

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
"Уфимский государственный авиационный технический университет"**

**Кафедра** Высокопроизводительных вычислительных технологий и систем

**Дисциплина:** Базы данных

**Отчет по лабораторной работе № 4**

**Тема:** ««Основы SQL-запросов. JOIN, агрегирование и группировка,  
подзапросы»»

Группа ПМ-353	Фамилия И.О.	Подпись	Дата	Оценка
Студент	Шамаев И.Р.			
Принял	Ямилева А.М.			

**Уфа 2022**

## Теоритическая часть

Для подключения базы данных используется команда

```
psql -d demo -U postgres
```

Для создания таблиц в языке SQL служит команда CREATE TABLE.

Для удаления таблицы служит команда DROP TABLE;

Для выполнения вводу данных в таблицу служит команда INSERT. Ее упрощенный формат таков:

```
INSERT INTO имя-таблицы [( имя-атрибута, имя-атрибута, ... )] VALUES  
( значение-атрибута, значение-атрибута, ... );
```

Для выборки информации из таблиц базы данных служит команда SELECT. Ее синтаксис, упрощенный до предела, таков:

```
SELECT имя-атрибута, имя-атрибута, ... FROM имя-таблицы;
```

Теперь мы ознакомимся с командой UPDATE, предназначенной для обновления данных в таблицах. Ее упрощенный синтаксис таков:

```
UPDATE имя-таблицы SET имя-атрибута1 = значение-атрибута1, имя-  
атрибута2 = значение-атрибута2, ... WHERE условие;
```

## Типы данных СУБД PostgreSQL

Группа числовых типов данных включает в себя целый ряд разновидностей: целочисленные типы, числа фиксированной точности, типы данных с плавающей точкой, последовательные типы (serial). В составе целочисленных типов находятся следующие представители: smallint, integer, bigint. Если атрибут таблицы имеет один из этих типов, то он позволяет хранить только целочисленные данные. При этом перечисленные типы различаются по количеству байтов, выделяемых для хранения данных.

Числа фиксированной точности представлены двумя типами — numeric и decimal. Однако они являются идентичными по своим возможностям. Для задания значения этого типа используются два базовых понятия: масштаб (scale) и точность (precision). Масштаб показывает число значащих цифр, стоящих справа от десятичной точки (запятой). Точность указывает общее число цифр как до десятичной точки, так и после нее.

Стандартные представители строковых типов — это типы character varying(n) и character(n), где параметр указывает максимальное число символов в строке, которую можно сохранить в столбце такого типа. При работе с многобайтовыми кодировками символов, например UTF-8, нужно учитывать, что речь идет о символах, а не о байтах. Если сохраняемая строка

символов будет короче, чем указано в определении типа, то значение типа `character` будет дополнено пробелами до требуемой длины, а значение типа `character varying` будет сохранено так, как есть.

PostgreSQL поддерживает все типы данных, предусмотренные стандартом SQL для даты и времени. Даты обрабатываются в соответствии с григорианским календарем, причем это делается даже в тех случаях, когда дата относится к тому моменту времени, когда этот календарь в данной стране еще не был принят. Для этих типов данных предусмотрены определенные форматы для ввода значений и для вывода. Причем эти форматы могут не совпадать.

В результате объединения типов даты и времени получается интегральный тип — временная отметка. Этот тип существует в двух вариантах: с учетом часового пояса — `timestamp with time zone`, либо без учета часового пояса — `timestamp`. Для первого варианта существует сокращенное наименование — `timestampz`, которое является расширением PostgreSQL. При вводе и выводе значений этого типа данных используются соответствующие форматы ввода и вывода даты и времени.

Для получения значения текущей временной отметки (т. е. даты и времени в одном ' значении) служит функция `current_timestamp`.

Последним типом является `interval`, который представляет продолжительность отрезка времени между двумя моментами времени. Его формат ввода таков: `quantity unit [quantity unit ...] direction`

Логический (`boolean`) тип может принимать три состояния: истина и ложь, а так же неопределенное состояние, которое можно представить значением `NULL`. Таким образом, тип `boolean` реализует трехзначную логику.

## Практическая часть

### Задание 4.2

```
demo=# SELECT model, range
demo=# FROM aircrafts;
      model      | range
-----+-----
Боинг 777-300    | 11100
Боинг 767-300    | 7900
Сухой Суперджет-100 | 3000
Аэробус A320-200  | 5700
Аэробус A321-200  | 5600
Аэробус A319-100  | 6700
Боинг 737-300    | 4200
Сессна 208 Караван | 1200
Бомбардье CRJ-200 | 2700
(9 строк)
```

### Задание 4.3

```
demo=# SELECT * FROM aircrafts WHERE range BETWEEN 4000 AND 10000;
aircraft_code |      model      | range
-----+-----+-----
763           | Боинг 767-300   | 7900
320           | Аэробус A320-200 | 5700
321           | Аэробус A321-200 | 5600
319           | Аэробус A319-100 | 6700
733           | Боинг 737-300   | 4200
(5 строк)
```

```
demo=# SELECT * FROM aircrafts WHERE CAST(range AS varchar) !~ '100$' AND range>6000;
aircraft_code |      model      | range
-----+-----+-----
763           | Боинг 767-300   | 7900
319           | Аэробус A319-100 | 6700
(2 строки)
```

## Задание 4.4

```
demo=# SELECT DISTINCT flight_no, actual_departure FROM flights WHERE actual_arrival > scheduled_arrival AND status='Arrived';
flight_no |      actual_departure
-----+-----
PG0473    | 2017-07-22 07:03:00+00
PG0412    | 2017-07-24 08:02:00+00
PG0649    | 2017-08-11 12:40:00+00
PG0227    | 2017-08-13 06:46:00+00
PG0184    | 2017-07-19 15:52:00+00
PG0542    | 2017-08-11 10:40:00+00
PG0286    | 2017-07-30 12:50:00+00
PG0174    | 2017-08-03 16:06:00+00
PG0648    | 2017-08-13 08:38:00+00
PG0233    | 2017-08-05 09:17:00+00
PG0510    | 2017-08-06 15:03:00+00
PG0196    | 2017-07-27 09:14:00+00
PG0324    | 2017-07-30 03:52:00+00
PG0574    | 2017-08-09 08:49:00+00
PG0587    | 2017-08-05 15:50:00+00
PG0159    | 2017-07-26 07:45:00+00
PG0469    | 2017-07-26 09:38:00+00
PG0513    | 2017-07-26 15:41:00+00
PG0384    | 2017-07-26 09:15:00+00
PG0483    | 2017-08-08 04:23:00+00
PG0536    | 2017-08-02 05:29:00+00
PG0238    | 2017-08-09 08:07:00+00
PG0056    | 2017-07-31 13:33:00+00
PG0196    | 2017-08-02 09:12:00+00
PG0012    | 2017-07-18 07:59:00+00
PG0698    | 2017-07-16 13:38:00+00
PG0055    | 2017-08-15 11:12:00+00
PG0026    | 2017-07-30 05:08:00+00
PG0046    | 2017-08-02 15:53:00+00
PG0556    | 2017-08-12 09:47:00+00
PG0390    | 2017-08-11 04:14:00+00
PG0548    | 2017-07-19 05:42:00+00
```

## Задание 4.5

```
demo=# SELECT count (*) FROM flights WHERE EXTRACT(isodow from actual_arrival) = 4 AND EXTRACT(isodow from scheduled_arrival)= 4 AND status='Cancelled' AND departure_airport='LED';
count
-----
      0
(1 строка)
```

## Задание 4.6

```
demo=# SELECT b.passenger_name, a.amount, a.fare_conditions
demo=# FROM ticket_flights AS a
demo=# JOIN tickets AS b
demo=# ON a.ticket_no = b.ticket_no
demo=# WHERE amount > 70000 AND fare_conditions='Economy';
```

passenger_name	amount	fare_conditions
LYUDMILA AKIMOVA	73100.00	Economy
PETR VASILEV	73100.00	Economy
ELIZAVETA KARPOVA	73100.00	Economy
NIKOLAY ALEKSEEV	70800.00	Economy
SERGEY SEMENOV	70800.00	Economy
NATALYA FILATOVA	73100.00	Economy
OLGA VASILEVA	73100.00	Economy
OLGA IVANOVA	73100.00	Economy
MARINA KUZNECOVA	73100.00	Economy
PETR EGOROV	73100.00	Economy
MARINA KARPOVA	73100.00	Economy
OLGA EGOROVA	70800.00	Economy
OKSANA DMITRIEVA	70800.00	Economy
EKATERINA ISAEVA	70800.00	Economy
ELENA KOVALEVA	70800.00	Economy
ARTEMIY KOZLOV	70800.00	Economy
VIKTORIYA MOISEEVA	73100.00	Economy
ELENA FADEEVA	73100.00	Economy
ALEKSANDRA TIKHONOVA	74500.00	Economy
VARVARA NAUMOVA	74500.00	Economy
EKATERINA SERGEEVA	74500.00	Economy
NATALYA ZAKHAROVA	74500.00	Economy
GALINA EFREMOVA	74500.00	Economy
ANDREY ZAKHAROV	74500.00	Economy
ALEKSANDR CHERNOV	74500.00	Economy
ARTUR KOROLEV	74500.00	Economy
PETR DENISOV	74500.00	Economy
SERGEY ANTONOV	74500.00	Economy
LYUBOV STEPANOVA	74500.00	Economy
MARINA FILIPPOVA	74500.00	Economy
PAVEL ZAKHAROV	74500.00	Economy
ELENA MELNIKOVA	74500.00	Economy
PETR KOROLEV	74500.00	Economy
ALEKSANDRA EGOROVA	74500.00	Economy
GALINA MEDVEDEVA	74500.00	Economy
MIKHAIL MATVEEV	73200.00	Economy
LIDIYA CHERNOVA	73200.00	Economy
VASILISA ZAYCEVA	73200.00	Economy
VASILIIY MARTYNOV	73200.00	Economy
KLAVDIYA BOGDANOVA	73200.00	Economy
MIKHAIL BELOV	73200.00	Economy
SERGEY PETROV	73200.00	Economy

## Задание 4.9

```
demo=# SELECT DISTINCT c.seat_no, a.flight_id, a.scheduled_departure
demo=# FROM flights AS a
demo=# JOIN boarding_passes AS b
demo=# ON a.flight_id=b.flight_id
demo=# JOIN seats AS c
demo=# ON b.seat_no!=c.seat_no
demo=# WHERE departure_airport='AAQ' AND arrival_airport='SVO';
```

seat_no	flight_id	scheduled_departure
36E	20991	2017-08-13 10:05:00+00
32D	20996	2017-08-06 10:05:00+00
29J	20991	2017-08-13 10:05:00+00
27F	21002	2017-07-20 10:05:00+00
6F	21037	2017-08-10 10:05:00+00
9C	20986	2017-07-25 10:05:00+00
46B	21006	2017-07-24 10:05:00+00
10C	20993	2017-07-16 10:05:00+00
13K	20997	2017-08-11 10:05:00+00
17F	21016	2017-08-05 10:05:00+00
33E	21041	2017-07-18 10:05:00+00
39J	21041	2017-07-18 10:05:00+00
5G	20997	2017-08-11 10:05:00+00
22H	20982	2017-07-17 10:05:00+00
27A	21039	2017-07-23 10:05:00+00
46J	21012	2017-08-02 10:05:00+00
46C	20993	2017-07-16 10:05:00+00
4H	21012	2017-08-02 10:05:00+00
47H	21037	2017-08-10 10:05:00+00
42J	20986	2017-07-25 10:05:00+00
15B	21033	2017-08-12 10:05:00+00
11K	21016	2017-08-05 10:05:00+00
36D	21006	2017-07-24 10:05:00+00
12F	21037	2017-08-10 10:05:00+00
30K	21007	2017-08-07 10:05:00+00
41D	20981	2017-07-21 10:05:00+00
50C	20999	2017-07-22 10:05:00+00
10F	21012	2017-08-02 10:05:00+00
25J	20995	2017-07-27 10:05:00+00
8F	20996	2017-08-06 10:05:00+00
41G	20999	2017-07-22 10:05:00+00
20G	21040	2017-07-28 10:05:00+00
45E	20997	2017-08-11 10:05:00+00
38H	20999	2017-07-22 10:05:00+00
11E	21028	2017-07-30 10:05:00+00
40E	21031	2017-08-08 10:05:00+00
30F	21031	2017-08-08 10:05:00+00
10E	21038	2017-08-15 10:05:00+00
42D	21040	2017-07-28 10:05:00+00

## Задание 4.10

```
demo=# SELECT avg(a.amount)
demo=# FROM ticket_flights as a
demo=# JOIN flights as b
demo=# ON a.flight_id=b.flight_id
demo=# WHERE departure_airport='VOZ' AND arrival_airport='LED';
      avg
-----
10601.876675603217
(1 строка)

demo=# SELECT max(a.amount)
demo=# FROM ticket_flights as a
demo=# JOIN flights as b
demo=# ON a.flight_id=b.flight_id
demo=# WHERE departure_airport='VOZ' AND arrival_airport='LED';
      max
-----
11500.00
(1 строка)

demo=# SELECT sum(a.amount)
demo=# FROM ticket_flights as a
demo=# JOIN flights as b
demo=# ON a.flight_id=b.flight_id
demo=# WHERE departure_airport='VOZ' AND arrival_airport='LED';
      sum
-----
3954500.00
(1 строка)

demo=# SELECT min(a.amount)
demo=# FROM ticket_flights as a
demo=# JOIN flights as b
demo=# ON a.flight_id=b.flight_id
demo=# WHERE departure_airport='VOZ' AND arrival_airport='LED';
      min
-----
10500.00
(1 строка)
```

## Задание 4.11

```
demo=# SELECT fare_conditions, avg(amount) FROM ticket_flights GROUP BY fare_conditions;
fare_conditions |      avg
-----+-----
Business        | 51143.416138681927
Comfort         | 32740.552888786074
Economy         | 15959.813334810321
(3 строки)
```



## Задание 4.12

```
demo=# SELECT a.model, count(b.seat_no)
demo=# FROM aircrafts AS a
demo=# JOIN seats AS b
demo=# ON a.aircraft_code=b.aircraft_code
demo=# GROUP BY a.model;
```

model	count
Сессна 208 Караван	12
Боинг 737-300	130
Аэробус A321-200	170
Аэробус A320-200	140
Бомбардье CRJ-200	50
Боинг 767-300	222
Аэробус A319-100	116
Сухой Суперджет-100	97
Боинг 777-300	402

(9 строк)

## Задание 4.13

```
demo=# SELECT a.airport_code, count(b.flight_id)
demo=# FROM airports AS a
demo=# JOIN flights AS b
demo=# ON b.arrival_airport=a.airport_code
demo=# GROUP BY a.airport_code
demo=# HAVING count(b.flight_id) >= 500;
```

airport_code	count
AER	584
LED	1902
KJA	707
BZK	610
SVX	688
DME	3217
PEE	619
ROV	620
VKO	1717
NUX	506
OVB	1055
OVS	549
SVO	2982
HMA	515
SGC	530

(15 строк)

## Задание 4.20

ОШИБКА: Подзапрос в выражении вернул больше одной строки

```
demo=# SELECT model FROM aircrafts
demo=# WHERE range > 5000 AND interval '6' hour < (SELECT max(actual_duration) FROM flights_v);
      model
-----
Боинг 777-300
Боинг 767-300
Аэробус А320-200
Аэробус А321-200
Аэробус А319-100
(5 строк)
```

## Задание 4.21

```
demo=# SELECT count(scheduled_departure - actual_departure) FROM flights WHERE scheduled_departure - actual_departure + interval '4' hour < interval '1' second;
      count
-----
         25
(1 строка)
```

ОШИБКА: Оператор с данными именем и типами аргументов не найден. Возможно, вам следует добавить явные приведения типов.

```
demo=# SELECT scheduled_departure - actual_departure AS res FROM flights WHERE scheduled_departure - actual_departure + interval '4' hour < interval '1' second ORDER BY res ASC;
      res
-----
-04:37:00
-04:28:00
-04:27:00
-04:20:00
-04:18:00
-04:18:00
-04:16:00
-04:16:00
-04:14:00
-04:08:00
-04:08:00
-04:08:00
-04:08:00
-04:08:00
-04:06:00
-04:04:00
-04:03:00
-04:03:00
-04:02:00
-04:02:00
-04:02:00
-04:02:00
-04:02:00
-04:01:00
-04:01:00
-04:00:00
(25 строк)
```

## Задание 4.22

```
demo=# SELECT DISTINCT a.arrival_airport, (SELECT count(flight_id) FROM flights WHERE a.arrival_airport=arrival_airport) * (SELECT count(flight_id) FROM flights WHERE a.arrival_airport=departure_airport) as res
FROM flights AS a ORDER BY res DESC LIMIT 10;
 arrival_airport | res
-----
DME               | 6434
SVO               | 5963
LED               | 3802
VKO               | 3436
OVB               | 2110
KJA               | 1414
SVX               | 1377
PEE               | 1238
ROV               | 1237
BZK               | 1220
(10 строк)
```

### **Список литературы**

1. PostgreSQL. Основы языка SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова. — СПб.: БХВ-Петербург, 2018. — 336 с.