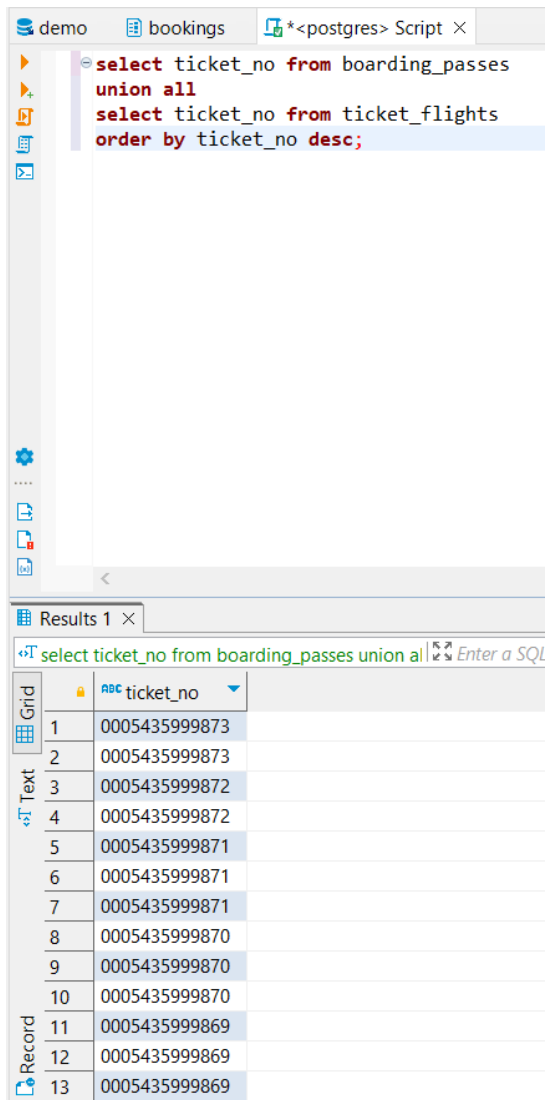


## dump в консоли

```
PS C:\Users\TheMysticSword\Documents\ITIS\sql\hw6> psql -f demo-small-20170815.sql -U postgres
Пароль пользователя postgres:
SET
SET
SET
SET
SET
SET
SET
SET
SET
psql:demo-small-20170815.sql:17: P?P?P?P?P?P?: P+P°P·P° P?P°P?P?C<C: "demo" P?Pч C?C?C%P
чC?C'P?C?PчC'
CREATE DATABASE
Вы подключены к базе данных "demo" как пользователь "postgres".
SET
SET
SET
SET
SET
SET
SET
SET
SET
CREATE SCHEMA
COMMENT
CREATE EXTENSION
COMMENT
SET
CREATE FUNCTION
CREATE FUNCTION
COMMENT
SET
SET
CREATE TABLE
COMMENT
COMMENT
COMMENT
COMMENT
CREATE VIEW
COMMENT
```

# Объединение данных

```
select ticket_no from boarding_passes
union all
select ticket_no from ticket_flights
order by ticket_no desc;
```



The screenshot shows a PostgreSQL client window with a script editor and a results pane. The script editor contains the following SQL query:

```
select ticket_no from boarding_passes
union all
select ticket_no from ticket_flights
order by ticket_no desc;
```

The results pane displays the output of the query, showing 13 rows of ticket numbers. The results are sorted in descending order of ticket number. The first row is 0005435999873, and the last row is 0005435999869.

	ticket_no
1	0005435999873
2	0005435999873
3	0005435999872
4	0005435999872
5	0005435999871
6	0005435999871
7	0005435999871
8	0005435999870
9	0005435999870
10	0005435999870
11	0005435999869
12	0005435999869
13	0005435999869

# Запрос WHERE + LIMIT

```
select passenger_id, passenger_name from tickets
where length(passenger_name) >= 18
limit 5;
```

The screenshot shows a database client interface with a tab labeled "demo" and "bookings". The active tab is "\*<postgres> Script". The SQL query is entered in the script editor:

```
select passenger_id, passenger_name from tickets
where length(passenger_name) >= 18
limit 5;
```

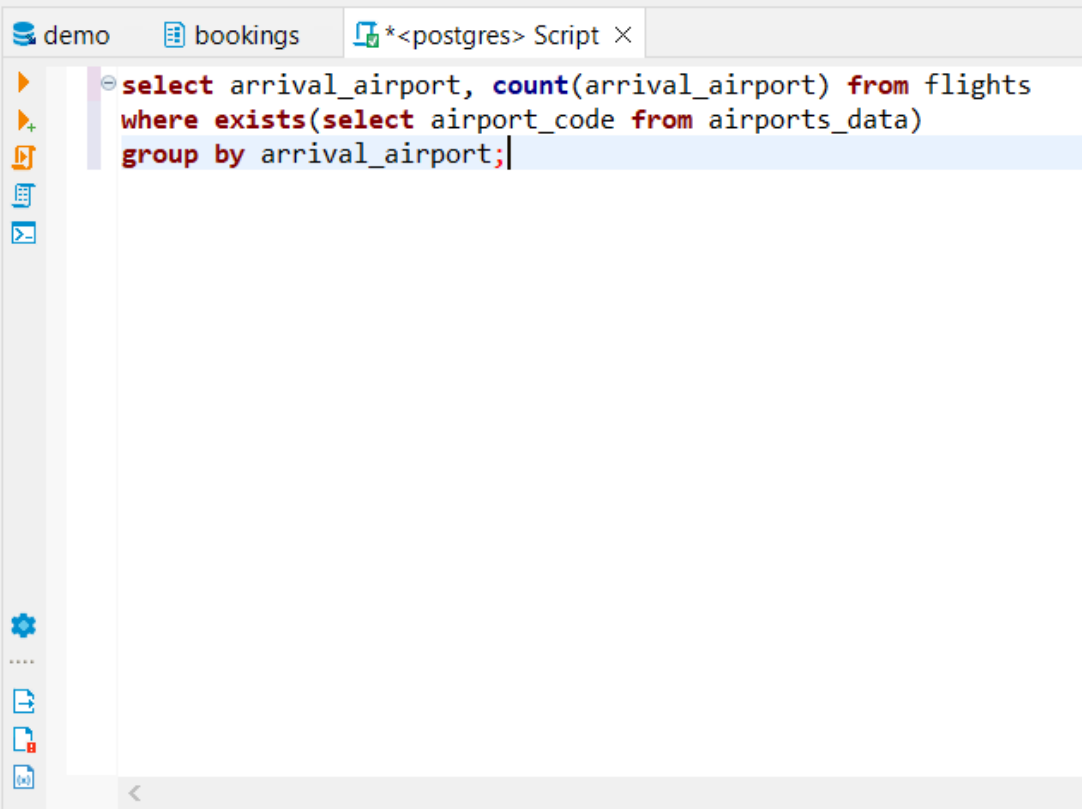
Below the script editor, the results of the query are displayed in a table. The table has two columns: "passenger\_id" and "passenger\_name". The results are as follows:

	passenger_id	passenger_name
1	7277 323748	VALENTINA SOROKINA
2	1683 370090	VALENTINA KAZAKOVA
3	2252 361597	ANASTASIYA SOKOLOVA
4	0752 320983	IRINA ALEKSANDROVA
5	2567 811995	SVETLANA POLYAKOVA

# Аналитический запрос

“Кол-во рейсов в каждый аэропорт, который зарегистрирован в airports\_data”

```
select arrival_airport, count(arrival_airport) from flights
where exists(select airport_code from airports_data)
group by arrival_airport;
```



The screenshot shows a database client window with a tab labeled "demo". Inside, there's a "bookings" section and a "Script" editor. The script contains the SQL query: `select arrival_airport, count(arrival_airport) from flights where exists(select airport_code from airports_data) group by arrival_airport;`. Below the script editor, there's a "flights 1" tab showing the results of the query. The results are displayed in a table with two columns: "arrival\_airport" and "count". The table has 13 rows of data, each with a checkbox icon next to the airport code. The interface also includes a sidebar with various icons and a search bar at the top of the results section.

	arrival_airport	count
1	<input checked="" type="checkbox"/> KXK	17
2	<input checked="" type="checkbox"/> IKT	365
3	<input checked="" type="checkbox"/> REN	305
4	<input checked="" type="checkbox"/> TBW	244
5	<input checked="" type="checkbox"/> NBC	226
6	<input checked="" type="checkbox"/> AER	584
7	<input checked="" type="checkbox"/> KRR	236
8	<input checked="" type="checkbox"/> PKC	26
9	<input checked="" type="checkbox"/> NYA	26
10	<input checked="" type="checkbox"/> EGO	366
11	<input checked="" type="checkbox"/> KLF	122
12	<input checked="" type="checkbox"/> GOJ	357
13	<input checked="" type="checkbox"/> VKT	279

# Уникальные пары

**select** arrival\_airport, departure\_airport **from** flights  
**group by** arrival\_airport, departure\_airport;

The screenshot shows a database client window with a tab labeled "demo". The main pane displays a SQL query: `select arrival_airport, departure_airport from flights group by arrival_airport, departure_airport;`. Below the query editor, a results pane titled "flights 1" shows the output of the query. The results are displayed in a table with two columns: "arrival\_airport" and "departure\_airport". The table contains 12 rows of data, each with a unique pair of airport codes. The interface also includes a sidebar with icons for various database operations and a bottom status bar indicating that 618 rows were fetched in 9ms on 2023-11-30 at 23:52:49.

	arrival_airport	departure_airport
1	URS	KUF
2	DME	GDZ
3	AAQ	EGO
4	GDZ	ROV
5	NOZ	KRR
6	NYM	NSK
7	REN	ESL
8	PYJ	OVB
9	LPK	PES
10	KRR	JOK
11	GOJ	NAL
12	CNN	ASF

618 row(s) fetched - 9ms, on 2023-11-30 at 23:52:49