



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01

## О Т Ч Е Т

по лабораторной работе № 1

Название: Введение в SQL

Дисциплина: Базы данных

Студент

ИУ6-31Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата)

Белоусов Н.И.  
(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

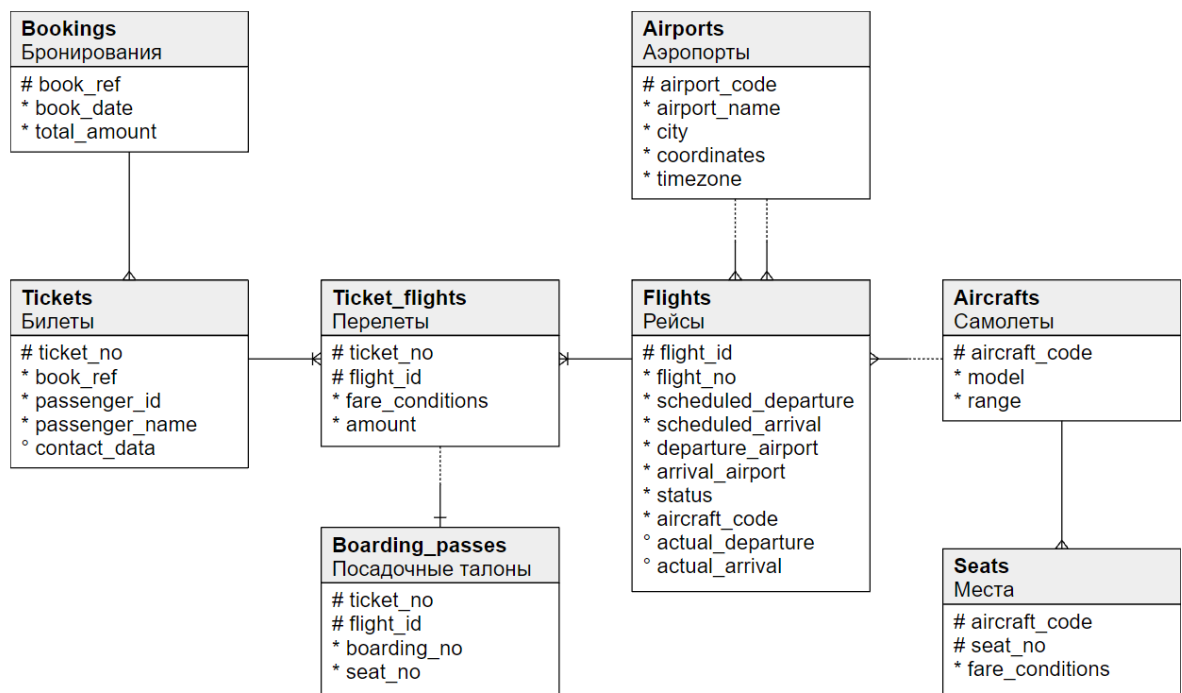
Миков Д.А.  
(И.О. Фамилия)

Москва, 2021

**Вариант 5**

**Задание:** Первое практическое задание заключается в знакомстве со средой pgAdmin и написании SQL-запросов с использованием оператора SELECT. Для модельной базы данных должны быть составлены 4 произвольных SELECT-запроса, демонстрирующие полученные знания. Запросы должны охватывать проработанные темы. После составления запросов следует убедиться в их правильности при помощи более простых запросов. Дополнительные вопросы могут заключаться в построении более сложных запросов или объяснении работы подготовленных заданий.

Для выполнения данной работы использовалась база данных об авиаперевозках со следующей структурой:



## Практическая часть

### Запрос 1

Все рейсы из Абакана на Сухом Суперджете 1 августа с 2:45 до 5:00.

Query

Query History

1

2

3

4

5

6

7

SELECT flights.scheduled\_departure AS "Departure", aircrafts.model AS "Aircraft",

airports.airport\_name AS "Airport", flights.flight\_no AS "Flight"

FROM bookings.flights, bookings.aircrafts, bookings.airports

WHERE model = 'Сухой Суперджет-100'

AND airport\_name = 'Абакан'

AND scheduled\_departure BETWEEN '2017-08-01 02:45:00' AND '2017-08-01 05:00:00'

ORDER BY scheduled\_departure

Data output

Messages

Notifications

	Departure timestamp with time zone	Aircraft text	Airport text	Flight character (6)
1	2017-08-01 02:45:00+03	Сухой Суперджет-100	Абакан	PG0328
2	2017-08-01 03:20:00+03	Сухой Суперджет-100	Абакан	PG0244
3	2017-08-01 03:40:00+03	Сухой Суперджет-100	Абакан	PG0037
4	2017-08-01 03:50:00+03	Сухой Суперджет-100	Абакан	PG0398
5	2017-08-01 03:55:00+03	Сухой Суперджет-100	Абакан	PG0660
6	2017-08-01 04:25:00+03	Сухой Суперджет-100	Абакан	PG0710
7	2017-08-01 04:35:00+03	Сухой Суперджет-100	Абакан	PG0199
8	2017-08-01 05:00:00+03	Сухой Суперджет-100	Абакан	PG0654

Total rows: 8 of 8

Query complete 00:00:21.297

Для проверки рассмотрим все рейсы из Абакана на Сухом Суперджете и найдем нужные рейсы. Можно увидеть, что все рейсы были корректно отображены при первом запросе: без потерь данных и без лишних записей.

Query
Query History

```

1 SELECT flights.scheduled_departure AS "Departure", aircrafts.model AS "Aircraft",
2 airports.airport_name AS "Airport", flights.flight_no AS "Flight"
3 FROM bookings.flights, bookings.aircrafts, bookings.airports
4 WHERE model = 'Сухой Суперджет-100'
5 AND airport_name = 'Абакан'
6 ORDER BY scheduled_departure

```

Data output
Messages
Notifications

+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	Departure timestamp with time zone	Aircraft text	Airport text	Flight character (6)
2	2017-07-16 02:45:00+03	Сухой Суперджет-1...	Абакан	PG0328
3	2017-07-16 03:20:00+03	Сухой Суперджет-1...	Абакан	PG0244
4	2017-07-16 03:40:00+03	Сухой Суперджет-1...	Абакан	PG0037
5	2017-07-16 03:50:00+03	Сухой Суперджет-1...	Абакан	PG0398
6	2017-07-16 03:55:00+03	Сухой Суперджет-1...	Абакан	PG0660
7	2017-07-16 04:10:00+03	Сухой Суперджет-1...	Абакан	PG0582
8	2017-07-16 04:35:00+03	Сухой Суперджет-1...	Абакан	PG0199
9	2017-07-16 05:00:00+03	Сухой Суперджет-1...	Абакан	PG0654

Total rows: 1000 of 33121
Query complete 00:00:00.404

## Запрос 2

Найдем пассажира, который забронировал билет 1 августа 2017 года и прибыл в пункт назначения раньше прочих.

Query    Query History

```
1  SELECT t.passenger_name AS passenger
2  FROM bookings.bookings b, bookings.tickets t,
3  bookings.ticket_flights tf, bookings.flights f
4  WHERE b.book_ref = t.book_ref
5  AND t.ticket_no = tf.ticket_no
6  AND tf.flight_id = f.flight_id
7  AND b.book_date = '2017-08-01'
8  ORDER BY f.scheduled_arrival
9  LIMIT 1
```

Data output    Messages    Notifications



	<div>passenger</div> <div>text</div> <div></div>
1	NATALIYA KONDRATEVA

Для проверки можем рассмотреть полный список всех пассажиров, забронировавших билет 1 августа 2017 года, а для удобства отсортируем записи по дате прибытия. Найдем 4 удовлетворяющих запрос записи, однако для поставленной задачи хватит одной - Наталья Кондратьева.

Query

Query History

1

```
SELECT b.book_date AS booking_time, f.scheduled_arrival AS arrived,
2 t.passenger_name AS passenger
3 FROM bookings.bookings b, bookings.tickets t,
4 bookings.ticket_flights tf, bookings.flights f
5 WHERE b.book_ref = t.book_ref
6 AND t.ticket_no = tf.ticket_no
7 AND tf.flight_id = f.flight_id
8 AND b.book_date = '2017-08-01'
9 ORDER BY arrived
10 LIMIT 4
```

Data output

Messages

Notifications

≡+

	<div>booking_time</div> <div>timestamp with time zone</div> <div>🔒</div>	<div>arrived</div> <div>timestamp with time zone</div> <div>🔒</div>	<div>passenger</div> <div>text</div> <div>🔒</div>
1	2017-08-01 00:00:00+03	2017-08-12 15:20:00+03	NATALIYA KONDRATE...
2	2017-08-01 00:00:00+03	2017-08-12 18:20:00+03	ANATOLIY AFANASEV
3	2017-08-01 00:00:00+03	2017-08-12 18:20:00+03	VLADIMIR MIRONOV
4	2017-08-01 00:00:00+03	2017-08-13 09:15:00+03	NATALIYA KONDRATE...

### Запрос 3

Найдем три самых вместительных самолета.

Query

Query History

1

SELECT a.model AS "Aircraft", count(\*) AS "Count\_of\_seats"

2

FROM bookings.aircrafts a, bookings.seats s

3

WHERE a.aircraft\_code = s.aircraft\_code

4

GROUP BY a.model

5

ORDER BY count(\*) DESC

6

LIMIT 3

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	Aircraft text	Count_of_seats bigint
1	Боинг 777-300	402
2	Боинг 767-300	222
3	Аэробус A321-200	170

Для проверки рассмотрим вместительность всех самолетов и обнаружим, что из списка были выбраны верные самолеты.

Query Query History

```
1 SELECT a.model AS "Aircraft", count(*) AS "Count_of_seats"
2 FROM bookings.aircrafts a, bookings.seats s
3 WHERE a.aircraft_code = s.aircraft_code
4 GROUP BY a.model
```

Data output    Messages    Notifications

	Aircraft text	Count_of_seats bigint
1	Сессна 208 Кар...	12
2	Боинг 737-300	130
3	Аэробус A321-2...	170
4	Аэробус A320-2...	140
5	Бомбардье CRJ-...	50
6	Боинг 767-300	222
7	Аэробус A319-1 ...	116
8	Сухой Супердж...	97
9	Боинг 777-300	402

Total rows: 9 of 9      Query complete 00:00:00.142

### *Запрос 4*

Найдем среднюю стоимость билета на самолет до Абакана, для упрощения проверки возьмем вылеты 1 августа до 12 утра



Query

Query History

1

SELECT a.city AS "City", round(avg(tf.amount)) AS "price\_of\_flight"

2

FROM bookings.airports a, bookings.flights f, bookings.ticket\_flights tf

3

WHERE a.airport\_code = f.departure\_airport

4

AND f.flight\_id = tf.flight\_id

5

AND a.city = 'Абакан'

6

AND f.scheduled\_departure BETWEEN '2017-08-01 00:00:00' AND '2017-08-01 12:00:00'

7

GROUP BY a.city

Data output

Messages

Notifications

	City text	price_of_flight numeric
1	Абакан	5800

Для проверки выведем стоимость каждого билета этого же временного промежутка.

Query

Query History

```

1 SELECT a.city AS "City", tf.amount AS "price_of_flight"
2 FROM bookings.airports a, bookings.flights f, bookings.ticket_flights tf
3 WHERE a.airport_code = f.departure_airport
4 AND f.flight_id = tf.flight_id
5 AND a.city = 'Абакан'
6 AND f.scheduled_departure BETWEEN '2017-08-01 00:00:00' AND '2017-08-01 12:00:00'
7

```

a.airport\_code

a.airport\_name  
a.city  
a.coordinates  
a.timezone  
f.actual\_arrival  
f.actual\_departure  
f.aircraft\_code  
f.arrival\_airport  
f.departure\_airport  
f.flight\_id

Data

+

	City	price_of_flight
1	Абакан	5800.00
2	Абакан	5800.00
3	Абакан	5800.00
4	Абакан	5800.00
5	Абакан	5800.00
6	Абакан	5800.00
7	Абакан	5800.00
8	Абакан	5800.00
9	Абакан	5800.00
10	Абакан	5800.00

## Ответы на контрольные вопросы

**Рассказать про операцию соединения (JOIN) и различные её разновидности.**

JOIN – способ связать таблицы: INNER JOIN – пересечение двух множеств (условий)

RIGHT/LEFT JOIN – Пересечение двух таблиц и остальные значения из одной из таблиц

FULL JOIN – Объединение двух множеств (условий)

**Рассказать про агрегатные функции, предложения GROUP BY и HAVING.**

Агрегатные функции позволяют вычислить сумму, среднее, максимум и т.д. значений указанного поля для всех записей с соответствующими значениями полей, указанных в GROUP BY. HAVING работает аналогично WHERE, только уже для агрегированной таблицы

**Как выбрать только уникальные значения какого-либо столбца?**

Использовать SELECT DISTINCT

**Как осуществить сортировку по возрастанию/убыванию по значению какого-либо столбца?**

Использовать ORDER BY <поле> ASC/DESC (по умолчанию выбирается сортировка по возрастанию)

**Как агрегатные функции ведут себя по отношению к неопределённым значениям?**

Агрегатные функции игнорируют NULL. Исключение – count(\*) – она считает кол-во записей, поэтому даже если все поля записи будут NULL, count(\*) его посчитает

**Рассказать о теоретико-множественных операциях в SQL.**

При перечислении таблиц после FROM производится декартово произведение, результатом которого является таблица, составленная из полей исходных (каждая строка с каждой)

Если две таблицы имеют одинаковое кол-во полей и их порядок, то их можно объединить – UNION, а можно пересечь – INTERSECT, ещё можно исключить записи одной таблицы из другой – EXCEPT

**Чем отличаются UNION и UNION ALL?**

В результате объединения UNION нет дубликатов, а при UNION\_ALL они могут быть

**Чем отличаются COUNT(\*) и COUNT(field)?**

Count(field) не будет считать записи, где field = NULL, а count(\*) их посчитает

**Как подсчитать количество уникальных значений столбца?**

SELECT field, count(field) FROM table GROUP BY field

**Как можно осуществить проверку на неопределенное значение?**

WHERE field IS NULL

**Рассказать про предикат LIKE.**

LIKE сравнивает строку с указанным шаблоном WHERE field LIKE '%ing'

**Как можно выбрать только определенное количество строк?**

LIMIT – ограничение кол-ва строк в результирующей таблице

**Чем SQL-таблица отличается от отношения?**

Отношение можно назвать основным объектом реляционной алгебры

Это связанные между собой наборы данных, а sql таблица – это способ СУБД (программиста) представить это отношение

**Округлить результирующее значение до 3 знаков после точки.**

ROUND(field, 3)

**Округлить вещественное число до целого без нулей после точки.**

ROUND(field)

**Переписать запрос, не используя функцию MAX (MIN).**

Отсортировать по убыванию/возрастанию и взять первый элемент

**Изменить формат вывода данных (например, формат даты и времени).**

DATE\_PART('hour', field)

**Вывод:** в ходе выполнения лабораторной работы которой познакомился со средой pgAdmin, СУБД PostgreSQL, на практике опробовал знания по написанию SQL-запросов.