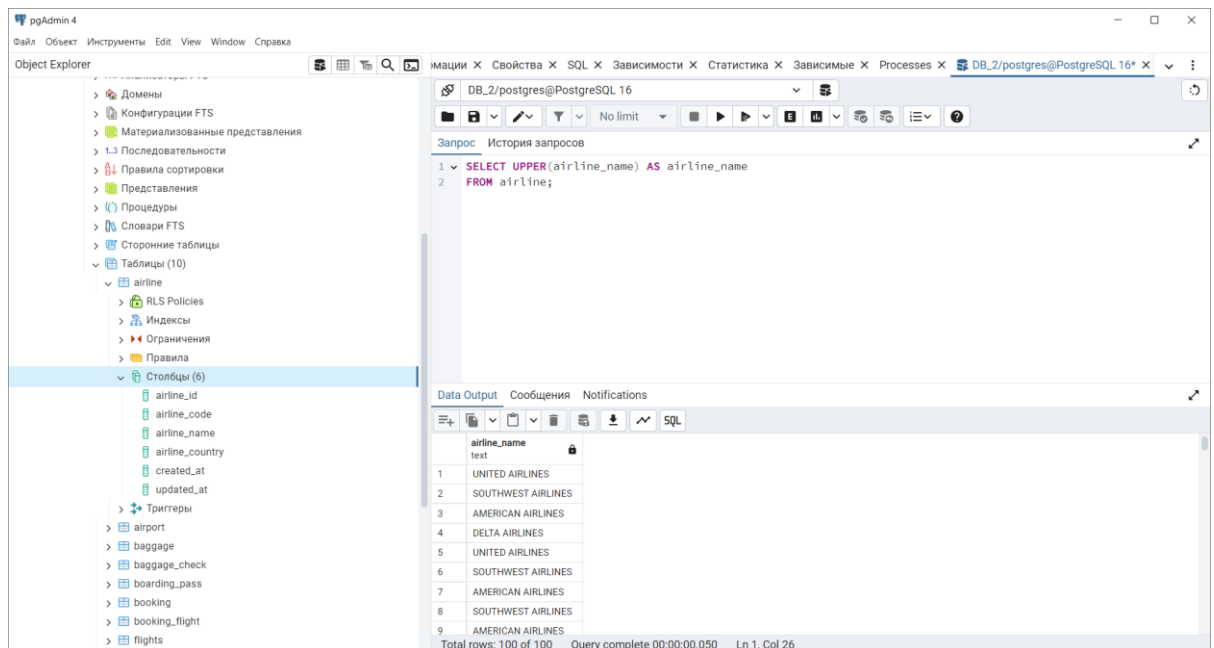


# Laboratory work 4

## Tasks:

1. Retrieve all airline names in uppercase.



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays the database structure, with the 'airline' table selected under 'Tables (10)'. The table's columns are listed: airline\_id, airline\_code, airline\_name, airline\_country, created\_at, and updated\_at. The main pane shows a SQL query:

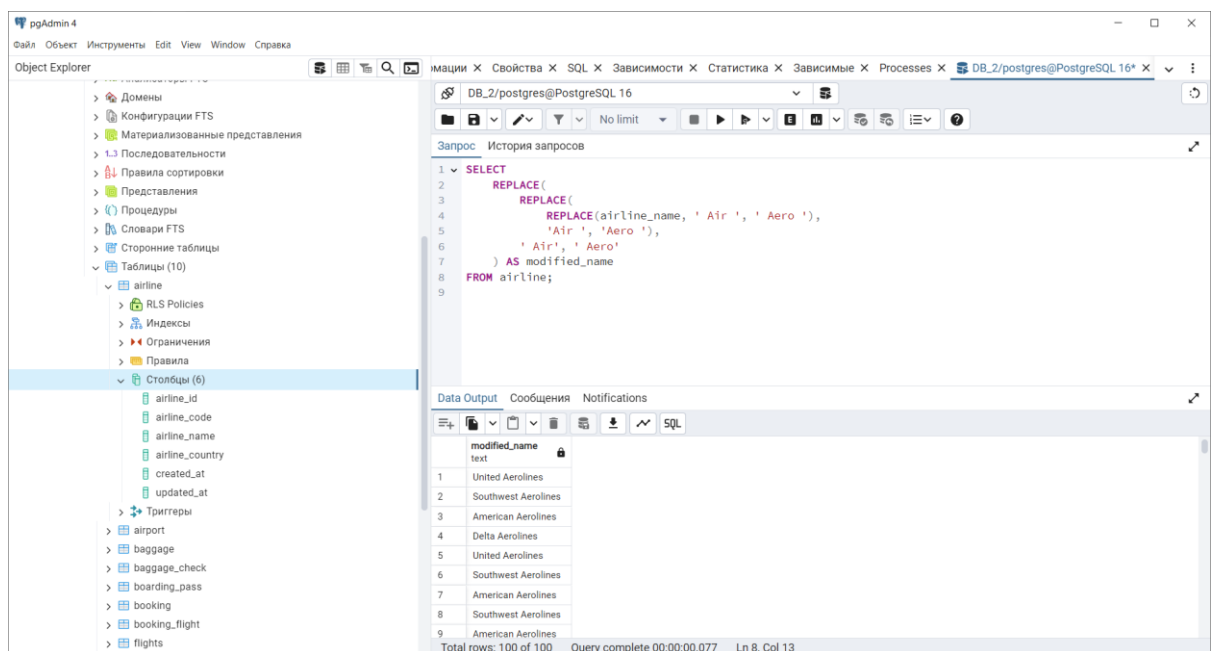
```
1 SELECT UPPER(airline_name) AS airline_name
2 FROM airline;
```

The Data Output pane at the bottom shows the results of the query, displaying a list of airline names in uppercase:

airline_name
UNITED AIRLINES
SOUTHWEST AIRLINES
AMERICAN AIRLINES
DELTA AIRLINES
UNITED AIRLINES
SOUTHWEST AIRLINES
AMERICAN AIRLINES
SOUTHWEST AIRLINES
AMERICAN AIRLINES

Total rows: 100 of 100 Query complete 00:00:00.050 Ln 1, Col 26

2. Replace any occurrence of the word "Air" in airline names with "Aero".



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays the database structure, with the 'airline' table selected under 'Tables (10)'. The table's columns are listed: airline\_id, airline\_code, airline\_name, airline\_country, created\_at, and updated\_at. The main pane shows a SQL query:

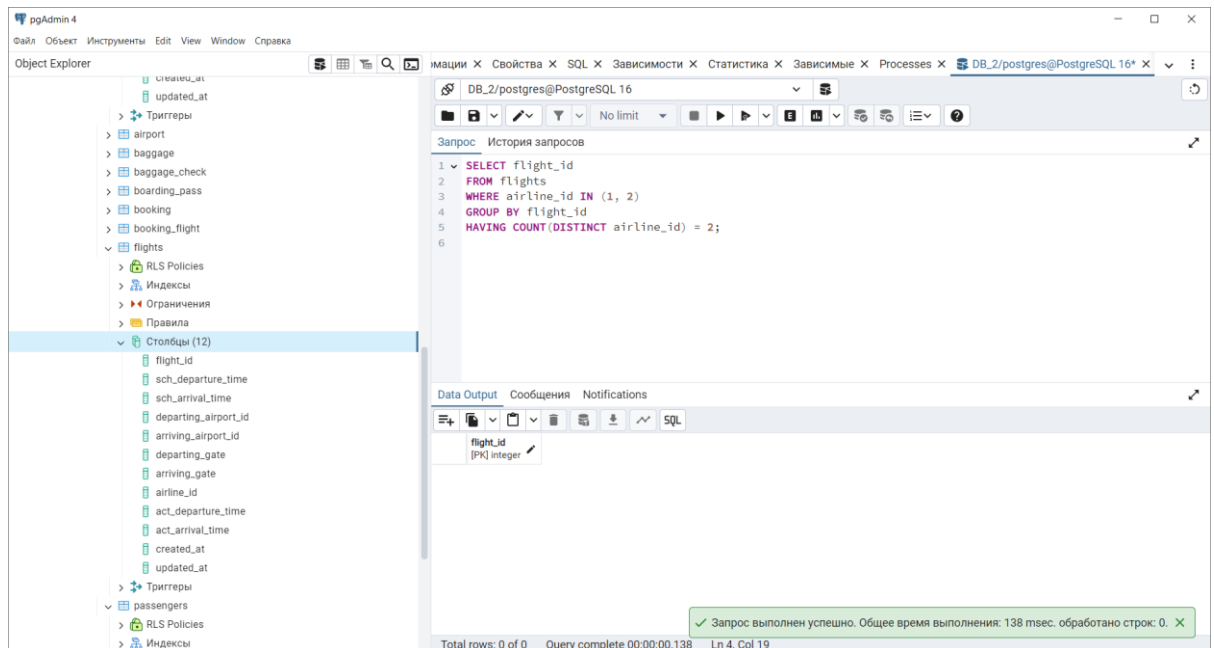
```
1 SELECT
2     REPLACE(
3         REPLACE(
4             REPLACE(airline_name, ' Air ', ' Aero '),
5             'Air ', ' Aero '),
6             ' Air', ' Aero'
7         ) AS modified_name
8 FROM airline;
```

The Data Output pane at the bottom shows the results of the query, displaying a list of modified airline names:

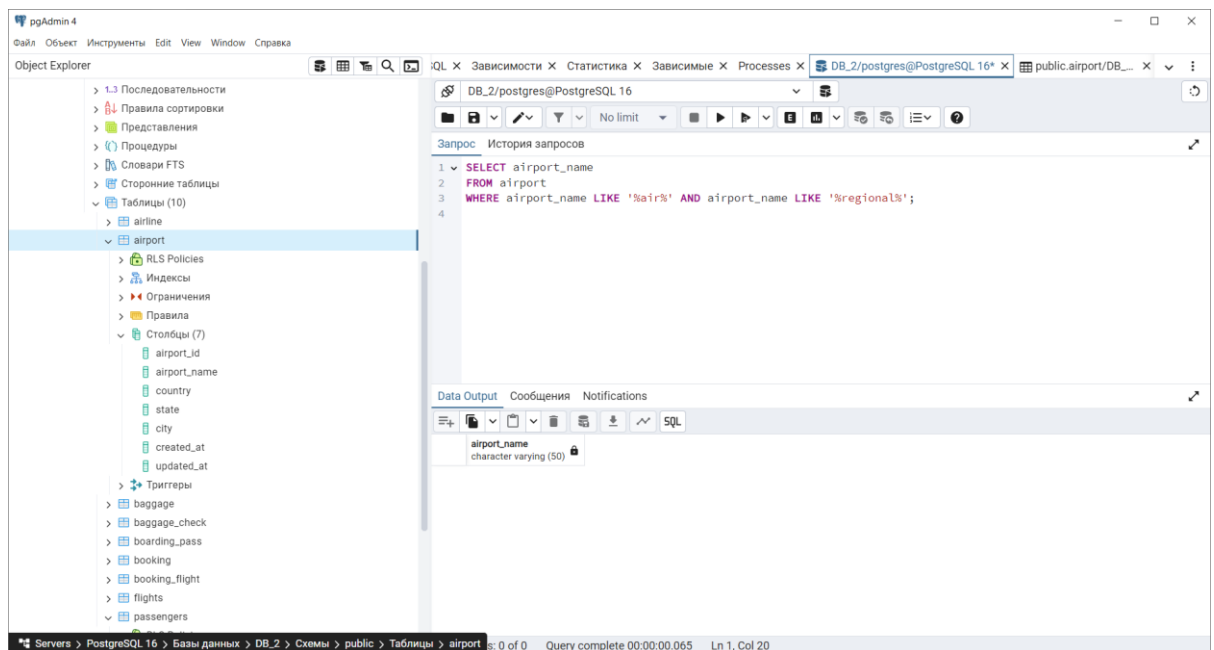
modified_name
United Aerolines
Southwest Aerolines
American Aerolines
Delta Aerolines
United Aerolines
Southwest Aerolines
American Aerolines
Southwest Aerolines
American Aerolines

Total rows: 100 of 100 Query complete 00:00:00.077 Ln 8, Col 13

3. Find all flight numbers that coordinates with both airline 1 and airline 2.



4. Retrieve airports that contain the word "Reginal" and "Air" in their names.



5. Retrieve passenger names and format their birth dates as 'Month DD, YYYY'.

pgAdmin 4

Object Explorer

- airport
- baggage
- baggage\_check
- boarding\_pass
- booking
- booking\_flight
- flights
- passengers
  - RLS Policies
  - Индексы
  - Ограничения
  - Правила
  - Столбцы (10)
    - passenger\_id
    - first\_name
    - last\_name
    - date\_of\_birth
    - gender
    - country\_of\_citizenship
    - country\_of\_residence
    - passport\_number
    - created\_at
    - updated\_at
  - Триггеры
  - security\_check
  - Типы
  - Триггерные функции
  - Функции
  - Шаблоны FTS
- Языки
- postgres

DB\_2/postgres@PostgreSQL 16

Запрос История запросов

```

1 SELECT first_name, last_name, TO_CHAR(date_of_birth, 'Month DD, YYYY') AS formatted_birth_date
2 FROM passengers;
3

```

Data Output Сообщения Notifications

	first_name character varying (50)	last_name character varying (50)	formatted_birth_date text
1	John	Doe	May 15, 1985
2	Jane	Smith	July 20, 1990
3	Mike	Johnson	March 12, 1982
4	Emily	Davis	September 30, 1995
5	Chris	Brown	November 25, 1988
6	Sarah	Wilson	December 01, 1992
7	David	Garcia	April 10, 1986
8	Laura	Martinez	August 22, 1983
9	James	Rodriguez	October 16, 1994

Total rows: 100 of 100 Query complete 00:00:00.105 Ln 3, Col 1

6. Find flight numbers that have been delayed based on the actual arrival time.

pgAdmin 4

Object Explorer

- airport
- baggage
- baggage\_check
- boarding\_pass
- booking
- booking\_flight
- flights
- passengers
  - RLS Policies
  - Индексы
  - Ограничения
  - Правила
  - Столбцы (10)
    - passenger\_id
    - first\_name
    - last\_name
    - date\_of\_birth
    - gender
    - country\_of\_citizenship
    - country\_of\_residence
    - passport\_number
    - created\_at
    - updated\_at
  - Триггеры
  - security\_check
  - Типы
  - Триггерные функции
  - Функции
  - Шаблоны FTS
- Языки
- postgres

DB\_2/postgres@PostgreSQL 16

Запрос История запросов

```

1 SELECT flight_id
2 FROM flights
3 WHERE act_arrival_time > sch_arrival_time;
4

```

Data Output Сообщения Notifications

	flight_id [PK] integer
1	1001
2	1002
3	1003
4	1004
5	1005
6	1006
7	1007
8	1008
9	1009

Total rows: 100 of 100 Query complete 00:00:00.061 Ln 4, Col 1

7. Create a query that divides passengers into age groups like 'Young' and 'Adult' based on their birth date. Young passengers age between 18 and 35, Adult passengers age between 36 and 55.

Query:

```

1 SELECT
2   first_name,
3   last_name,
4   date_of_birth,
5   CASE
6     WHEN (EXTRACT(YEAR FROM AGE(date_of_birth)) BETWEEN 18 AND 35) THEN 'Young'
7     WHEN (EXTRACT(YEAR FROM AGE(date_of_birth)) BETWEEN 36 AND 55) THEN 'Adult'
8     ELSE 'Other'
9   END AS age_group
10  FROM
11    passengers
12  WHERE
13    date_of_birth IS NOT NULL;
14

```

Data Output:

	first_name	last_name	date_of_birth	age_group
1	John	Doe	1985-05-15	Adult
2	Jane	Smith	1990-07-20	Young
3	Mike	Johnson	1982-03-12	Adult
4	Emily	Davis	1995-09-30	Young
5	Chris	Brown	1988-11-25	Young
6	Sarah	Wilson	1992-12-01	Young
7	David	Garcia	1986-04-10	Adult
8	Laura	Martinez	1983-08-22	Adult
9	James	Rodriguez	1994-10-16	Young

Total rows: 100 of 100 Query complete 00:00:00.069 Ln 14, Col 1

8. Create a query that categorizes ticket prices based on their price as "Cheap," "Medium" or "Expensive."

Query:

```

1 SELECT
2   ticket_price,
3   CASE
4     WHEN ticket_price < 100 THEN 'Cheap'
5     WHEN ticket_price BETWEEN 100 AND 500 THEN 'Medium'
6     ELSE 'Expensive'
7   END AS price_category
8  FROM
9    booking;
10

```

Data Output:

	ticket_price	price_category
1	165.00	Medium
2	220.55	Medium
3	192.50	Medium
4	242.83	Medium
5	198.00	Medium
6	176.55	Medium
7	225.50	Medium
8	209.83	Medium
9	187.00	Medium

Total rows: 100 of 100 Query complete 00:00:00.058 Ln 9, Col 12

9. Find number of airline names in each airline country.

pgAdmin 4

Файл Объект Инструменты Edit View Window Справка

Object Explorer

- Материализованные представления
- 1.3 Последовательности
- Правила сортировки
- Представления
- Процедуры
- Словари FTS
- Сторонние таблицы
- Таблицы (10)
  - airline
  - airport
  - baggage
  - baggage\_check
  - boarding\_pass
  - booking
    - RLS Policies
    - Индексы
    - Ограничения
    - Правила
    - Столбцы (8)
      - booking\_id
      - flight\_id
      - passenger\_id
      - booking\_platform
      - created\_at
      - updated\_at
      - status
      - ticket\_price
    - Триггеры
    - booking\_flight
    - flightite

DB\_2/postgres@PostgreSQL 16

Запрос История запросов

```

1 SELECT
2   airline_country,
3   COUNT(airline_name) AS airline_count
4 FROM
5   airline
6 GROUP BY
7   airline_country;
8

```

Data Output Сообщения Notifications

airline_country	airline_count
Turkey	1
Nigeria	1
Burkina Faso	1
Indonesia	12
El Salvador	1
Zimbabwe	1
Hungary	1
Russia	4
China	25

Total rows: 37 of 37 Query complete 00:00:00.055 Ln 7, Col 13

✓ Запрос выполнен успешно. Общее время выполнения: 55 msec. обработано строк: 37. ✕

10. Find flights that arrived late according to their actual arrival time compared to the scheduled arrival time.

S

pgAdmin 4

Файл Объект Инструменты Edit View Window Справка

Object Explorer

- airport
- baggage
- baggage\_check
- boarding\_pass
- booking
- booking\_flight
- flights
  - RLS Policies
  - Индексы
  - Ограничения
  - Правила
  - Столбцы (12)
    - flight\_id
    - sch\_departure\_time
    - sch\_arrival\_time
    - departing\_airport\_id
    - arriving\_airport\_id
    - departing\_gate
    - arriving\_gate
    - airline\_id
    - act\_departure\_time
    - act\_arrival\_time
    - created\_at
    - updated\_at
  - Триггеры
  - passengers
  - security\_check
  - Типы
  - Триггерные функции
  - Функции

DB\_2/postgres@PostgreSQL 16

Запрос История запросов

```

1 SELECT
2   flight_id,
3   sch_arrival_time,
4   act_arrival_time
5 FROM
6   flights
7 WHERE
8   act_arrival_time > sch_arrival_time;
9

```

Data Output Сообщения Notifications

flight_id	sch_arrival_time	act_arrival_time
1001	2024-10-01 10:00:00	2024-10-01 10:05:00
1002	2024-10-01 11:45:00	2024-10-01 11:50:00
1003	2024-10-01 09:15:00	2024-10-01 09:20:00
1004	2024-10-01 15:20:00	2024-10-01 15:25:00
1005	2024-10-01 08:45:00	2024-10-01 08:50:00
1006	2024-10-02 14:00:00	2024-10-02 14:05:00
1007	2024-10-02 17:50:00	2024-10-02 17:55:00
1008	2024-10-02 13:20:00	2024-10-02 13:25:00
1009	2024-10-02 12:30:00	2024-10-02 12:35:00

Total rows: 100 of 100 Query complete 00:00:00.067 Ln 9, Col 27

✓ Запрос выполнен успешно. Общее время выполнения: 67 msec. обработано строк: 100. ✕