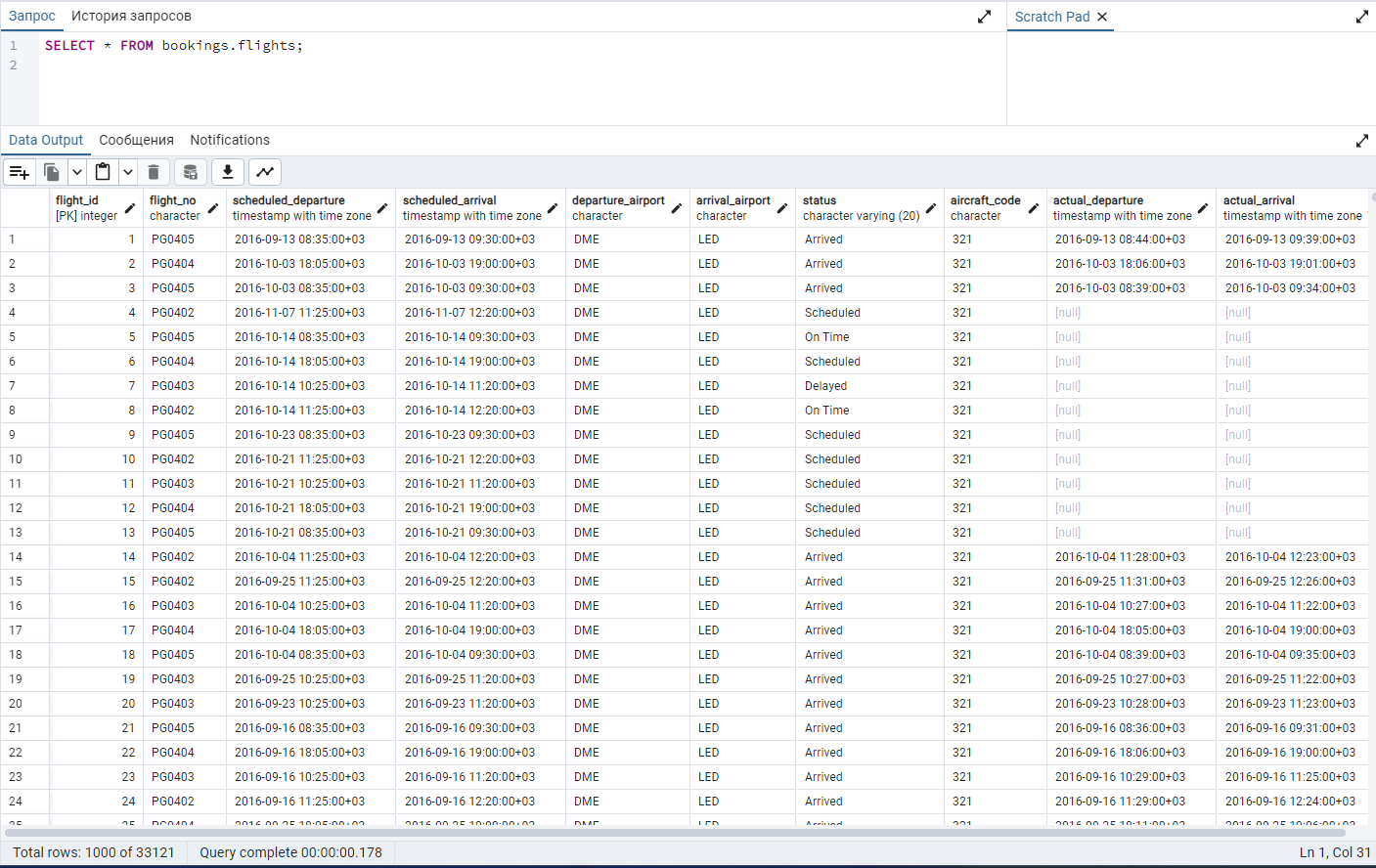
Выполнила: Саулова А.М.

Задание №1

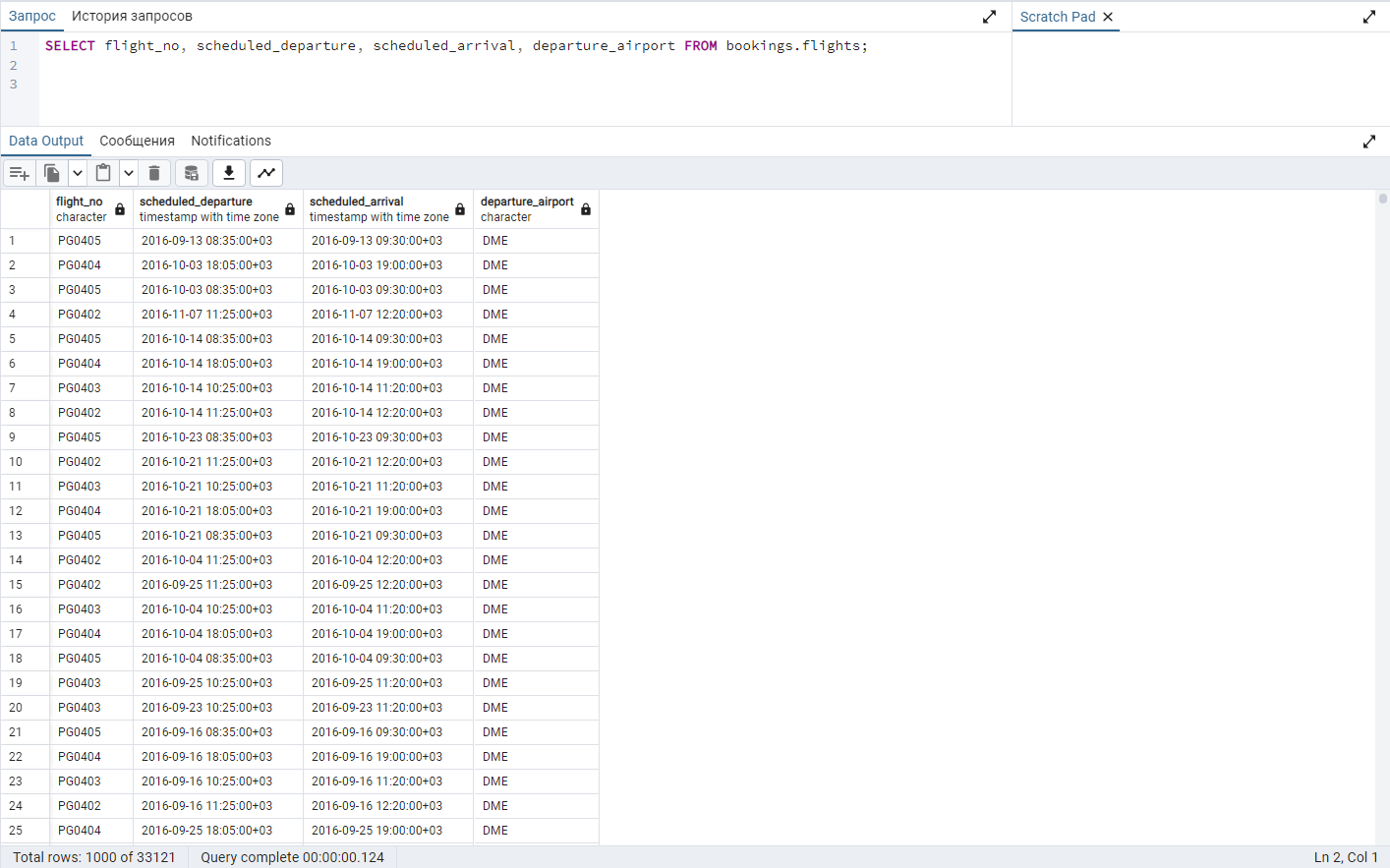
1. SELECT col1, col2, ...colN FROM tableName;

SELECT \* FROM bookings.flights;



* выборка всех записей, содержащихся в таблице.

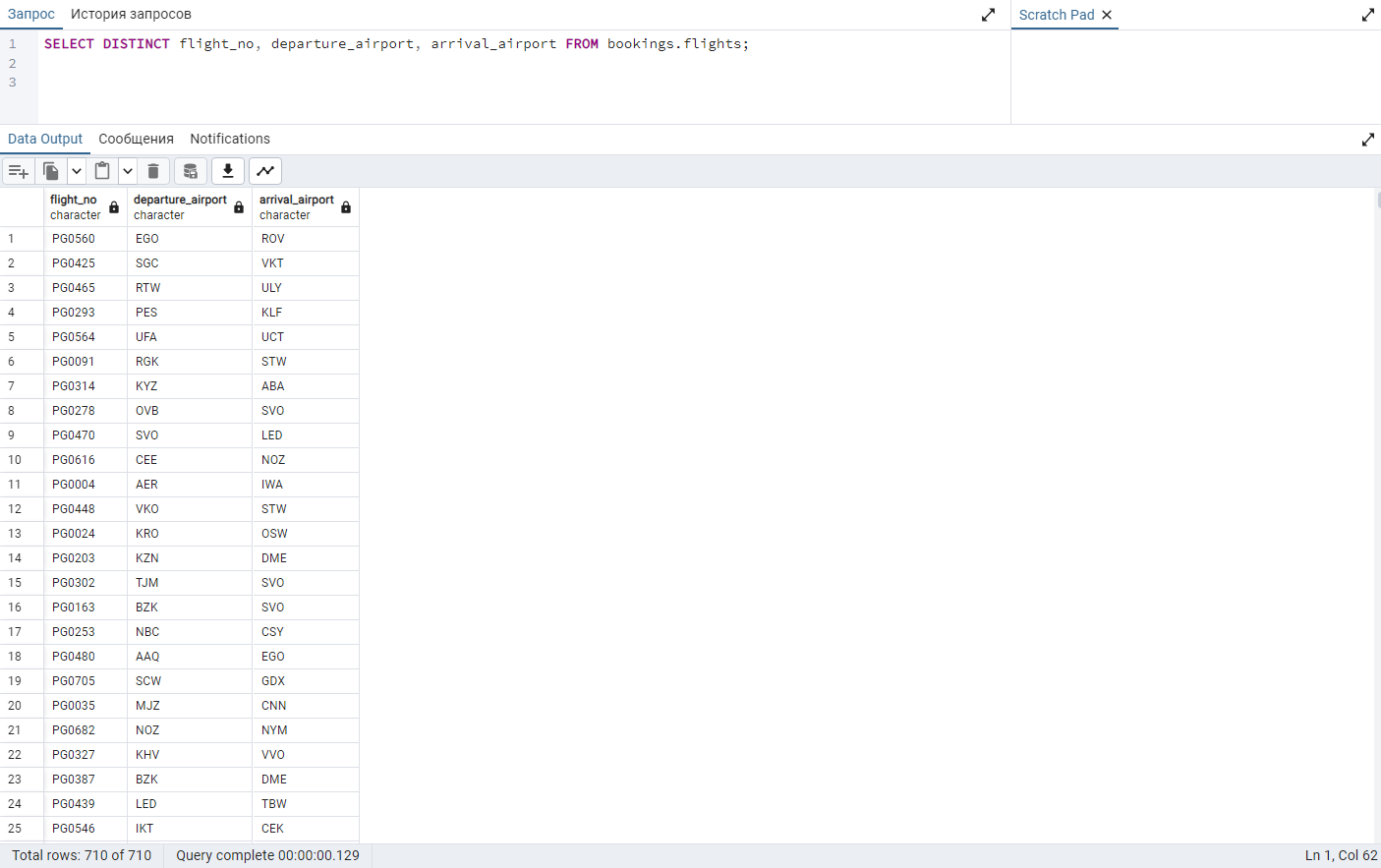
SELECT flight\_no, scheduled\_departure, scheduled\_arrival, departure\_airport FROM bookings.flights;



* выборка всех записей, содержащихся в таблице, по определенным столбцам

1. SELECT DISTINCT col1, col2, ...colN FROM tableName;

SELECT DISTINCT flight\_no, departure\_airport, arrival\_airport FROM bookings.flights;



- выборка различных значений по полям.

1. SELECT col1, col2, ...colN

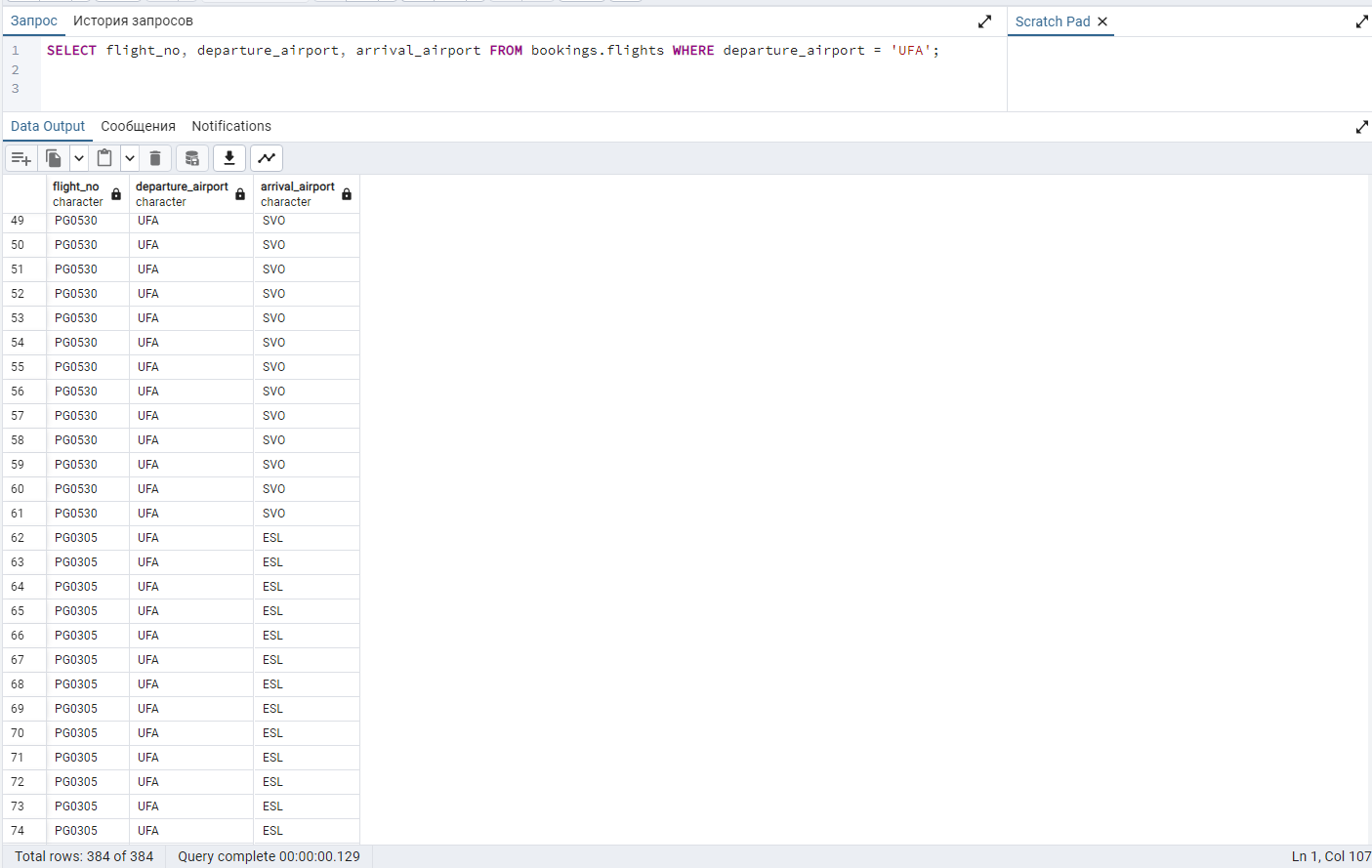
FROM tableName

WHERE condition;

SELECT flight\_no, departure\_airport, arrival\_airport

FROM bookings.flights

WHERE departure\_airport = 'UFA';



* Выборка записей, где departure\_airport равен UFA.

1. использовать как AND так и OR в одном условии.

SELECT col1, col2, ...colN

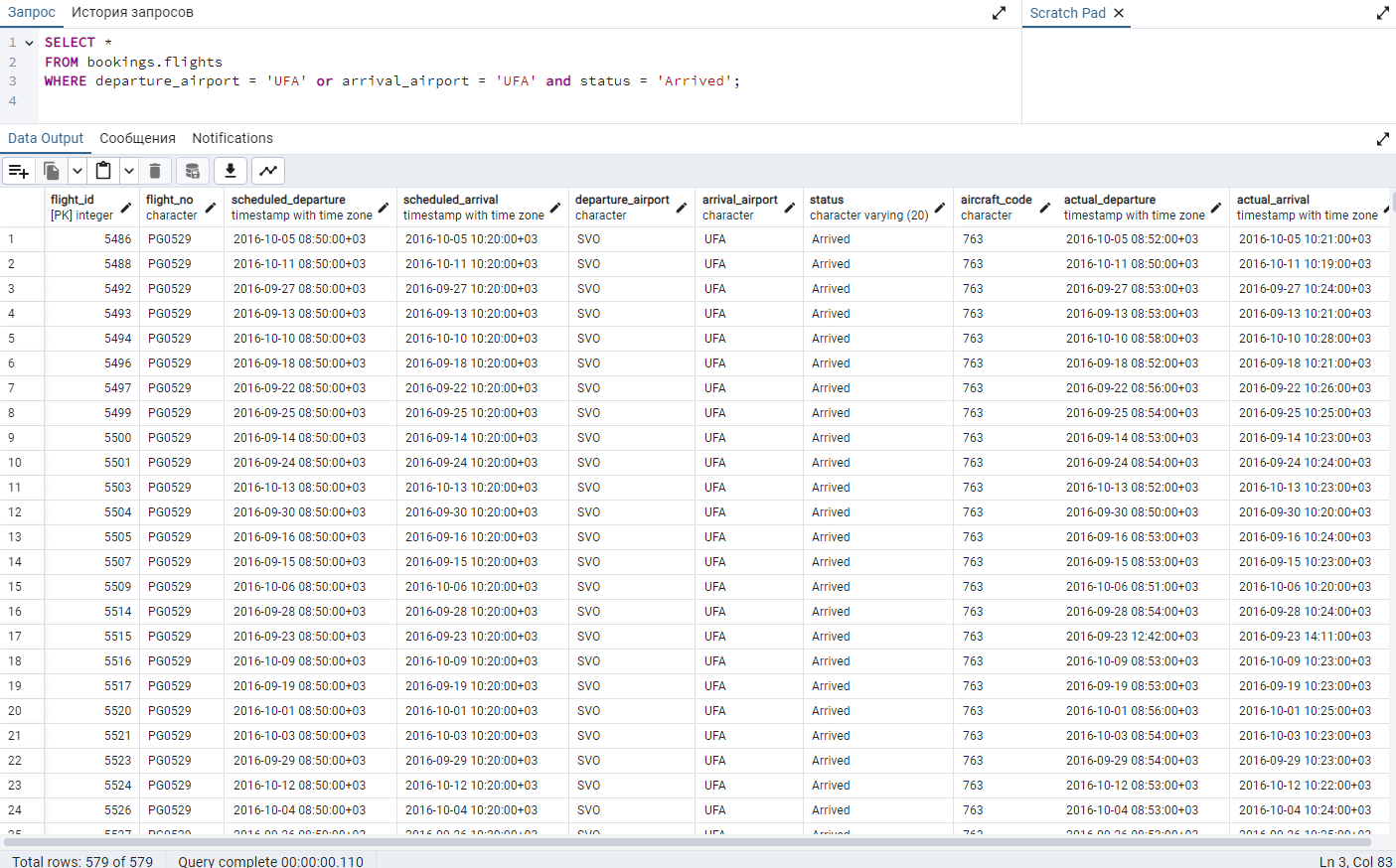
FROM tableName

WHERE condition1 AND|OR condition2;

SELECT \*

FROM bookings.flights

WHERE departure\_airport = 'UFA' or arrival\_airport = 'UFA' and status = 'Arrived';



* Выборка записей, где departure\_airport или arrival\_airport равен UFA и status равен Arrived

1. SELECT col2, col2, ...colN

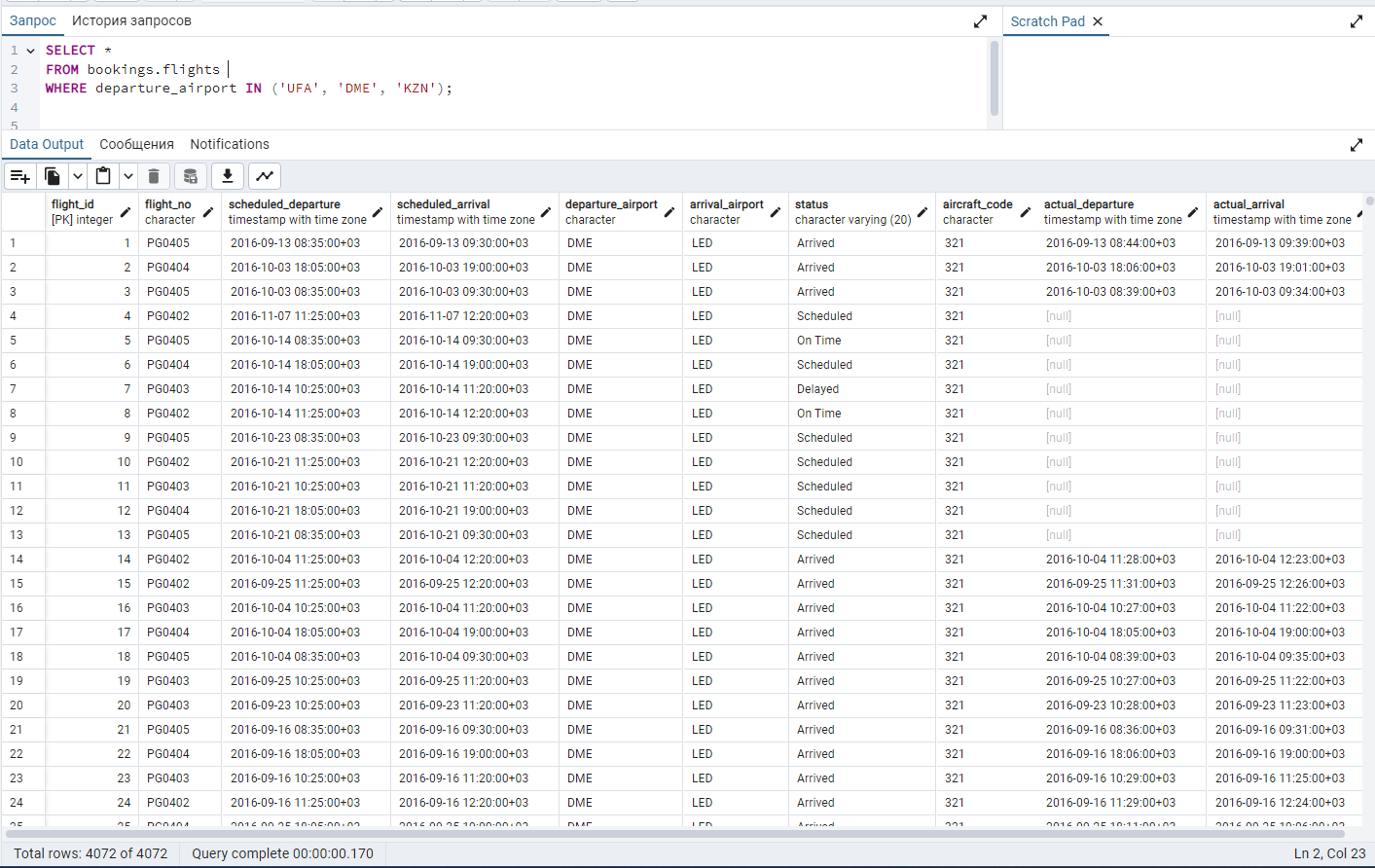
FROM tableName

WHERE colName IN (val1, val2, ...valN);

SELECT \*

FROM bookings.flights

WHERE departure\_airport IN ('UFA', 'DME', 'KZN');



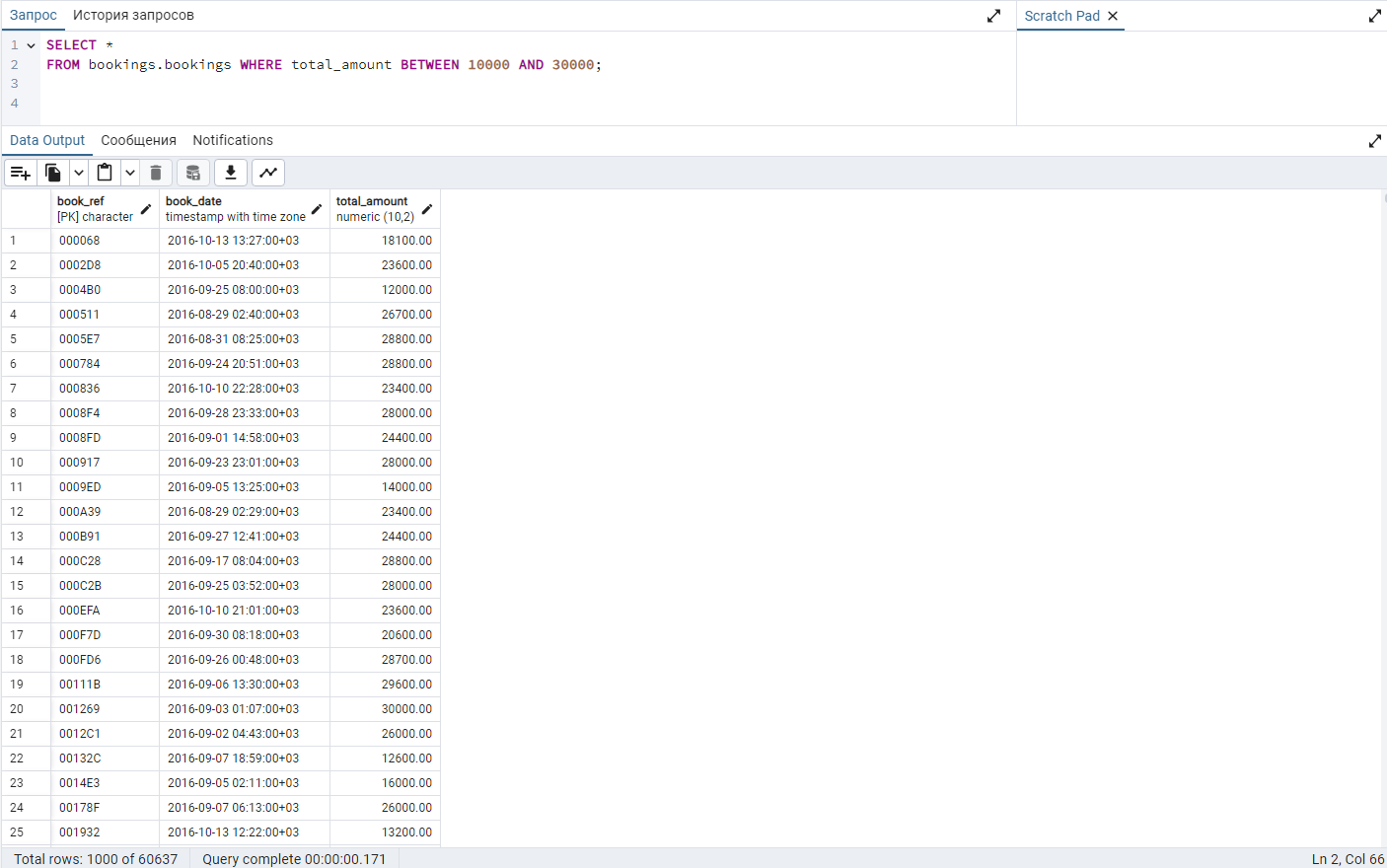
* Выборка записей, где departure\_airport равен UFA, DME или KZN.

1. SELECT col1, col2, ...colN

FROM tableName

WHERE colName BETWEEN val1 AND val2;

SELECT \* FROM bookings.bookings WHERE total\_amount BETWEEN 10000 AND 30000;



* Выборка записей, где total\_amount находится в диапазоне от 10000 до 30000.

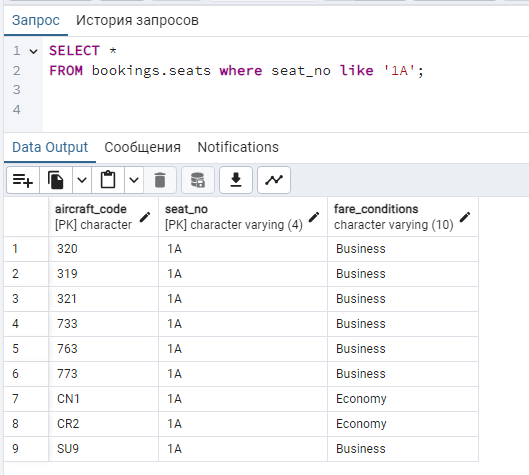
1. SELECT col1, col2, ...colN

FROM tableName

WHERE colName LIKE pattern;

SELECT \*

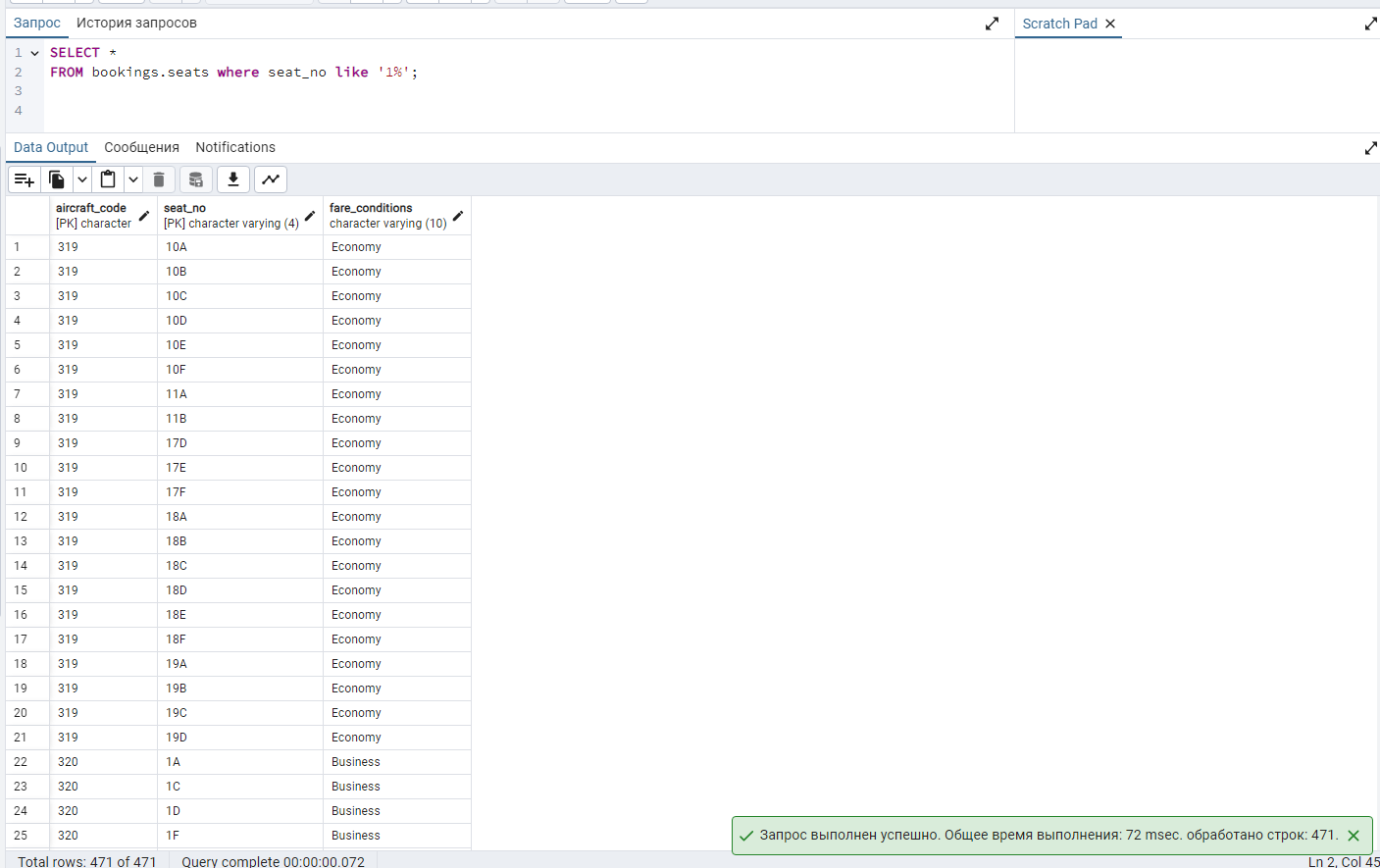
FROM bookings.seats where seat\_no like '1A';



* Выборка записей, где seat\_no соответствует шаблону ‘1A’;

SELECT \*

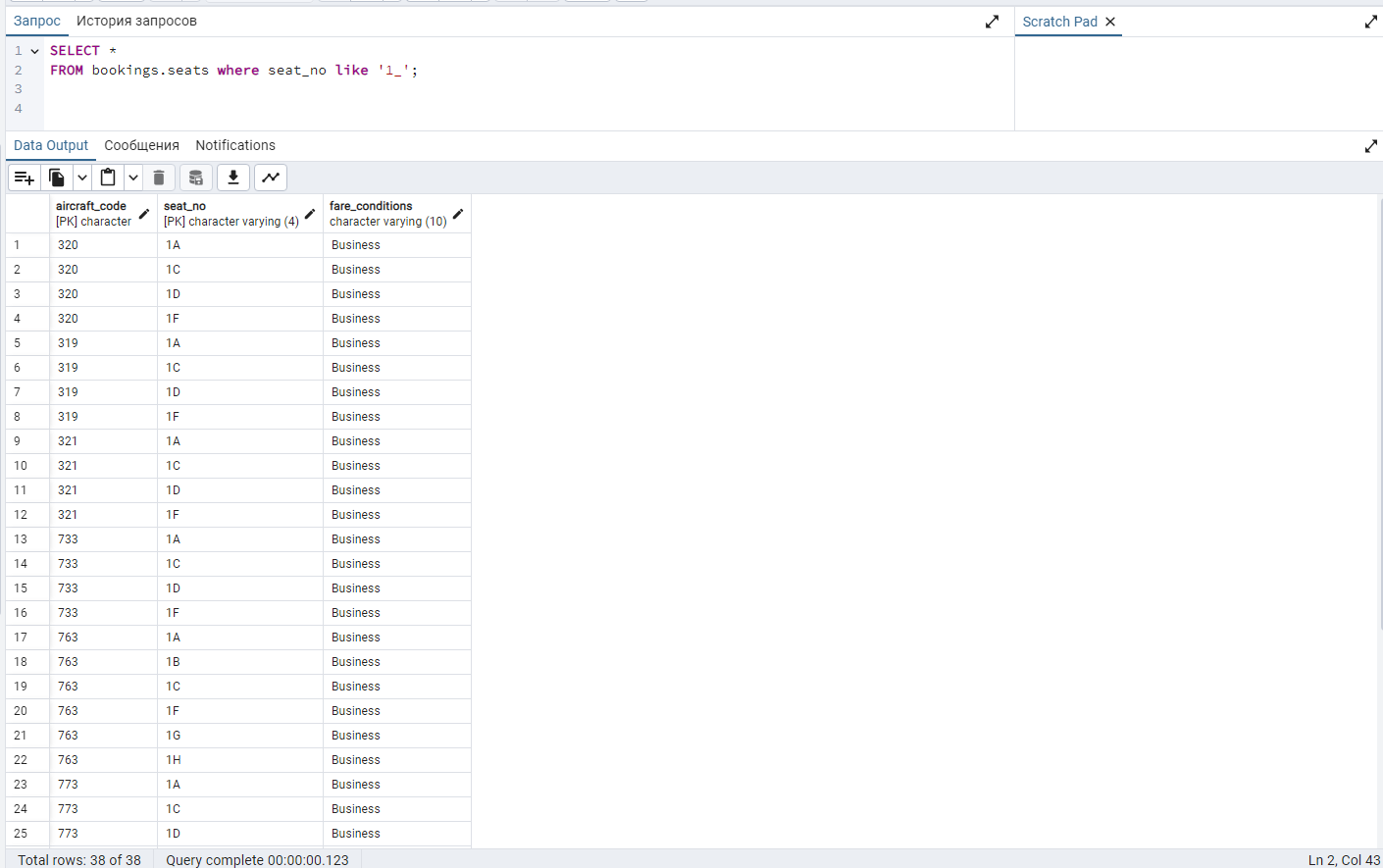
FROM bookings.seats where seat\_no like '1%';



* Выборка записей, где seat\_no начинается с ‘1’;

SELECT \*

FROM bookings.seats where seat\_no like '1\_';



* Выборка записей, где seat\_no начинается с ‘1’ и после должен идти только один символ;

1. Использовать одновременно ASC и DESC для разных столбцов

SELECT col1, col2, ...colN

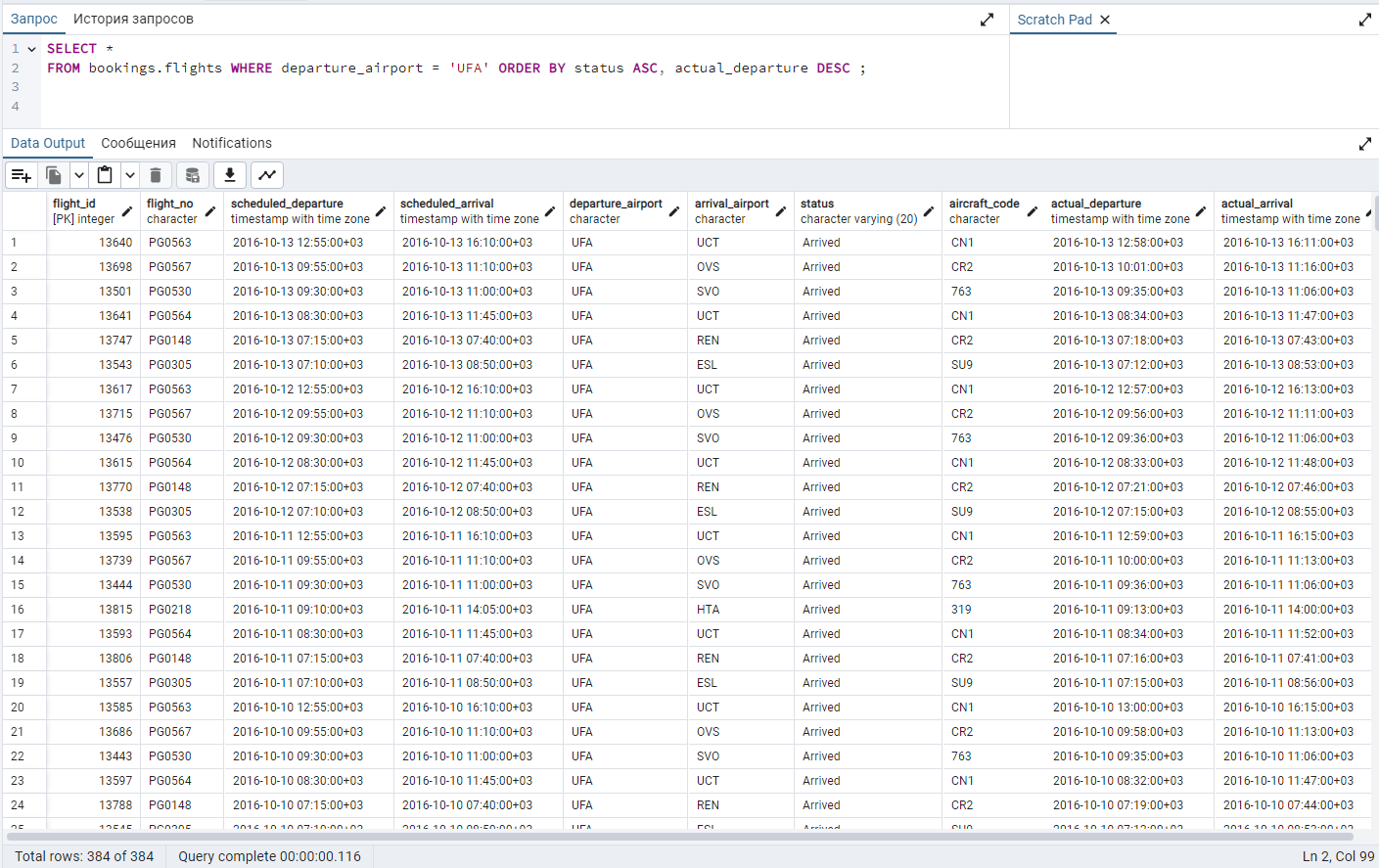
FROM tableName

WHERE condition

ORDER BY colName [ASC|DESC];

SELECT \*

FROM bookings.flights WHERE departure\_airport = 'UFA' ORDER BY status ASC, actual\_departure DESC ;



* Выборка записей, где departure\_airport равен 'UFA', записи сортируются по значению status в порядке возрастания и actual\_departure в порядке убывания.

1. SELECT SUM(colName)

FROM tableName

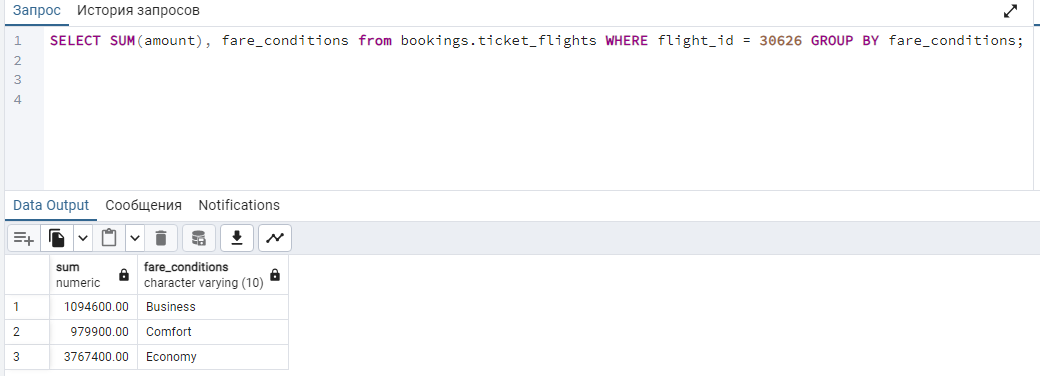
WHERE condition

GROUP BY colName;

SELECT SUM(amount), fare\_conditions from bookings.ticket\_flights

WHERE flight\_id = 30626

GROUP BY fare\_conditions;



* Выборка суммы amount и столбца fre\_conditions, где flight\_id равен 30626, сгруппированных по столбцу fare\_conditions (группируются строки с одинаковыми)

1. SELECT COUNT(colName)

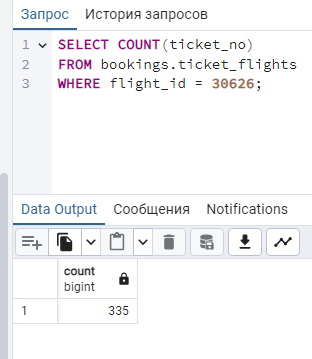
FROM tableName

WHERE condition;

SELECT COUNT(ticket\_no)

FROM bookings.ticket\_flights

WHERE flight\_id = 30626;



* Выборка количества билетов , где flight\_id = 30626

1. SELECT SUM(colName)

FROM tableName

WHERE condition

GROUP BY colName

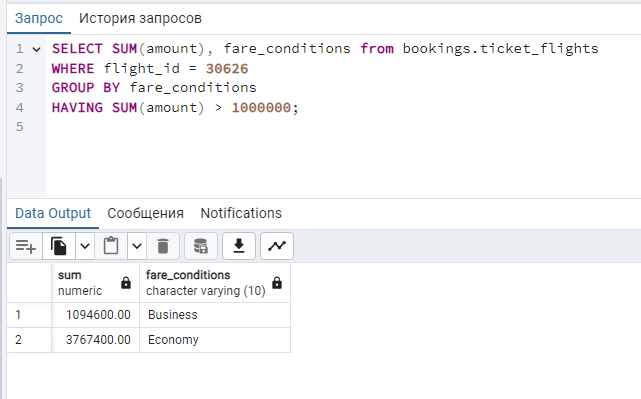
HAVING (function condition);

SELECT SUM(amount), fare\_conditions from bookings.ticket\_flights

WHERE flight\_id = 30626

GROUP BY fare\_conditions

HAVING SUM(amount) > 1000000;



* Выборка суммы amount и столбца fre\_conditions, где flight\_id равен 30626, сгруппированных по столбцу fare\_conditions (группируются строки с одинаковыми), где сумма amount больше 1000000

Задание №2

1. -- создание таблицы, не менее 4х колонок разного типа,

1 колонка первичный ключ, одна необнуляймая

CREATE TABLE tableName (

col1 datatype,

col2 datatype,

...

colN datatype,

PRIMARY KEY (одна или более колонка)

);

CREATE TABLE IF NOT EXISTS bookings.users

(

user\_id int NOT NULL,

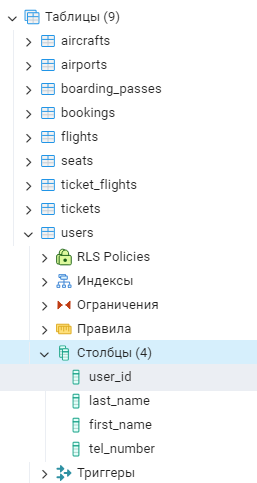
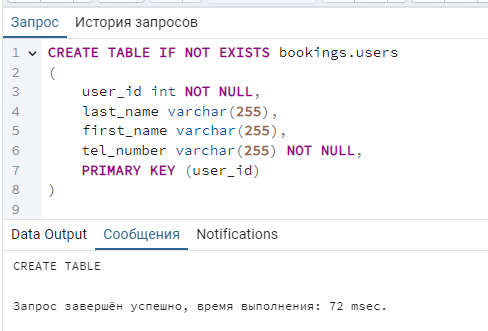
last\_name varchar(255),

first\_name varchar(255),

tel\_number varchar(255) NOT NULL,

PRIMARY KEY (user\_id)

)

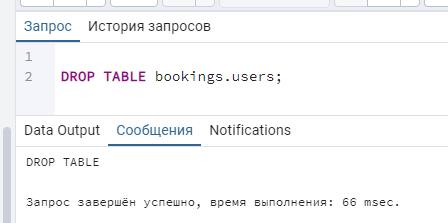


* Создание таблицы user.

1. -- удаление таблицы

DROP TABLE tableName;

DROP TABLE bookings.users;



- удаление таблицы user.

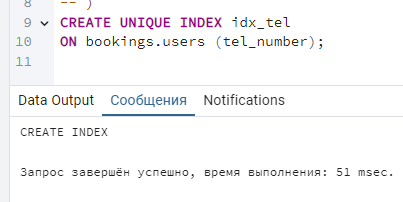
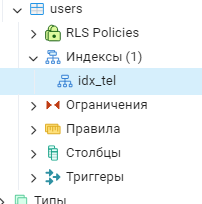
1. -- создание индекса

CREATE UNIQUE INDEX indexName

ON tableName (col1, col2, ...colN);

CREATE UNIQUE INDEX idx\_tel

ON bookings.users (tel\_number);

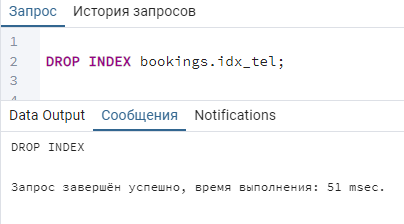
 

* создание индекса с именем idx\_tel в столбце tel\_number таблицы users

1. -- удаление индекса

ALTER TABLE tableName

DROP INDEX indexName;



* Удаление индекса

1. -- получение описания структуры таблицы

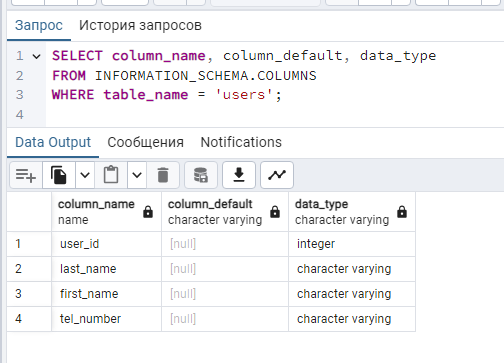
DESC tableName;

Для postgresql запрос будет другой:

SELECT column\_name, column\_default, data\_type

FROM INFORMATION\_SCHEMA.COLUMNS

WHERE table\_name = 'users';



1. -- очистка таблицы

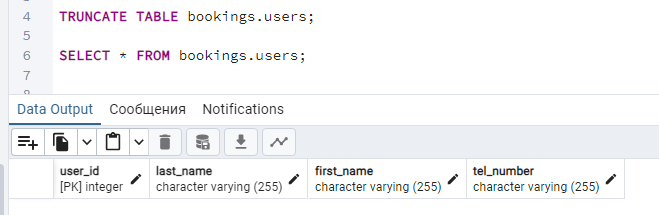
TRUNCATE TABLE tableName;

TRUNCATE TABLE bookings.users;

До выполнения запроса:



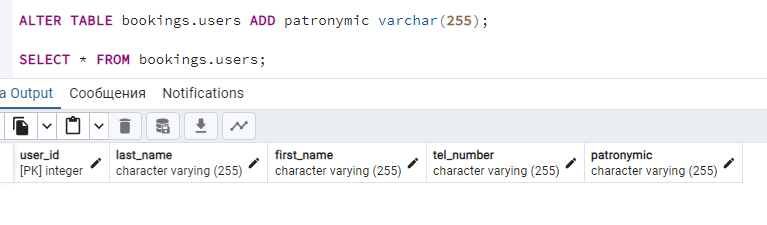
После:



* Очистка записей из таблицы

1. -- выбрать одно из вариантов: добавление/удаление/модификация колонок

ALTER TABLE tableName ADD|DROP|MODIFY colName [datatype];

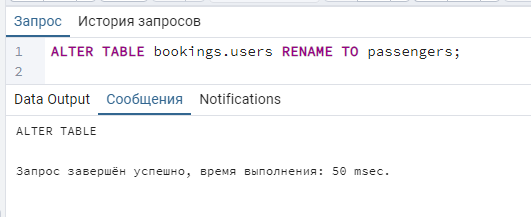


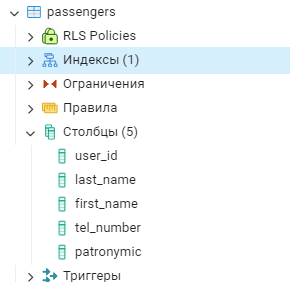
- добавление столбца patronymic в таблицу users

1. -- переименование таблицы

ALTER TABLE tableName RENAME TO newTableName;

ALTER TABLE bookings.users RENAME TO passengers;





1. -- вставка значений

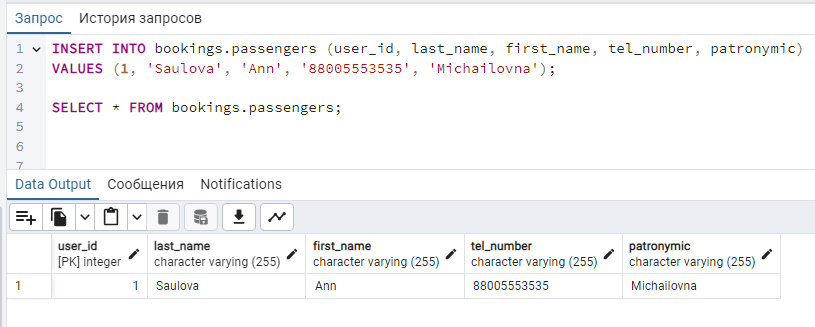
INSERT INTO tableName (col1, col2, ...colN)

VALUES (val1, val2, ...valN)

INSERT INTO bookings.passengers (user\_id, last\_name, first\_name, tel\_number, patronymic)

VALUES (1, 'Saulova', 'Ann', '88005553535', 'Michailovna');

SELECT \* FROM bookings.passengers;



1. -- обновление записей

UPDATE tableName

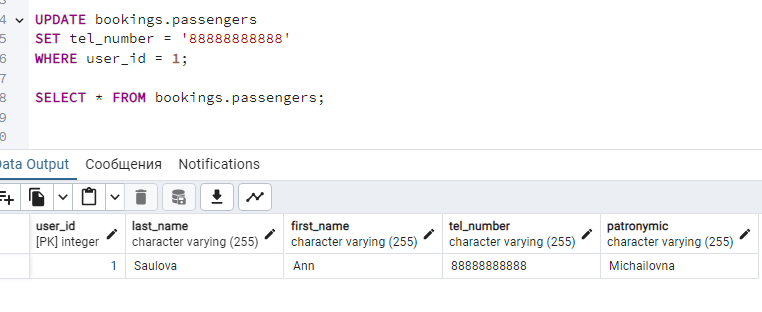
SET col1 = val1, col2 = val2, ...colN = valN

[WHERE condition];

UPDATE bookings.passengers

SET tel\_number = '88888888888'

WHERE user\_id = 1;



- обновление номера телефона у пользователя с user\_id равным 1.

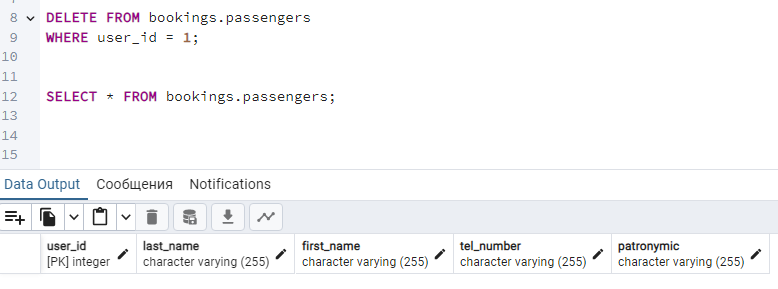
1. -- удаление записей

DELETE FROM tableName

WHERE condition;

DELETE FROM bookings.passengers

WHERE user\_id = 1;



- удаление пользователя с user\_id равным 1.