Промежуточная аттестация

Задание 1

Используя SQL язык и произвольные две таблицы из модели данных необходимо объединить их различными способами (UNION,JOIN)

**select**

t.passenger\_name **as** passenger

, t.contact\_data['phone'] **as** phone

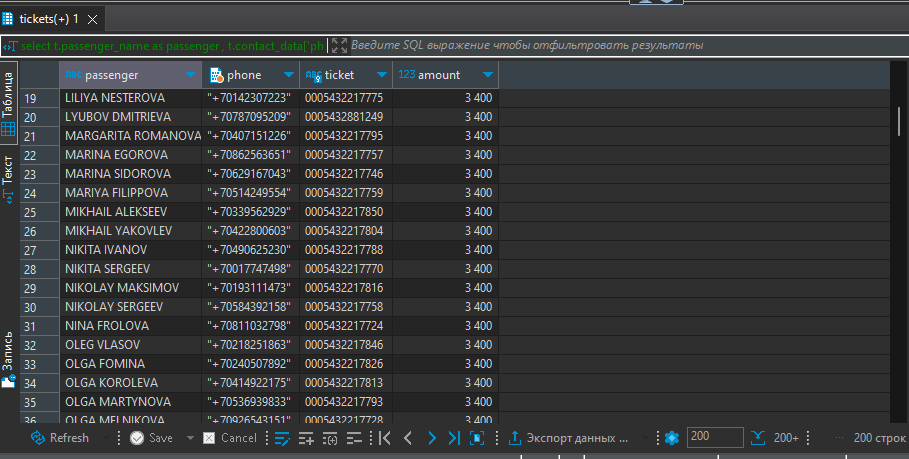
, t.ticket\_no **as** ticket

, b. total\_amount **as** amount

**from** tickets t

**left** **join** bookings b **on** b.book\_ref=t.book\_ref

**order** **by** b.total\_amount, t.passenger\_name



**select**

t.passenger\_name ,

tf.fare\_conditions ,

tf.amount

**from** ticket\_flights tf

**left** **join** tickets t **on** t.ticket\_no = tf.ticket\_no

**where** fare\_conditions = 'Business'

**union**

**select**

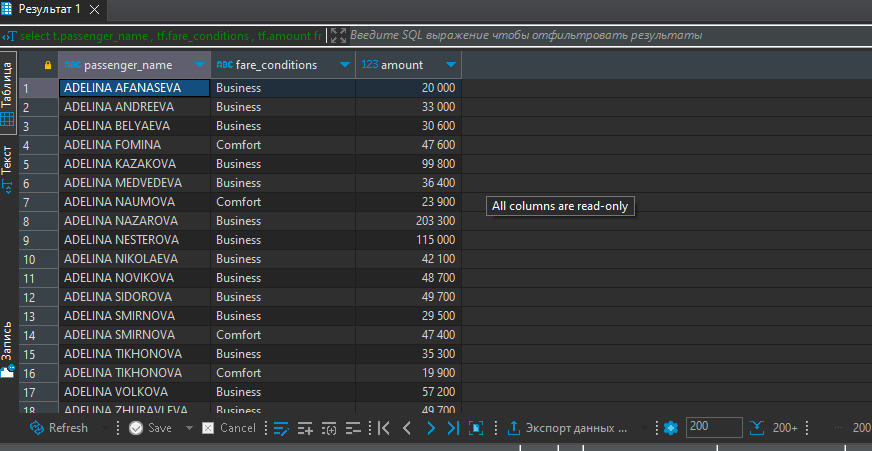
t.passenger\_name ,

tf.fare\_conditions ,

tf.amount

**from** ticket\_flights tf

**left** **join** tickets t **on** t.ticket\_no = tf.ticket\_no

**where** fare\_conditions = 'Comfort'

Задание 2

Используя SQL язык напишите запрос с любым фильтром WHERE к произвольной таблице и результат отсортируйте (ORDERBY) с ограничением вывода по количеству строк (LIMIT)

**select**

t.passenger\_name

, b.book\_date

, b.total\_amount

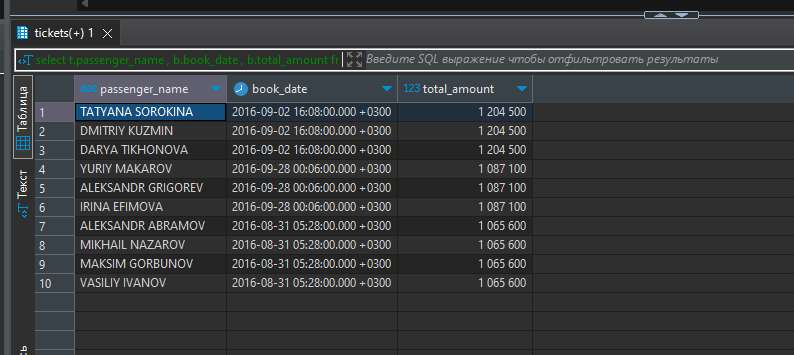
**from** bookings b

**inner** **join** tickets t **on** t.book\_ref = b.book\_ref

**where** book\_date **between** '2016-01-01' **and** '2016-12-31'

**order** **by** total\_amount **desc**

**limit** 10



Задание 3

Используя SQL язык напишите OLAP запрос к произвольной связке таблиц (в рамках JOIN оператора), используя оператор GROUPBY и любые агрегатные функции count, min, max, sum.

**select**

a.model ,

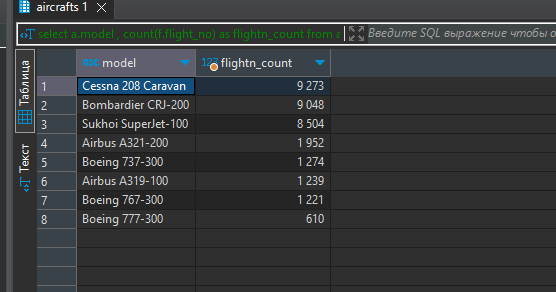
**count**(f.flight\_no) **as** flightn\_count

**from** aircrafts a **join**

flights f **on** f.aircraft\_code = a.aircraft\_code

**group** **by** a.model

**order** **by** **count**(f.flight\_no) **desc**



Задание 4

Используя SQL язык примените JOIN операторы (INNER, LEFT, RIGHT) для более чем двух таблиц из модели данных.

**SELECT**

f.flight\_no,

dep.airport\_name **AS** departure\_airport\_name,

dep.city **AS** departure\_city,

arr.airport\_name **AS** arrival\_airport\_name,

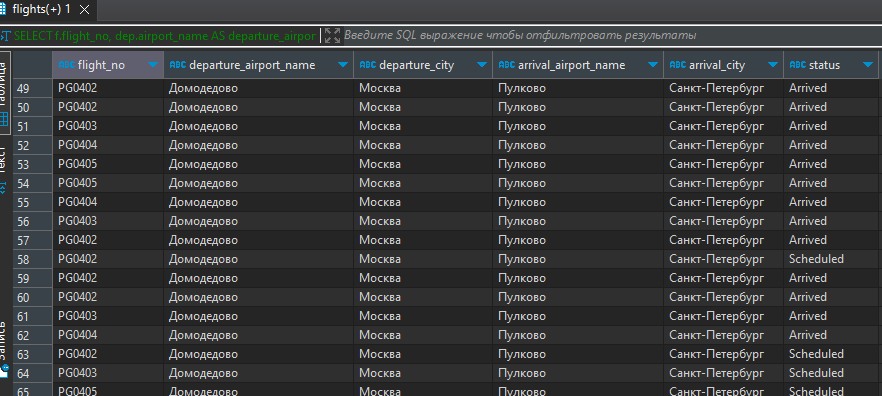
arr.city **AS** arrival\_city,

f.status

**FROM** flights f

**inner** **join** airports dep **on** f.departure\_airport = dep.airport\_code

**left** **join** airports arr **on** f.arrival\_airport = arr.airport\_code;



Задание 5

Создайте виртуальную таблицу VIEW с произвольным именем для SQL запроса из задания 2

**CREATE** **OR** **REPLACE** **VIEW** bookings\_v **as**

**select**

t.passenger\_name

, b.book\_date

, b.total\_amount

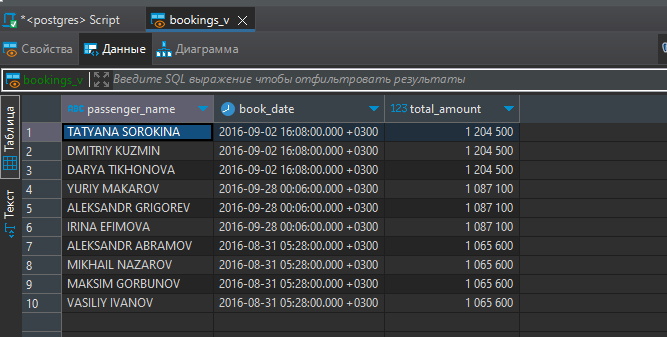
**from** bookings b

**inner** **join** tickets t **on** t.book\_ref = b.book\_ref

**where** book\_date **between** '2016-01-01' **and** '2016-12-31'

**order** **by** total\_amount **desc**

**limit** 10



Задание 6

Используя язык программирования Python, создайте Python-скрипт для вывода данных на экран (в консоль или IDE) из задания 5

import psycopg2  
connection = psycopg2.connect(database="demo", user="postgres", password="postgres", host="localhost", port=5432)  
cursor = connection.cursor()  
cursor.execute("SELECT \* from bookings.bookings\_v ;")  
# Получим данные  
columns = list(cursor.description)  
result = cursor.fetchall()  
  
# Создадим словарь  
results = []  
for row in result:  
 row\_dict = {}  
 for i, col in enumerate(columns):  
 row\_dict[col.name] = row[i]  
 results.append(row\_dict)  
  
# Выведем на экран  
for res in results:  
 print(res)

