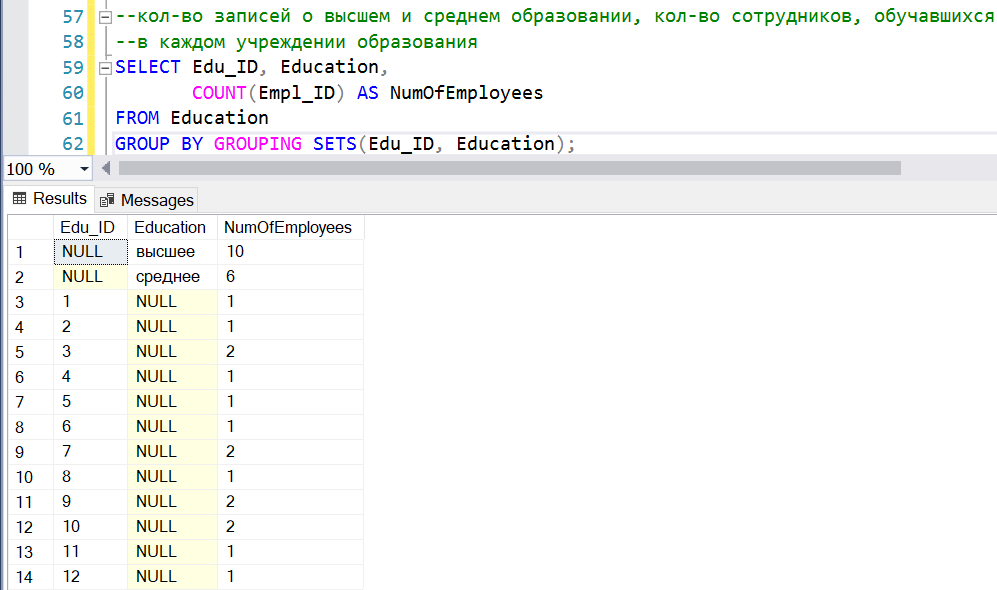


2) Вывести список сотрудников, у которых больше 1 записи об образовании

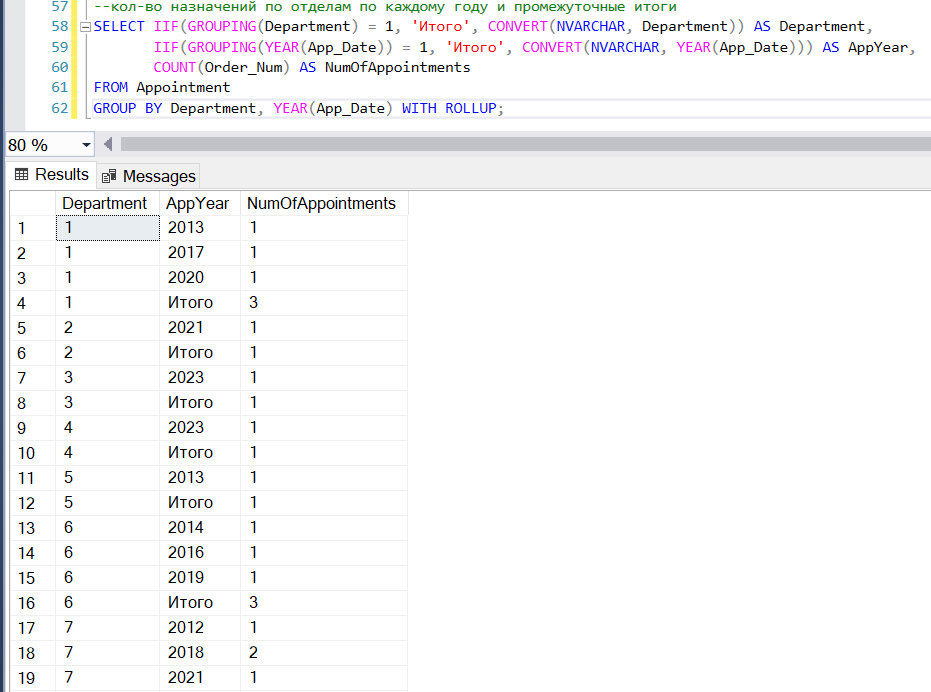


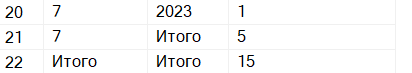
11. Написать запросы на расширения SQL Server для группировки (ROLLUP, CUBE, GROUPING SETS, OVER)

1) Вывести количество записей о высшем и среднем образовании, кол-во сотрудников, обучавшихся в каждом учреждении образования

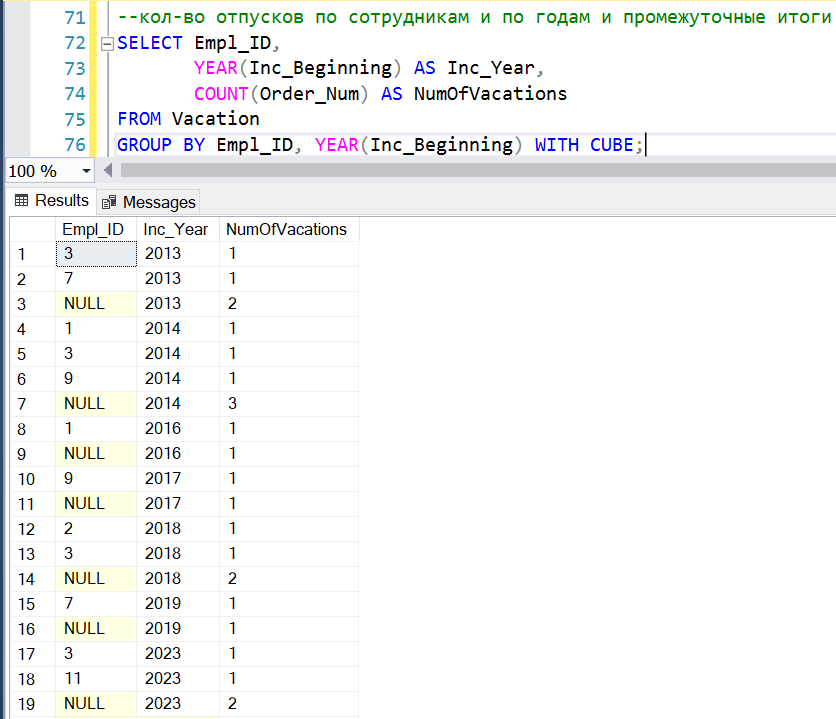


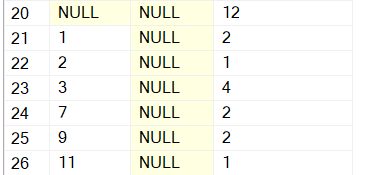
2) Вывести количество назначений по отделам по каждому году и промежуточные итоги





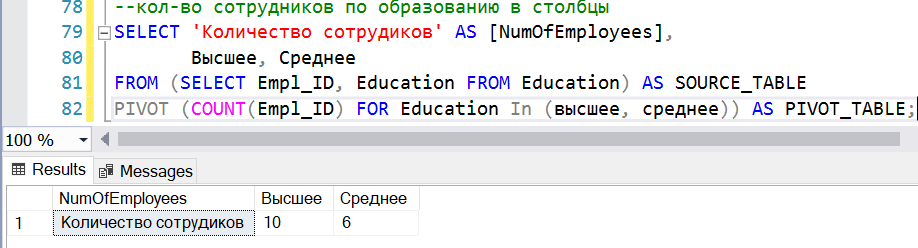
3) Вывести количество отпусков по сотрудникам и по годам и промежуточные итоги



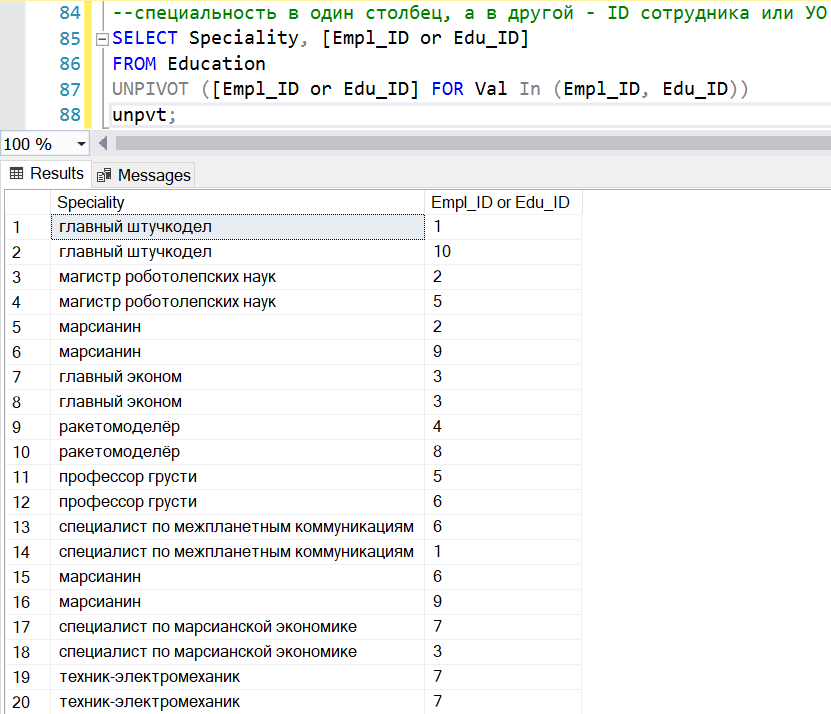


12. Написать запрос на разворачивание данных (PIVOT), отмена разворачивания (UNPIVOT).

1) Вывести количество сотрудников по образованию в столбцы



2) Вывести специальность в один столбец, а в другой - ID сотрудника или УО

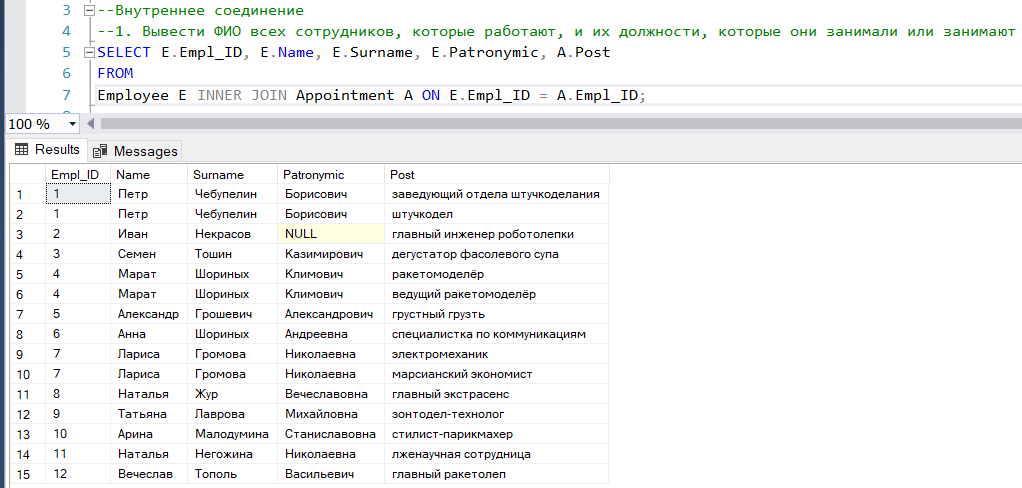
****

# **Дополнение к лабораторным работам 5 – 6**

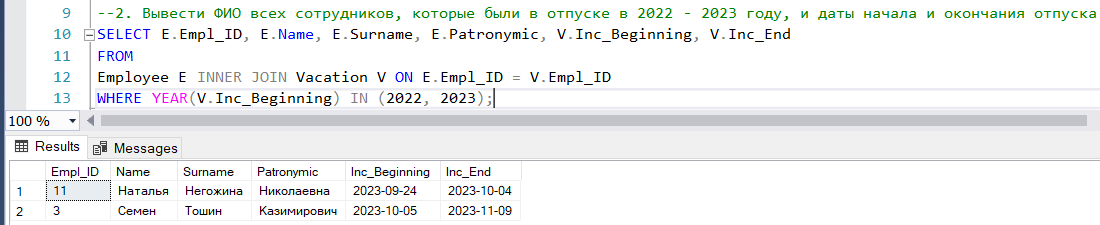
13. Написать по два запроса на каждое соединение таблиц: внутреннее; внешнее левое; внешнее правое; полное внешнее соединение; перекрестное соединение.

1. Внутреннее соединение

1.1 Вывести ФИО всех сотрудников, которые работают, и их должности, которые они занимали или занимают

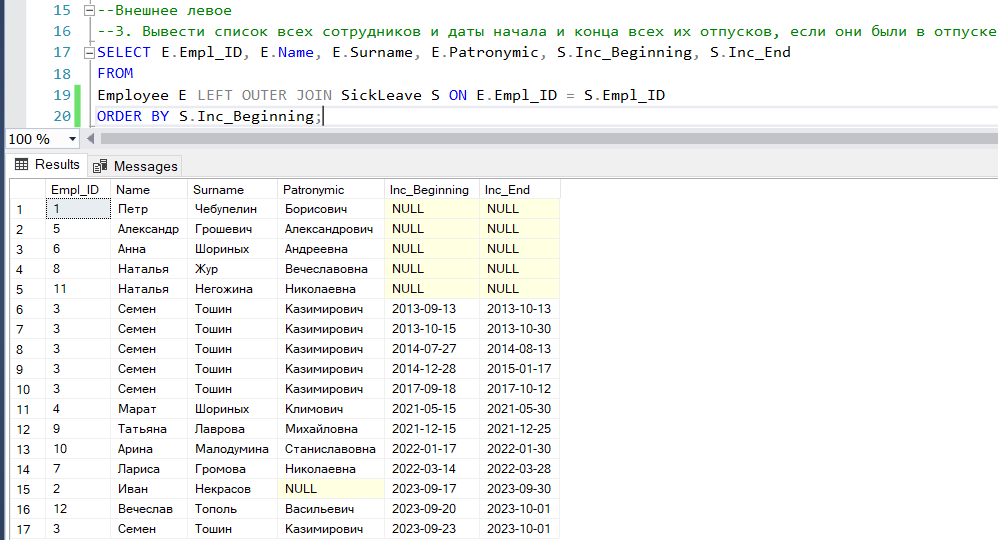


1.2 Вывести ФИО всех сотрудников, которые были в отпуске в 2022 - 2023 году, и даты начала и окончания отпуска

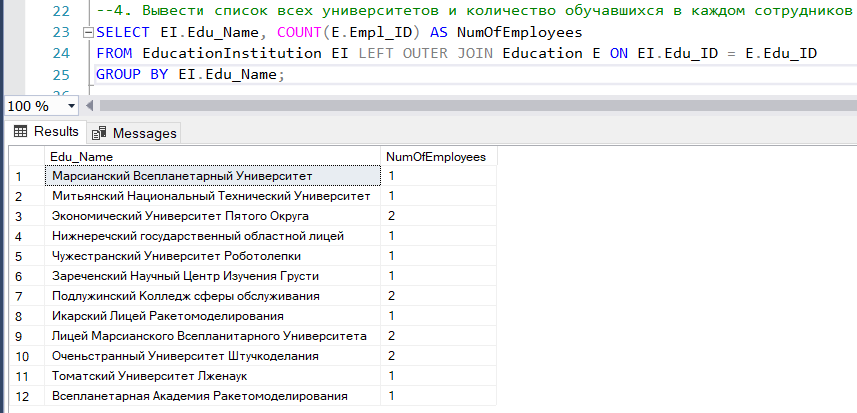


2. Внешнее левое

2.1 Вывести список всех сотрудников и даты начала и конца всех их отпусков, если они были в отпуске

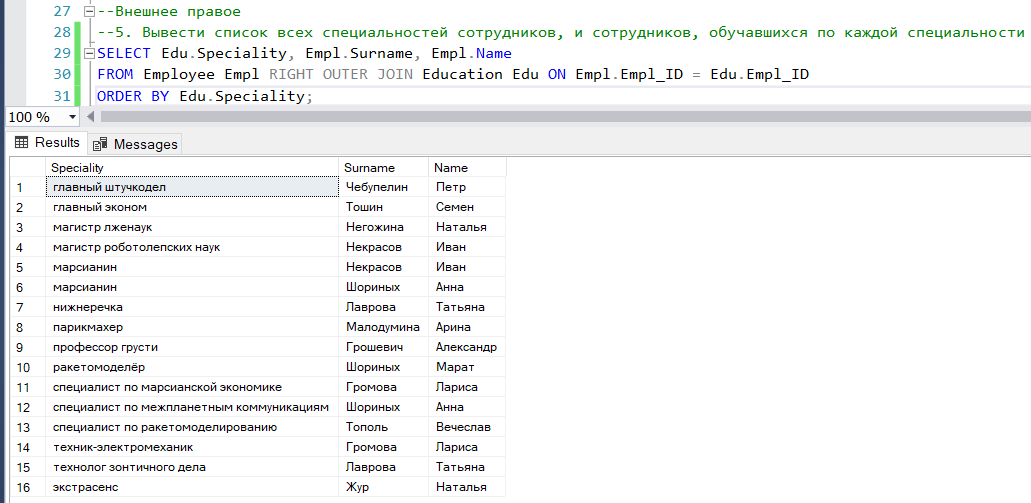


2.2 Вывести список всех университетов и количество обучавшихся в каждом сотрудников

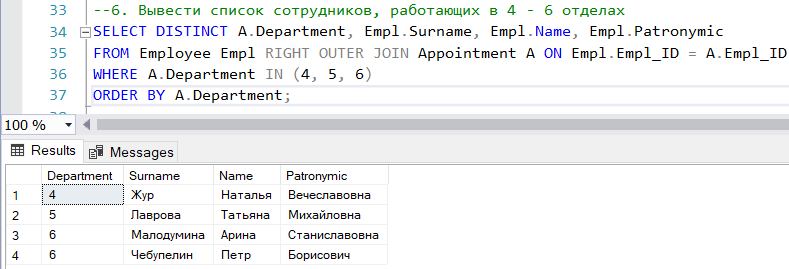


3. Внешнее правое

3.1 Вывести список всех специальностей сотрудников, и сотрудников, обучавшихся по каждой специальности

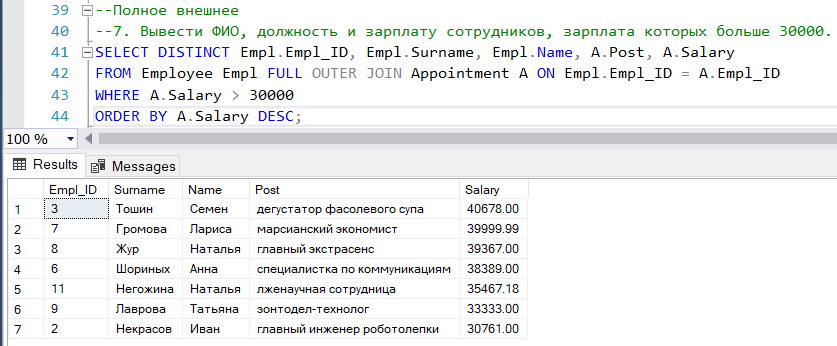


3.2 Вывести список сотрудников, работающих в 4 - 6 отделах

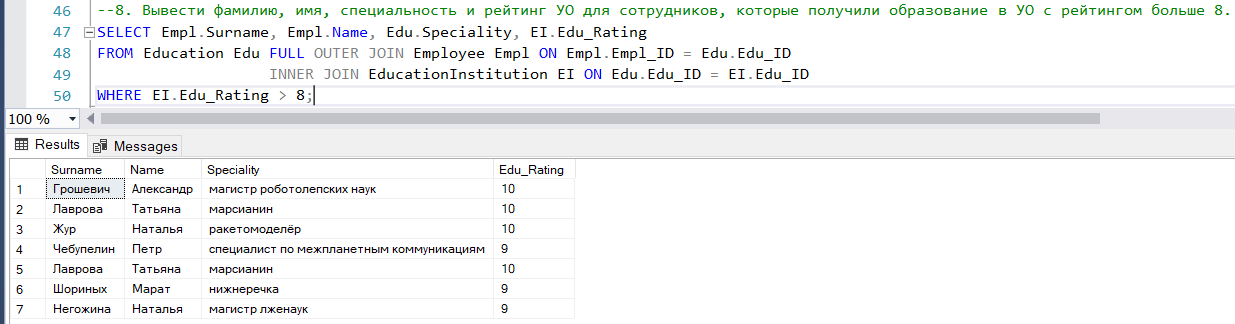


4. Полное внешнее

4.1 Вывести ФИО, должность и зарплату сотрудников, зарплата которых больше 30000.

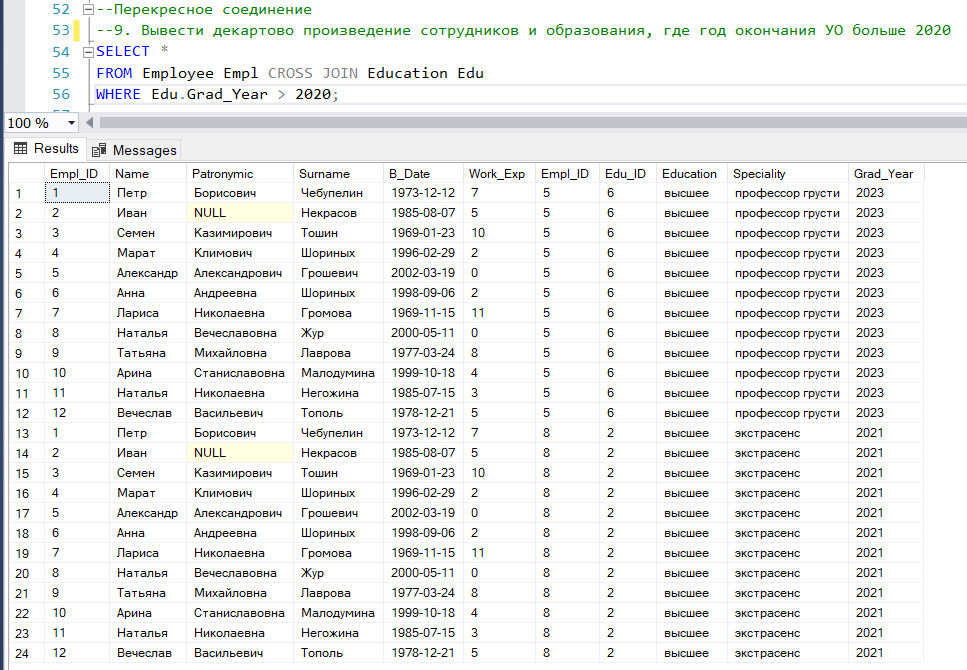


4.2 Вывести фамилию, имя, специальность и рейтинг УО для сотрудников, которые получили образование в УО с рейтингом больше 8.

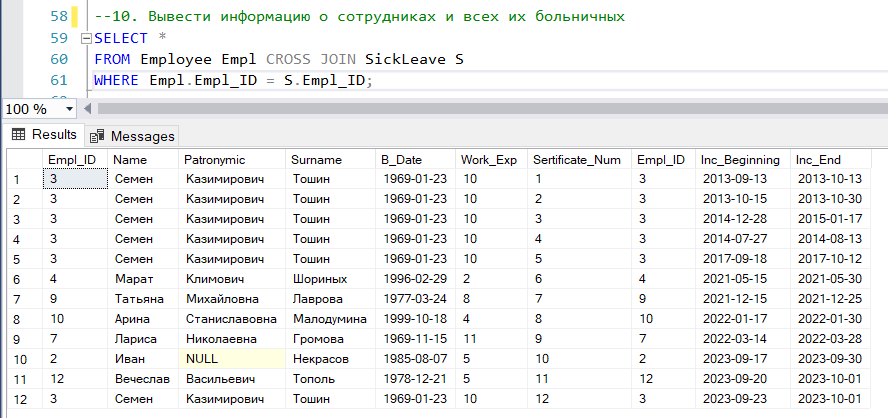


5. Перекресное соединение

5.1 Вывести декартово произведение сотрудников и образования, где год окончания УО больше 2020



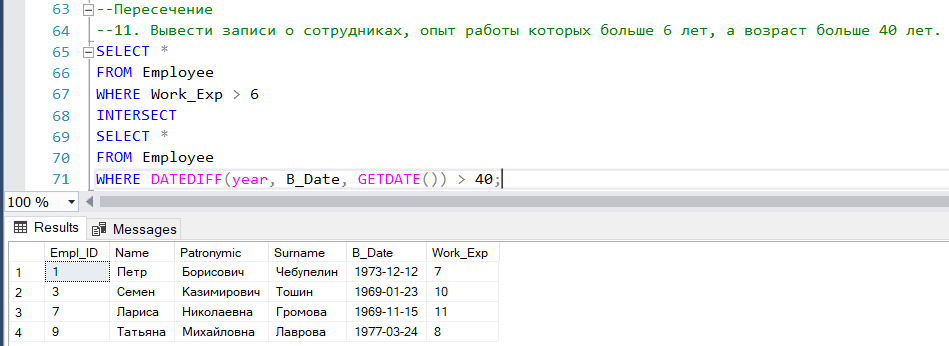
5.2 Вывести информацию о сотрудниках и всех их больничных



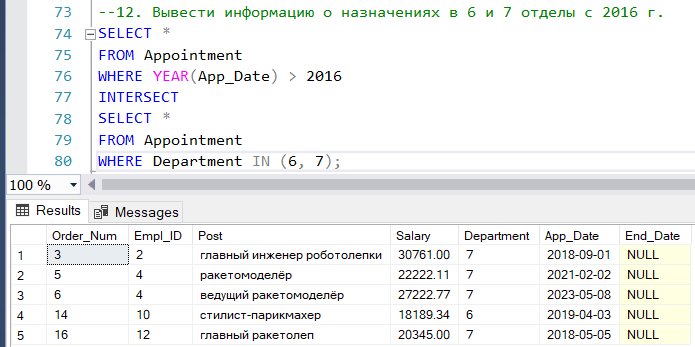
14. Написать по два запроса на пересечение, разность, объединение.

1. Пересечение

1.1 Вывести записи о сотрудниках, опыт работы которых больше 6 лет, а возраст больше 40 лет.

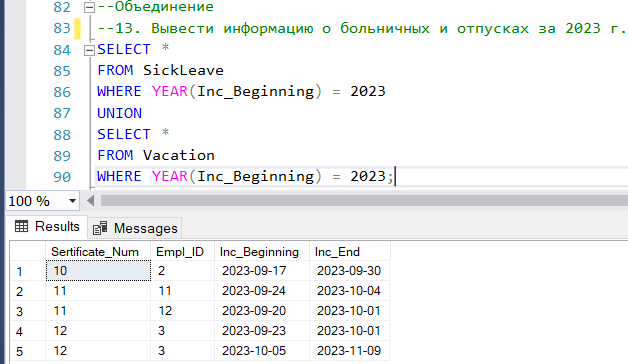


1.2 Вывести информацию о назначениях в 6 и 7 отделы с 2016 г.

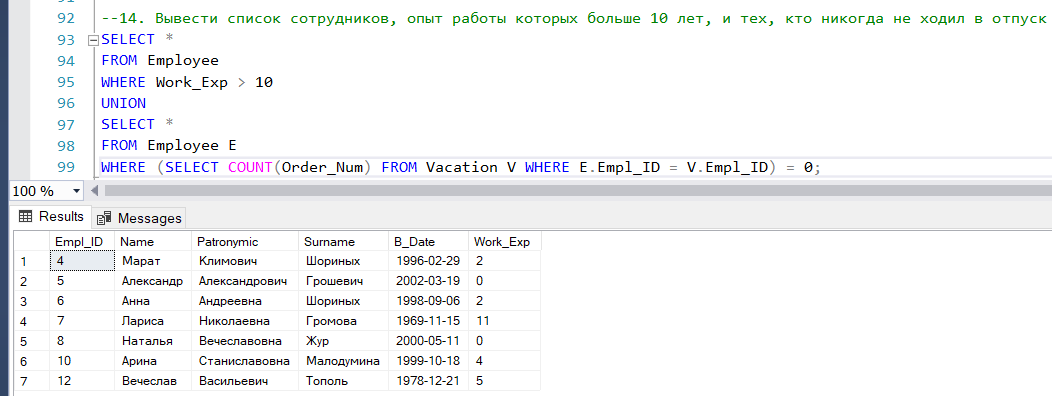


2. Объединение

2.1 Вывести информацию о больничных и отпусках за 2023 г.

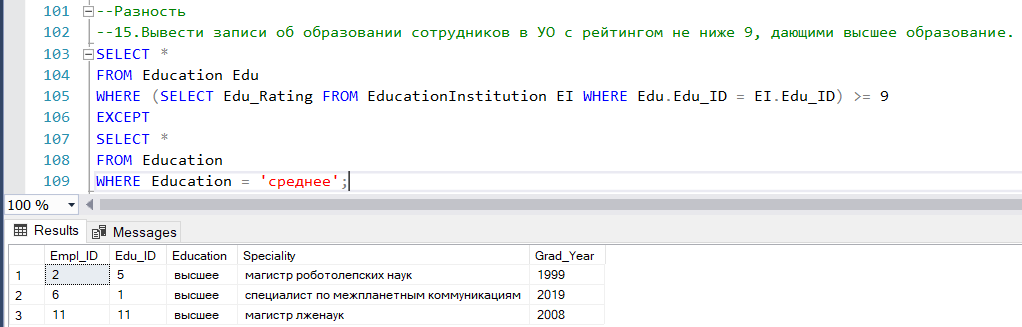


2.2 Вывести список сотрудников, опыт работы которых больше 10 лет, и тех, кто никогда не ходил в отпуск

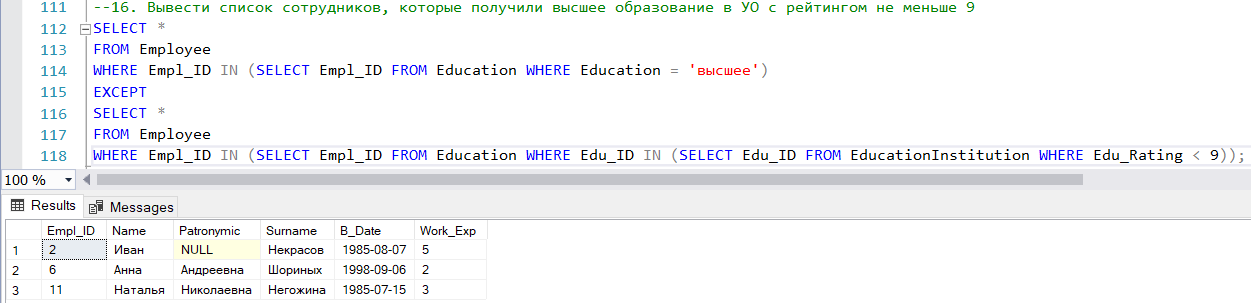


3. Разность

3.1 Вывести записи об образовании сотрудников в УО с рейтингом не ниже 9, дающими высшее образование.

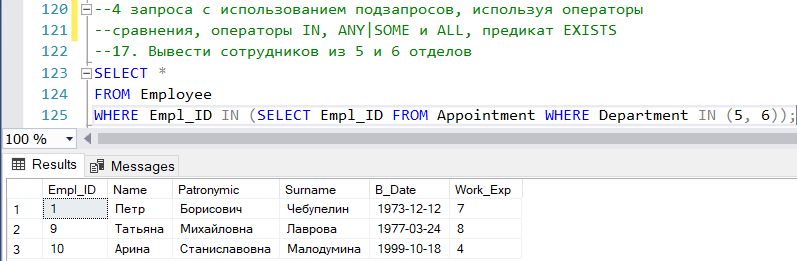


3.2 Вывести список сотрудников, которые получили высшее образование в УО с рейтингом не меньше 9

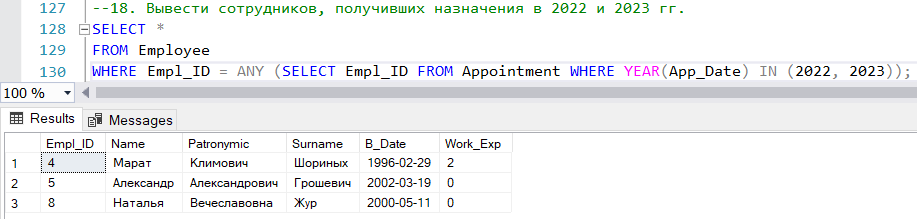


15. Написать 4 запроса с использованием подзапросов, используя операторы сравнения, операторы IN, ANY|SOME и ALL, предикат EXISTS

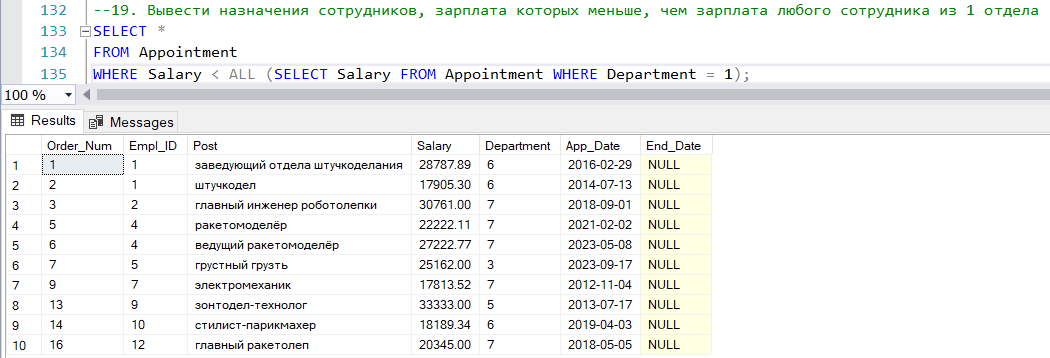
1. Вывести сотрудников из 5 и 6 отделов



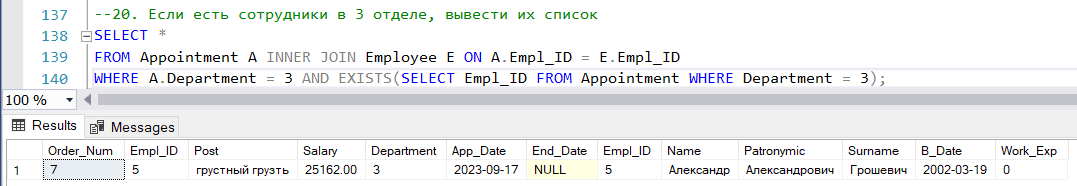
2. Вывести сотрудников, получивших назначения в 2022 и 2023 гг.



3. Вывести назначения сотрудников, зарплата которых меньше, чем зарплата любого сотрудника из 1 отдела



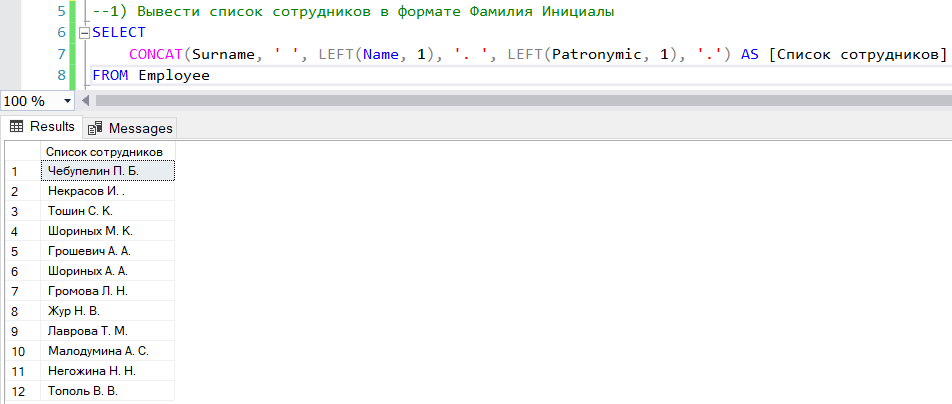
4. Если есть сотрудники в 3 отделе, вывести их список.



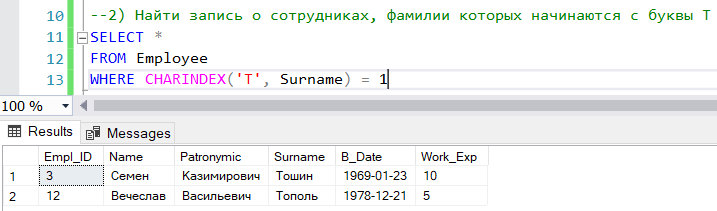
# **Дополнение к лабораторным работам 7 - 9**

1. Написать 4 запроса по строковым функциям

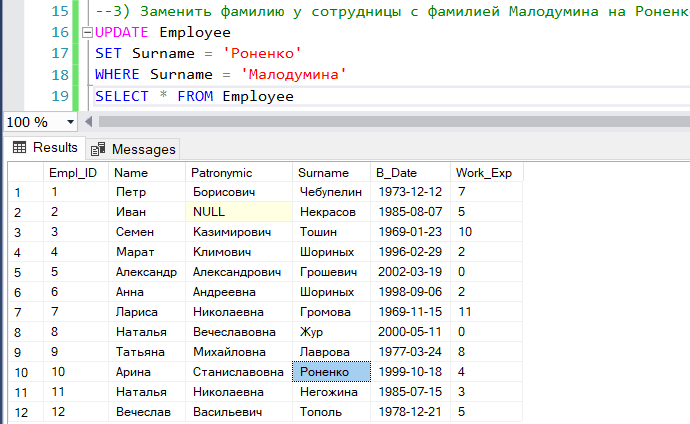
1) Вывести список сотрудников в формате Фамилия Инициалы



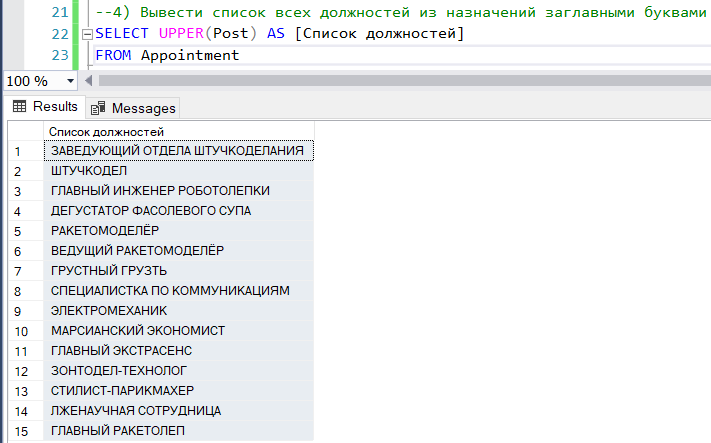
2) Найти запись о сотрудниках, фамилии которых начинаются с буквы Т



3) Заменить фамилию у сотрудницы с фамилией Малодумина на Роненко

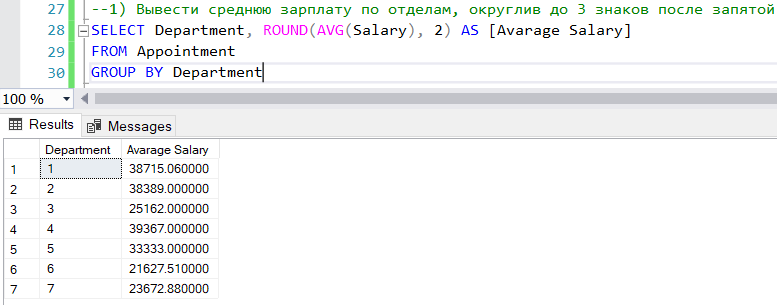


4) Вывести список всех должностей из назначений заглавными буквами

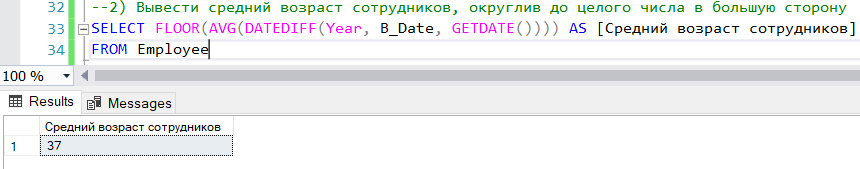


2. Написать 3 запроса по числовым функциям

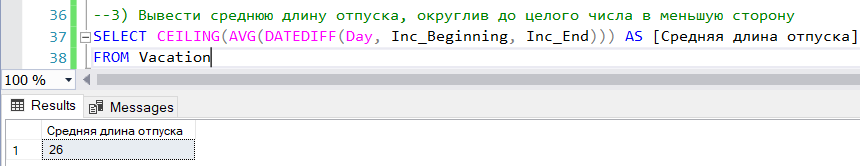
1) Вывести среднюю зарплату по отделам, округлив до 3 знаков после запятой



2) Вывести средний возраст сотрудников, округлив до целого числа в большую сторону

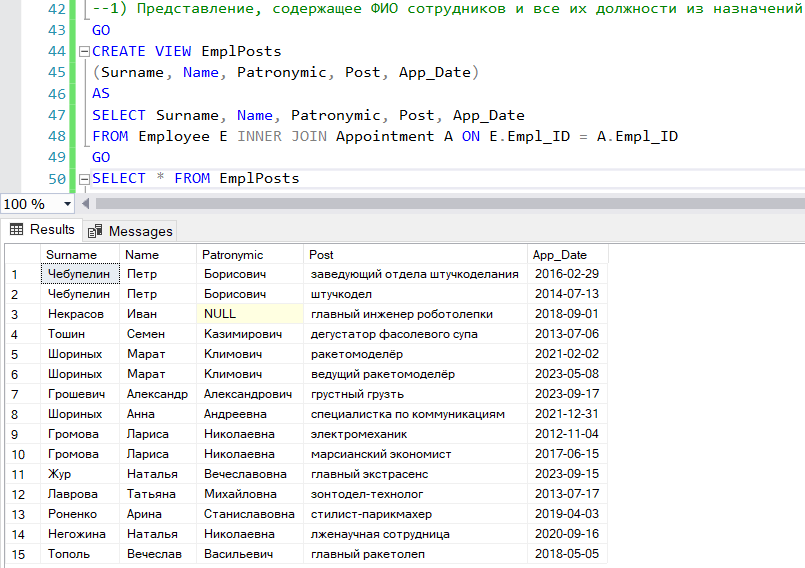


3) Вывести среднюю длину отпуска, округлив до целого числа в меньшую сторону

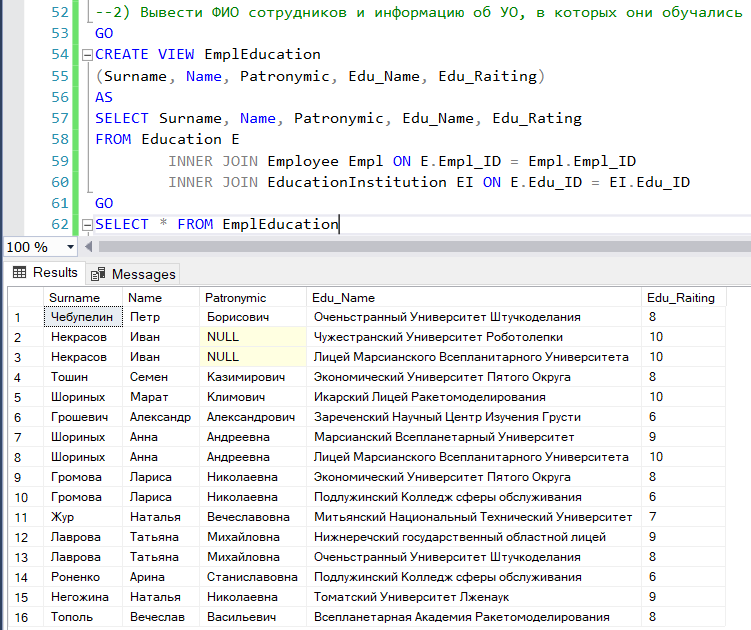


3. Создайте 5 представлений по своей БД

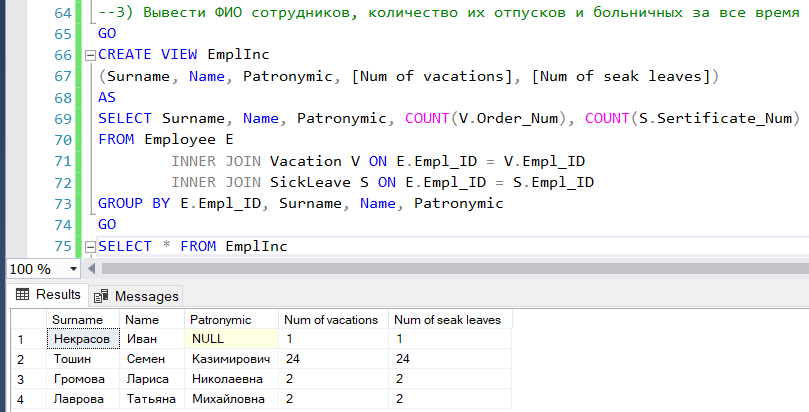
1) Представление, содержащее ФИО сотрудников и все их должности из назначений



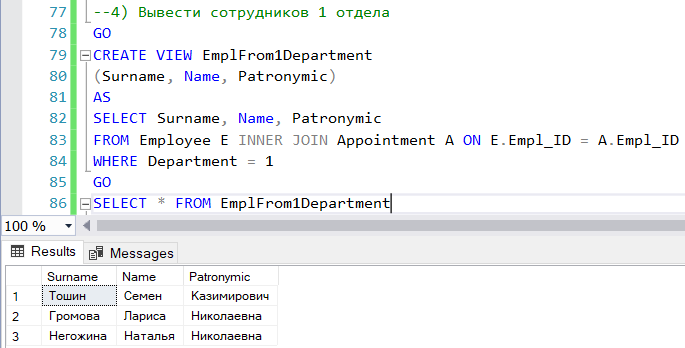
2) Вывести ФИО сотрудников и информацию об УО, в которых они обучались



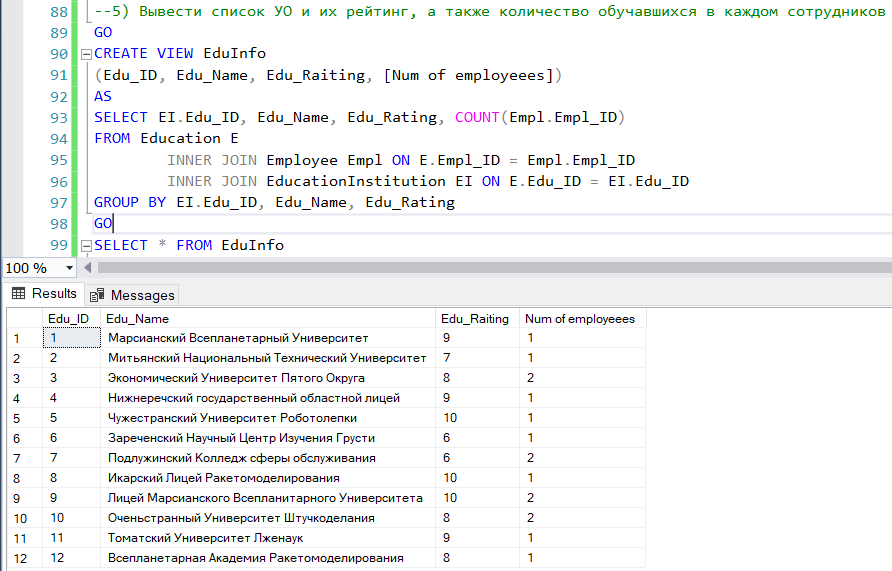
3) Вывести ФИО сотрудников, количество их отпусков и больничных за все время



4) Вывести сотрудников 1 отдела

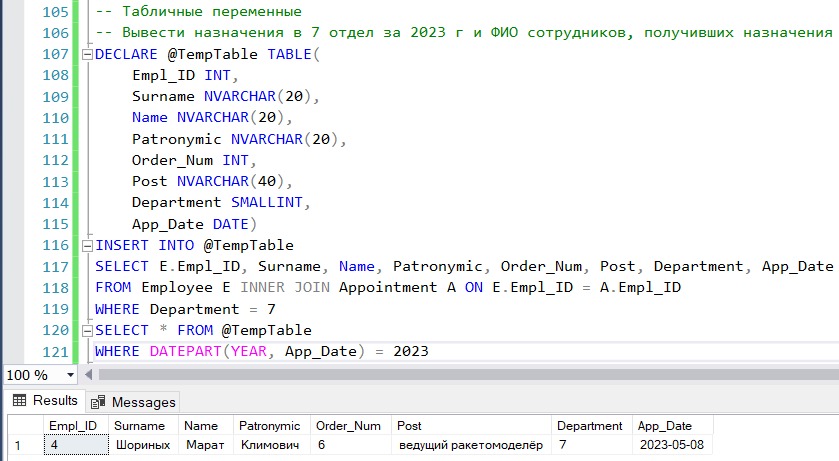


5) Вывести список УО и их рейтинг, а также количество обучавшихся в каждом сотрудников

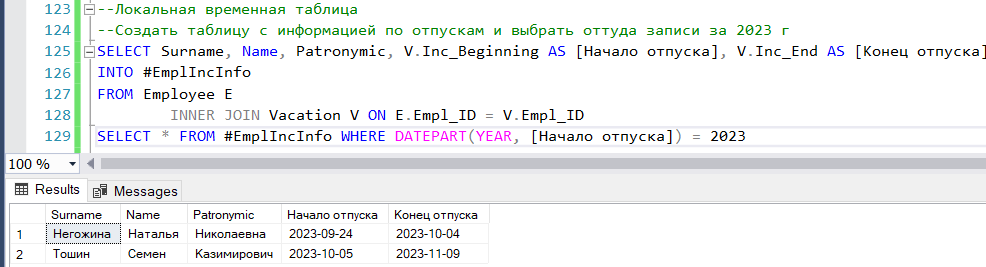


4. Покажите применение табличных переменных, временные локальных и глобальных таблиц, а так же обобщенных табличных выражений.

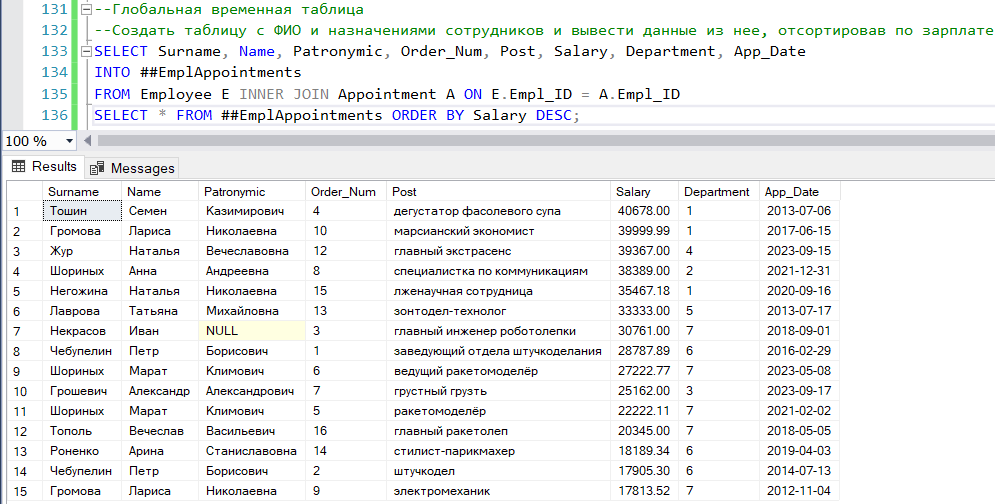
1) Вывести назначения в 7 отдел за 2023 г и ФИО сотрудников, получивших назначения



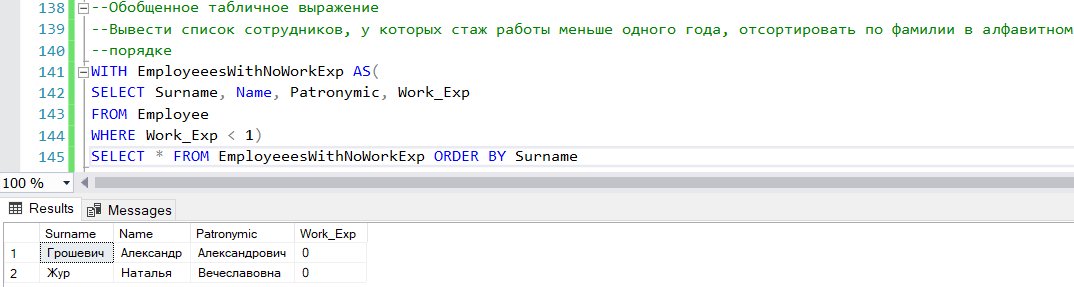
2) Создать таблицу с информацией по отпускам и выбрать оттуда записи за 2023 г



3) Создать таблицу с ФИО и назначениями сотрудников и вывести данные из нее, отсортировав по зарплате

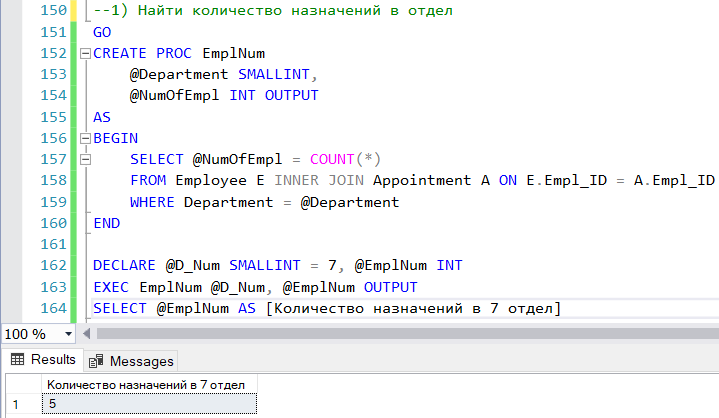


4) Вывести список сотрудников, у которых стаж работы меньше одного года, отсортировать по фамилии в алфавитном порядке

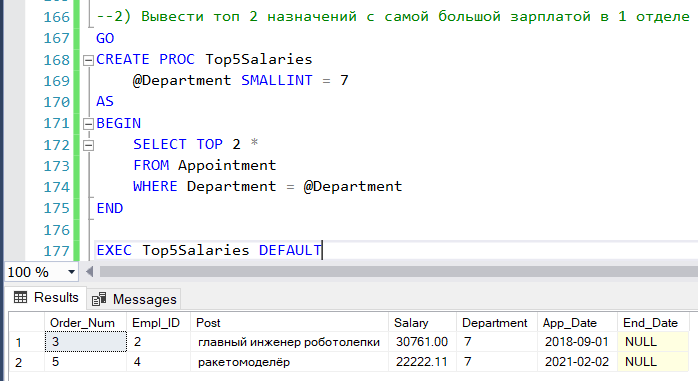


5. Создать 2 хранимые процедуры, одна из которых будет иметь выходные параметры.

1) Найти количество назначений в отдел

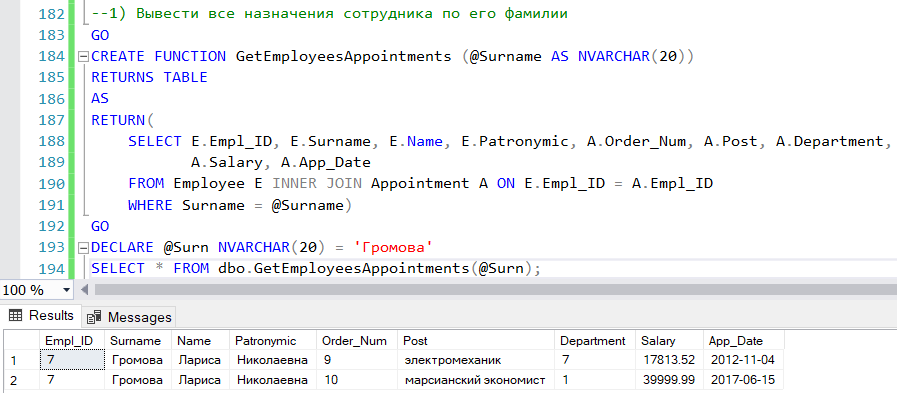


2) Вывести топ 2 назначений с самой большой зарплатой в 1 отделе

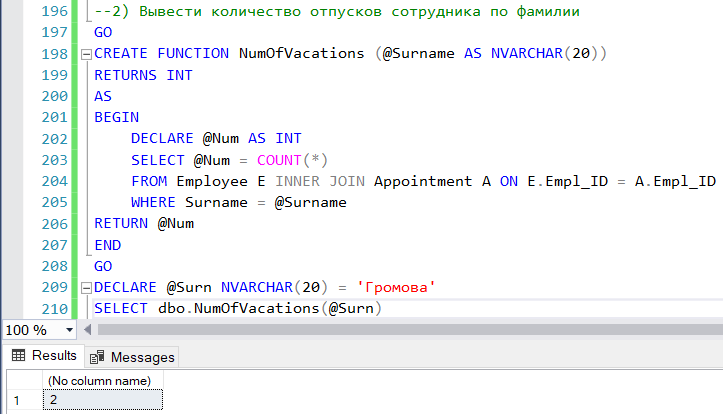


6. Создать 2 определяемые пользователем функции, одна из которых скалярная функция, другая возвращает табличное значение

1) Вывести все назначения сотрудника по его фамилии

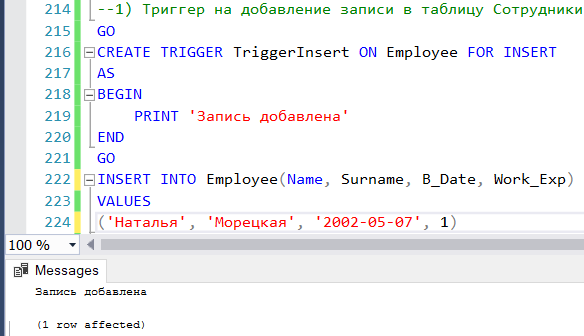


2) Вывести количество отпусков сотрудника по фамилии



7. Создайте два триггера: триггер AFTER и триггер INSTEAD OF

1) Триггер на добавление записи в таблицу Сотрудники



2) Триггер на удаление записи из таблицы Сотрудники

