**Промежуточная аттестация 2**

1. **Загрузить модель данных**

**Открываем командную строку cmd.**

**Прописываем путь в папку, где лежит СУБД (…\PostgreSQL\bin).**

**Закидываем в эту папку .sql файл с дампом модели данных.**

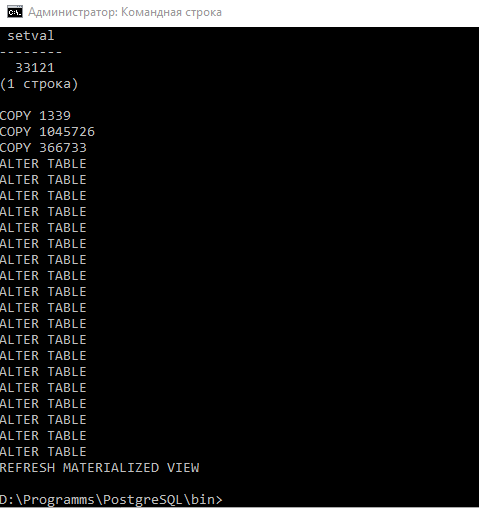
**Вводим в командную строку:**

**psql -U postgres**

**CREATE DATABASE demo;**

**\q**

**psql -U postgres -d demo < air.sql**

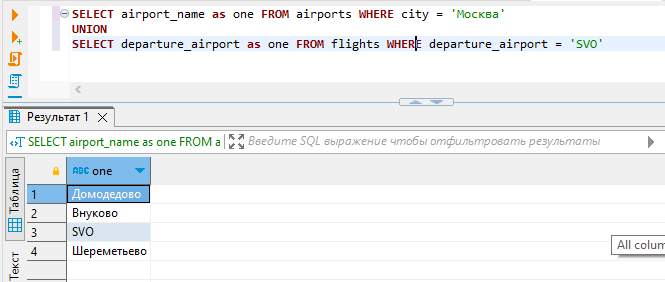
****

**1.1 Соединение двух произвольных таблиц при помощи UNION**

**SELECT** airport\_name **as** one **FROM** airports **WHERE** city = 'Москва'

**UNION**

**SELECT** departure\_airport **as** one **FROM** flights **WHERE** departure\_airport = 'SVO'



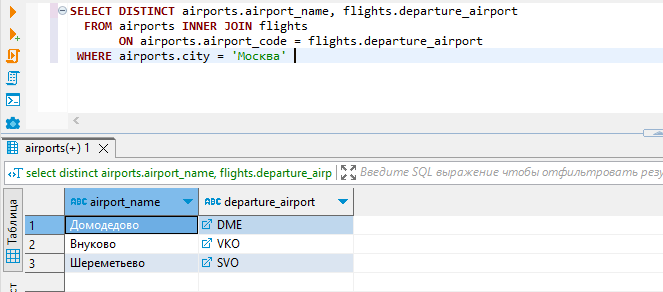
**1.2 Соединение двух произвольных таблиц при помощи JOIN**

**SELECT** **DISTINCT** airports.airport\_name, flights.departure\_airport

**FROM** airports **INNER** **JOIN** flights

**ON** airports.airport\_code = flights.departure\_airport

**WHERE** airports.city = 'Москва'

****

**2. Запрос с фильтром WHERE, сортировкой ORDER BY и ограничением по количеству строк LIMIT**

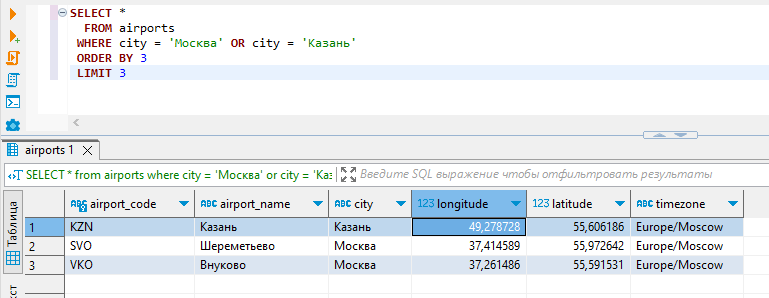
**SELECT** \*

**FROM** airports

**WHERE** city = 'Москва' **OR** city = 'Казань'

**ORDER** **BY** 3

**LIMIT** 3

****

**3. Выполнение SQL запроса в рамках JOIN оператора с использованием GROUP BY и любой из агрегатных функций (в данном случае – count(\*))**

**SELECT** airports.airport\_name **AS** "аэропорт отправления",

flights.departure\_airport **AS** "код аэропорта",

**count**(\*) **AS** "количество рейсов",

aircrafts.model **AS** "модель самолета"

**FROM** airports **INNER** **JOIN** flights

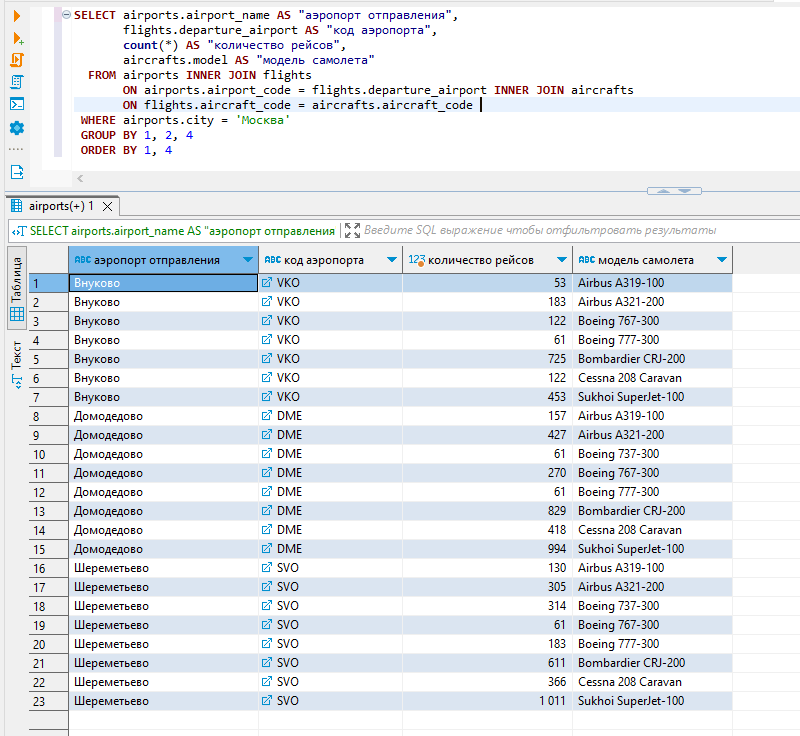
**ON** airports.airport\_code = flights.departure\_airport **INNER** **JOIN** aircrafts

**ON** flights.aircraft\_code = aircrafts.aircraft\_code

**WHERE** airports.city = 'Москва'

**GROUP** **BY** 1, 2, 4

**ORDER** **BY** 1, 4

****

**4. Выполнение SQL запроса с применением LEFT JOIN и RIGHT JOIN (применение INNER JOIN было показано в предыдущем запросе) для более чем двух (в данном случае трех) таблиц.**

**SELECT** airports.airport\_name **AS** "аэропорт отправления",

flights.departure\_airport **AS** "код аэропорта",

**count**(\*) **AS** "количество рейсов",

aircrafts.model **AS** "модель самолета"

**FROM** airports **RIGHT** **JOIN** flights

**ON** airports.airport\_code = flights.departure\_airport **LEFT** **JOIN** aircrafts

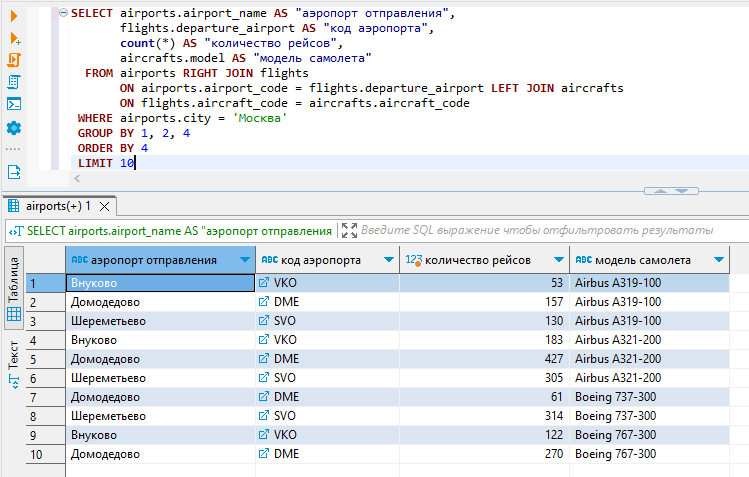
**ON** flights.aircraft\_code = aircrafts.aircraft\_code

**WHERE** airports.city = 'Москва'

**GROUP** **BY** 1, 2, 4

**ORDER** **BY** 4

**LIMIT** 10

****

**5. Создание виртуального представления VIEW с произвольным именем для SQL запроса из задания 2**

**CREATE** **VIEW** second\_task **AS**

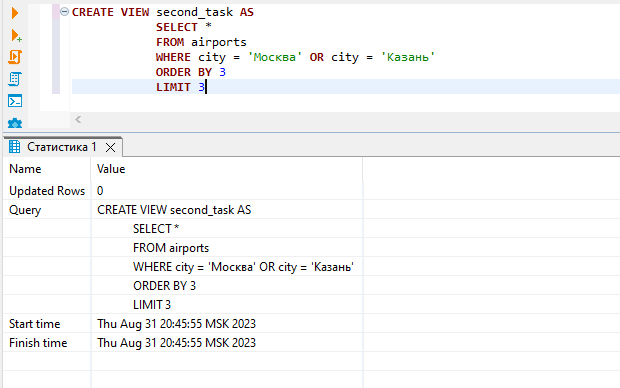
**SELECT** \*

**FROM** airports

**WHERE** city = 'Москва' **OR** city = 'Казань'

**ORDER** **BY** 3

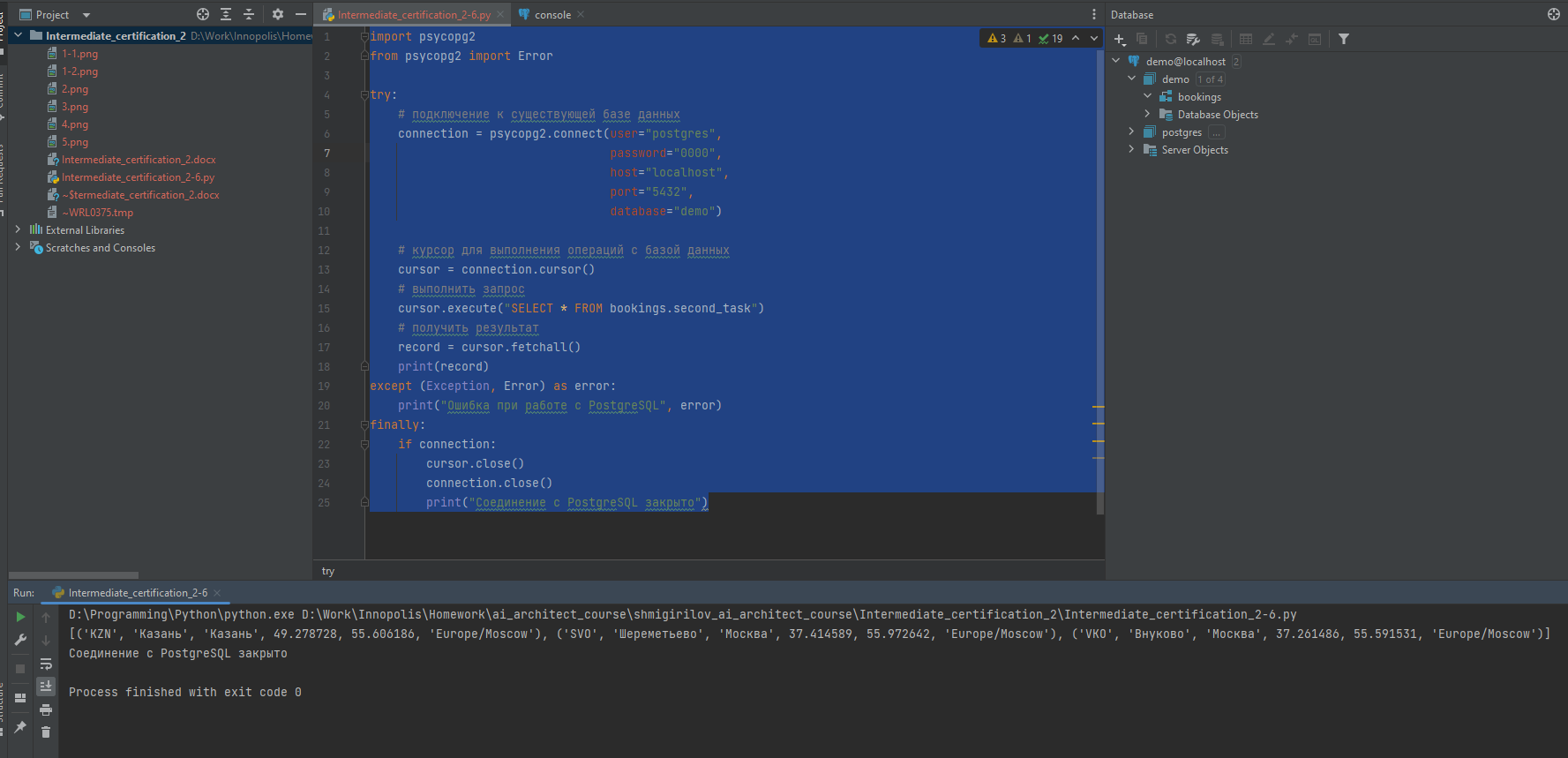
**LIMIT** 3

****

**6. Подключение к БД при помощи psycopg2 с выполнением запроса из PyCharm.**

**Вызвать виртуальное представление, созданное в задании 5.**

import psycopg2  
from psycopg2 import Error  
  
try:  
 # подключение к существующей базе данных  
 connection = psycopg2.connect(user="postgres",  
 password="0000",  
 host="localhost",  
 port="5432",  
 database="demo")  
  
 # курсор для выполнения операций с базой данных  
 cursor = connection.cursor()  
 # выполнить запрос  
 cursor.execute("SELECT \* FROM bookings.second\_task")  
 # получить результат  
 record = cursor.fetchall()  
 print(record)  
except (Exception, Error) as error:  
 print("Ошибка при работе с PostgreSQL", error)  
finally:  
 if connection:  
 cursor.close()  
 connection.close()  
 print("Соединение с PostgreSQL закрыто")

****