МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное   
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет   
имени академика С.П. Королева»

(Самарский университет)

Институт информатики, математики и электроники

Факультет информатики  
Кафедра суперкомпьютеров и общей информатики

**Отчет по лабораторной работе № 1**

Дисциплина: “Databases in Enterprise Systems” («Корпоративные базы данных»)

Выполнил: Мелешенко И. С.

Группа: 6133-010402D

Самара 2022

**СОДЕРЖАНИЕ**

[**Задание на лабораторную работу №1** 3](#_Toc103543528)

[**Задание 1. Выбор и описание ER-модели** 4](#_Toc103543529)

[**Задание 2. Разработать ER модель, включающую минимум 5-6 сущностей и типы связей: 1-N, N:M, 1-1.** 5](#_Toc103543530)

[**2.1 Описание сущностей.** 5](#_Toc103543531)

[**2.1.1 Сущность ActiveAircraft** 5](#_Toc103543532)

[**2.1.2 Сущность InformationAircraft** 6](#_Toc103543533)

[**2.1.3 Сущность Airport** 7](#_Toc103543534)

[**2.1.4 Сущность Runway** 8](#_Toc103543535)

[**2.1.5 Сущность Aircompany** 9](#_Toc103543536)

[**2.1.6 Сущность Passenger** 9](#_Toc103543537)

[**2.1.7 Сущность ClientAircompany** 10](#_Toc103543538)

[**2.2 Описание связей между сущностями** 11](#_Toc103543539)

[**2.2.1 Связь типа «Один к одному»** 11](#_Toc103543540)

[**2.2.2 Связь типа «Один к многим»** 12](#_Toc103543541)

[**2.2.3 Связь типа «Многие ко многим»** 12](#_Toc103543542)

[**Задание 3 - 4. Создать базу данных по модели в СУБД PostgreSQL. Определить индексы, уникальные индексы.** 13](#_Toc103543543)

[**Задание 3.1 Создание базы данных.** 13](#_Toc103543544)

[**Задание 3.2 Создание схемы ER-модели данных.** 13](#_Toc103543545)

[**Задание 4.1 Установка индексов.** 14](#_Toc103543546)

[**5. Разработать типовые запросы к СУБД на языке SQL. Получение списков данных. Агрегация. Поиск.** 15](#_Toc103543547)

# **Задание на лабораторную работу №1**

1. Выбрать предметную область

2. Разработать ER модель, включающую минимум 5-6 сущностей и типы связей: 1-N, N:M, 1-1.

3. Создать базу данных по модели в СУБД PostgreSQL.

4. Определить индексы, уникальные индексы.

5. Разработать типовые запросы к СУБД на языке SQL. Получение списков данных. Агрегация. Поиск.

6. Разработайте хранимые процедуры на языке PL/pgSQL для генерации случайных данных для базы данных.

7. Сгенерируйте тестовые данные при помощи разработанных процедур.

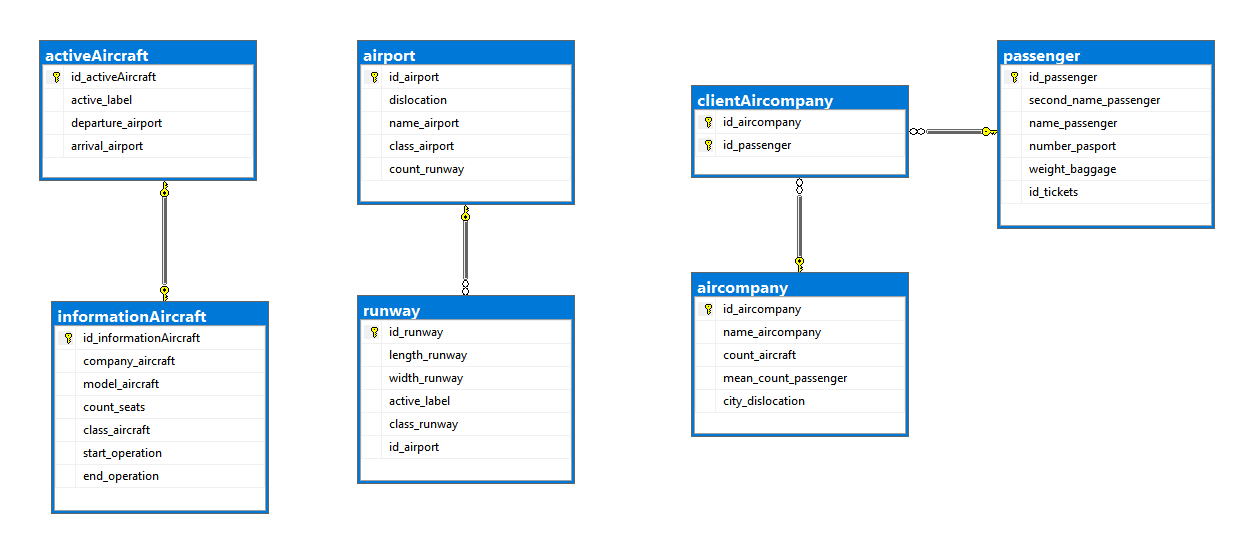
8. Протестируйте работу запросов на больших объёмах данных (Порядка 1 миллиона записей в основных таблицах).

9. Измените конфигурацию сервера PostgreSQL для достижения лучшей производительности на самых медленных запросах. Оптимизируйте схему БД и запросы для достижения лучшей производительности.

Пункты 6-7 допустимо реализовывать другими способами без PL/pgSQL

# **Задание 1. Выбор и описание ER-модели**

Для работы с базой данных была придумана ER-модель на основе бизнес-модели авиационных перевозок. Данная модель состоит из двух видов сущностей и трех типов связей. Схема модели представлена на рисунке 1.

  
Рисунок 1 – Схема ER-модели бизнес-процесса

**Задание 2. Разработать ER модель, включающую минимум 5-6 сущностей и типы связей: 1-N, N:M, 1-1.**

В моей модели реализовано 7 сущностей, которые описывают те или иные объекты/субъекты авиационных перевозок. Рассмотрим их по подробнее.

**2.1 Описание сущностей.**

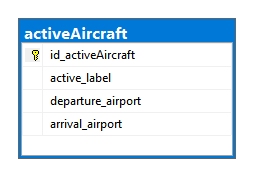
В качестве сущностей ER-модели выбраны те или иные реально существующие объекты модели авиационных перевозок. Описываемая модель включает следующие сущности:

* ActiveAircraft
* InformationAircraft
* Airport
* Runway
* Aircompany
* Passenger
* ClientAircompany

Далее рассмотрим каждую более подробно.

**2.1.1 Сущность ActiveAircraft**

Данная сущность необходима для описания действующий самолет. Сущность модели представлена на рисунке 2.

  
Рисунок 2 – Сущность ActiveAircraft

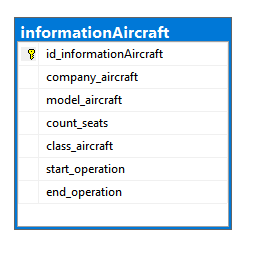
Рассмотрим их описание и характеристику.

Таблица 1 – Описание полей сущности ActiveAircraft

| Название поля | Тип данных | Описание | Ключ |
| --- | --- | --- | --- |
| id\_activeAircraft | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key |
| active\_label | bool | Является маркером действительности самолета. Если выбрано значение true, то самолет действителен может летать, вследствие чего должен иметь место вылета и место прилета. Если выбрано значение false самолет считается не действительным и летать не может, а в места вылета и прилета имеют значения NULL. | ––– |
| departure\_airport | varchar()  string | Является полем, содержащим информацию о месте вылета (город) самолета, в случае если он действителен. | ––– |
| arrival\_airport | varchar()  string | Является полем, содержащим информацию о месте прилета (город) самолета, в случае если он действителен. | ––– |

**2.1.2 Сущность InformationAircraft**

Данная сущность необходима для более полного описания действующего самолета и представлена на рисунке 3.

  
Рисунок 3 – Сущность InformationAircraft

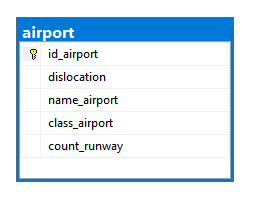
Рассмотрим их описание и характеристику.

Таблица 2 – Описание полей сущности InformationAircraft

| Название поля | Тип данных | Описание поля | Ключ |
| --- | --- | --- | --- |
| id\_informationAircraft | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key/Foreign Key |
| company\_aircraft | VARCHAR() String | Является полем, содержащим информацию о производителе самолета. | ––– |
| model\_aircraft | VARCHAR() String | Является полем, содержащим информацию о модели самолета. | ––– |
| count\_seats | int | Является полем, содержащим информацию о количестве мест в самолете. | ––– |
| class\_aircraft | VARCHAR() String | Является полем, содержащим информацию о классе самолета. | ––– |
| start\_operation | Date | Является полем, содержащим информацию о дате начала эксплуатации самолета. | ––– |
| end\_operation | Date | Является полем, содержащим информацию о дате окончания эксплуатации самолета. | ––– |

**2.1.3 Сущность Airport**

Данная сущность необходима для описания аэропорта и представлена на рисунке 4.

  
Рисунок 4 – Сущность Airport

Рассмотрим их описание и характеристику.

Таблица 3 – Описание полей сущности Airport

| Название поля | Тип данных | Описание | Ключ |
| --- | --- | --- | --- |
| id\_airport | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key |
| dislocation | VARCHAR()  string | Является полем, содержащим информацию  о городе дислокации аэропорта. | ––– |
| name\_airport | VARCHAR()  string | Является полем, содержащим информацию  о имени аэропорта. | ––– |
| class\_airport | VARCHAR()  string | Является полем, содержащим информацию  о типе/классе аэропорта. | ––– |
| count\_runway | int | Является полем, содержащим информацию  о количестве взлетных полос в аэропорту. | ––– |

**2.1.4 Сущность Runway**

Данная сущность необходима для описания летной полосы аэропорта и представлена на рисунке 5.

  
Рисунок 5 – Сущность Runway

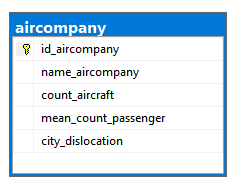
Рассмотрим их описание и характеристику.

Таблица 4 – Описание полей сущности Runway

| Название поля | Тип данных | Описание | Ключ |
| --- | --- | --- | --- |
| id\_runway | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key |
| lenght\_runway | int | Является полем, содержащим информацию  о длине взлетной полосы. | ––– |
| width\_runway | int | Является полем, содержащим информацию  о ширине взлетной полосы. | ––– |
| active\_label | bool | Является маркером действительности взлетной полосы.  Если выбрано значение true, то полоса доступна для работы,  иначе не доступна. | ––– |
| class\_runway | int | Является полем, содержащим информацию  о классе взлетной полосы. | ––– |
| id\_airport | int | Является полем, содержащим информацию  об аэропорте, в котором находится взлетная полоса. | Foreign Key |

**2.1.5 Сущность Aircompany**

Данная сущность необходима для описания авиакомпаний, осуществляющие авиаперевозки и представлена на рисунке 6.

  
Рисунок 6 – Сущность Aircompany

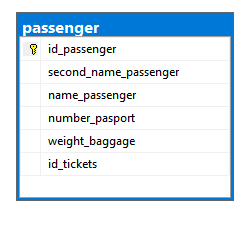
Рассмотрим их описание и характеристику.

Таблица 5 – Описание полей сущности Aircompany

|  |  |  |  |
| --- | --- | --- | --- |
| Название поля | Тип данных | Описание | Ключ |
| id\_aircompany | int | Является уникальным идентификатором, который используется для однозначной  маркировки записей в таблице. | Primary Key |
| name\_aircompany | VARCHAR()  string | Является полем, содержащим информацию о имени авиакомпании. | ––– |
| count\_aircraft | int | Является полем, содержащим информацию о количестве самолетов в компании | ––– |
| mean\_count\_passenger | int | Является полем, содержащим информацию о среднем количестве пассажиров, перевозимых компанией. | ––– |
| city\_dislocation | VARCHAR()  string | Является полем, содержащим информацию о городе дислокации авиакомпании. | ––– |

**2.1.6 Сущность Passenger**

Данная сущность необходима для описания клиентов авиакомпаний – пассажиров, и представлена на рисунке 7.

  
Рисунок 7 – Сущность Passenger

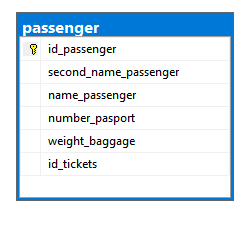
Рассмотрим их описание и характеристику.

Таблица 6 – Описание полей сущности Passenger

|  |  |  |  |
| --- | --- | --- | --- |
| Название поля | Тип данных | Описание | Ключ |
| id\_passenger | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key |
| second\_name\_passenger | VARCHAR()  String | Является полем, содержащим информацию о фамилии пассажира. | ––– |
| name\_passenger | VARCHAR()  String | Является полем, содержащим информацию о имени пассажира. | ––– |
| number\_passport | int | Является полем, содержащим информацию о номере паспорта пассажира. | ––– |
| weight\_baggage | int | Является полем, содержащим информацию о весе багажа пассажира. | ––– |
| id\_tickets | int | Является полем, содержащим информацию о номере билета пассажира. | ––– |

**2.1.7 Сущность ClientAircompany**

Данная сущность необходима для описания связи между пассажиром и авиакомпанией, и представлена на рисунке 8.

  
Рисунок 8 – Сущность Passenger

Рассмотрим их описание и характеристику.

Таблица 7 – Описание полей сущности Passenger

|  |  |  |  |
| --- | --- | --- | --- |
| Название поля | Тип данных | Описание | Ключ |
| id\_aircompany | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key  Foreign Key |
| id\_passenger | int | Является уникальным идентификатором, который используется для однозначной маркировки записей в таблице. | Primary Key  Foreign Key |

**2.2 Описание связей между сущностями**

В моей ER-модели реализовано 3 типа связей:

* *1-1 – связь один к одному,*
* *1-N – связь один ко многим,*
* *N:M – связь многие ко многим.*

Далее рассмотрим каждый тип связи, и посмотрим какие сущности каким типом связи связаны.

**2.2.1 Связь типа «Один к одному»**

|  |  |
| --- | --- |
| Тип связи | Описание |
| Один к одному | Данным типом связи, связаны 2 сущности:  ***"действующий" самолет***  и  ***информация об этом самолете,***  поскольку не может быть 2-х абсолютно одинаковых самолетов. |

**2.2.2 Связь типа «Один к многим»**

|  |  |
| --- | --- |
| Тип связи | Описание |
| Один к многим | Данным типом связи,  связаны 2 сущности:  ***аэропорт***  и  ***взлетные полосы***  поскольку аэропорт может иметь как одну так и несколько взлетных полос. |

**2.2.3 Связь типа «Многие ко многим»**

|  |  |
| --- | --- |
| Тип связи | Описание |
| Многие ко многим | Данным типом связи, связаны 2 сущности,  через промежуточную третью:  ***авиакомпания и пассажир***  связаны через  ***клиентов авиакомпаний.***  Поскольку, пассажир может быть зарегистрирован не в одной авиакомпании, а авиакомпании явно имеют более одного клиента(пассажира). |

**Задание 3 - 4. Создать базу данных по модели в СУБД PostgreSQL. Определить индексы, уникальные индексы.**

Для последующего выполнения лабораторной работы нам потребуются следующие инструменты:

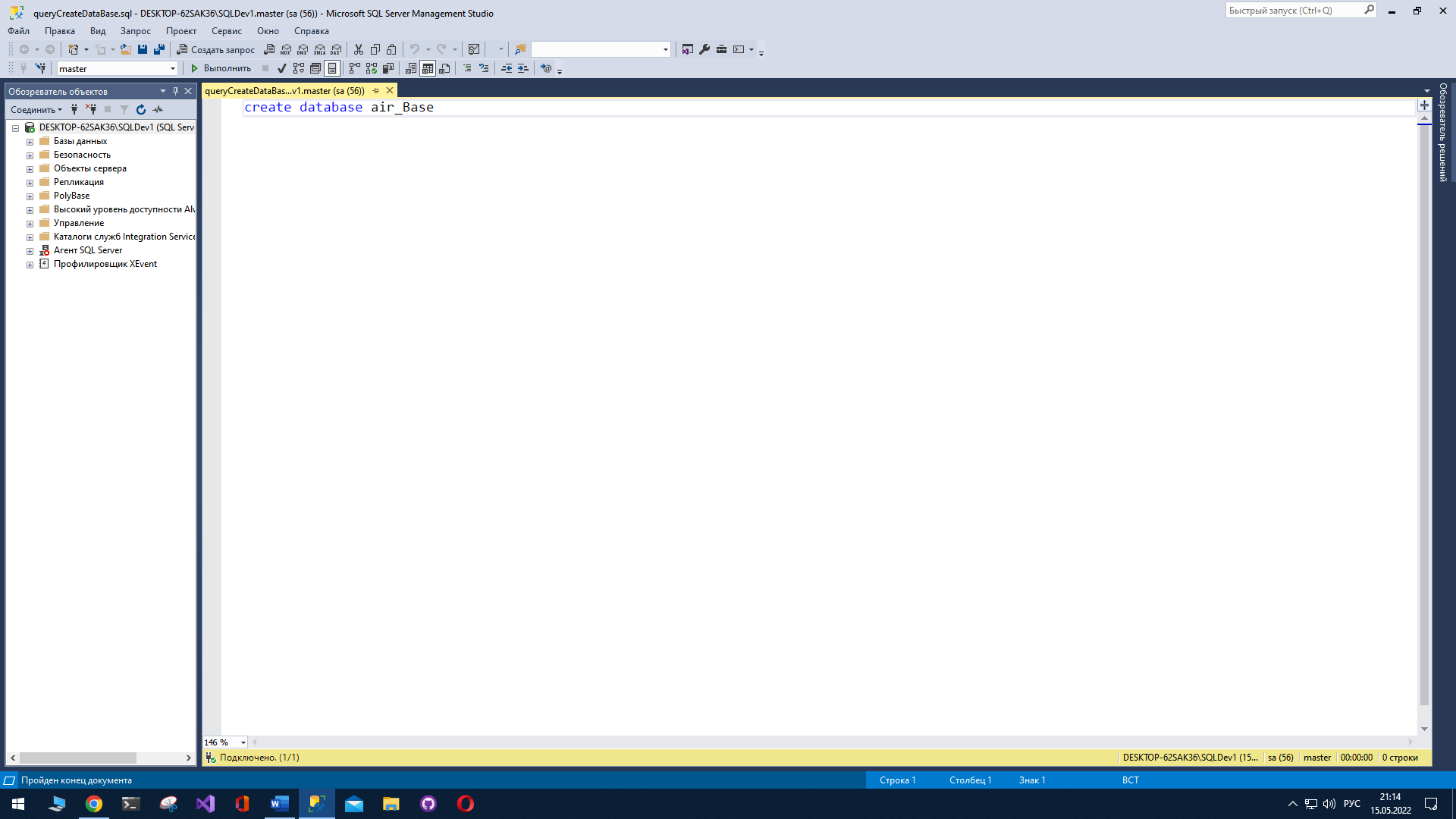
MS SQL Server 2019 Developer – SQL сервер, на котором мы и будем работать.

SQL Server Management Studio (SSMS) – утилита из Microsoft SQL Server 2019 для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server.

Далее будет показана, работа с базой данных. Скрипты, упоминаемые в данном отчете, расположены в следующем репозитории на сайте GitHub по ссылке: https://github.com/Black-Viking-63/EnterpriseDataBase.

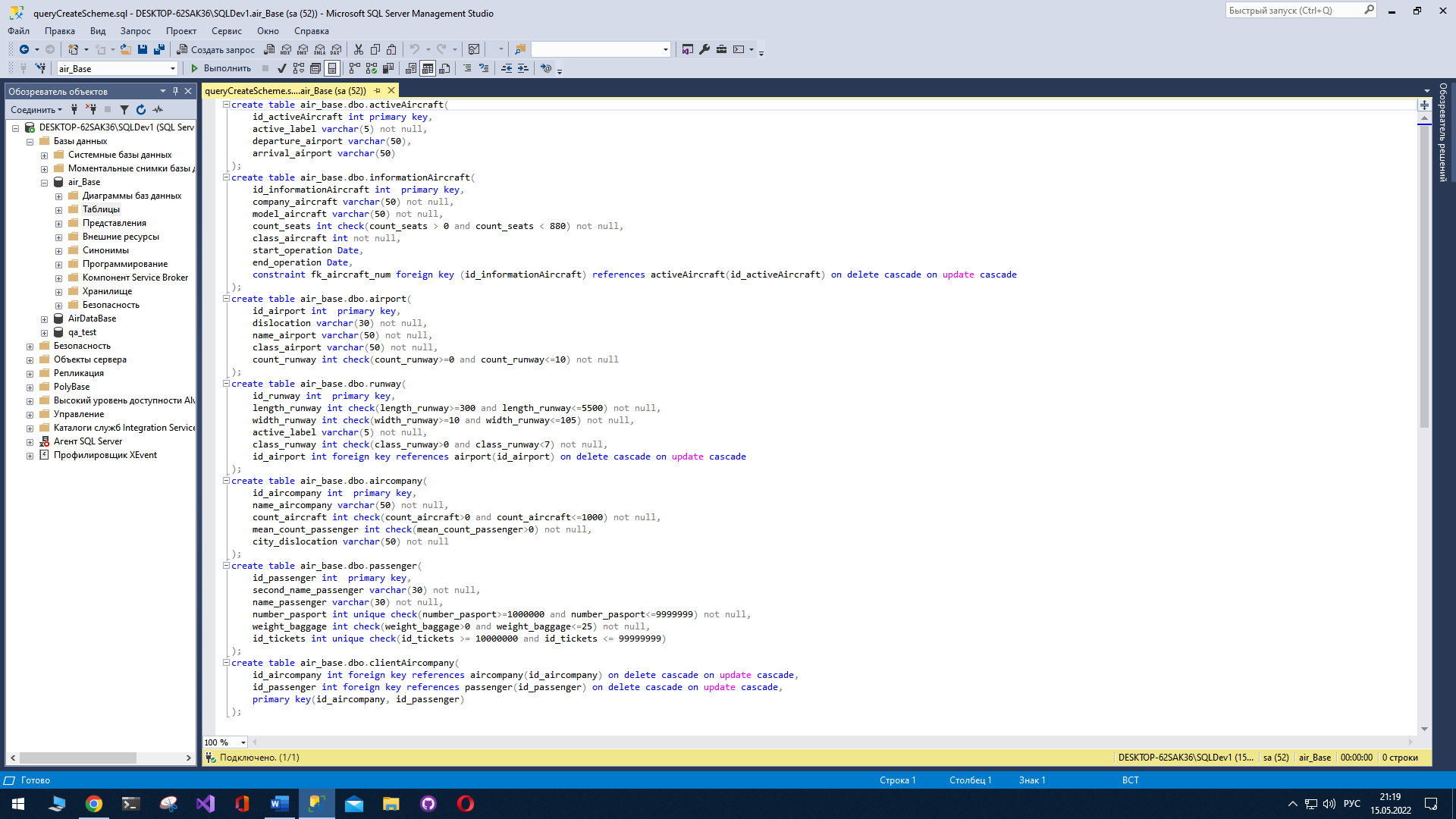
**Задание 3.1 Создание базы данных.**

Для создания базы данных в утилите SSMS выполним скрипт со следующей командой, как показано на рисунке 9.

  
Рисунок 9 – Создание базы данных

**Задание 3.2 Создание схемы ER-модели данных.**

Для создания схемы ER-модели данных в утилите SSMS выполним скрипт со следующими командами, как показано на рисунке 10.

  
Рисунок 10 – Создание схемы ER-модели данных

**Задание 4.1 Установка индексов.**

В качестве уникального индекса для каждой сущности был выбран его уникальный идентификационный номер – id.

**5. Разработать типовые запросы к СУБД на языке SQL. Получение списков данных. Агрегация. Поиск.**