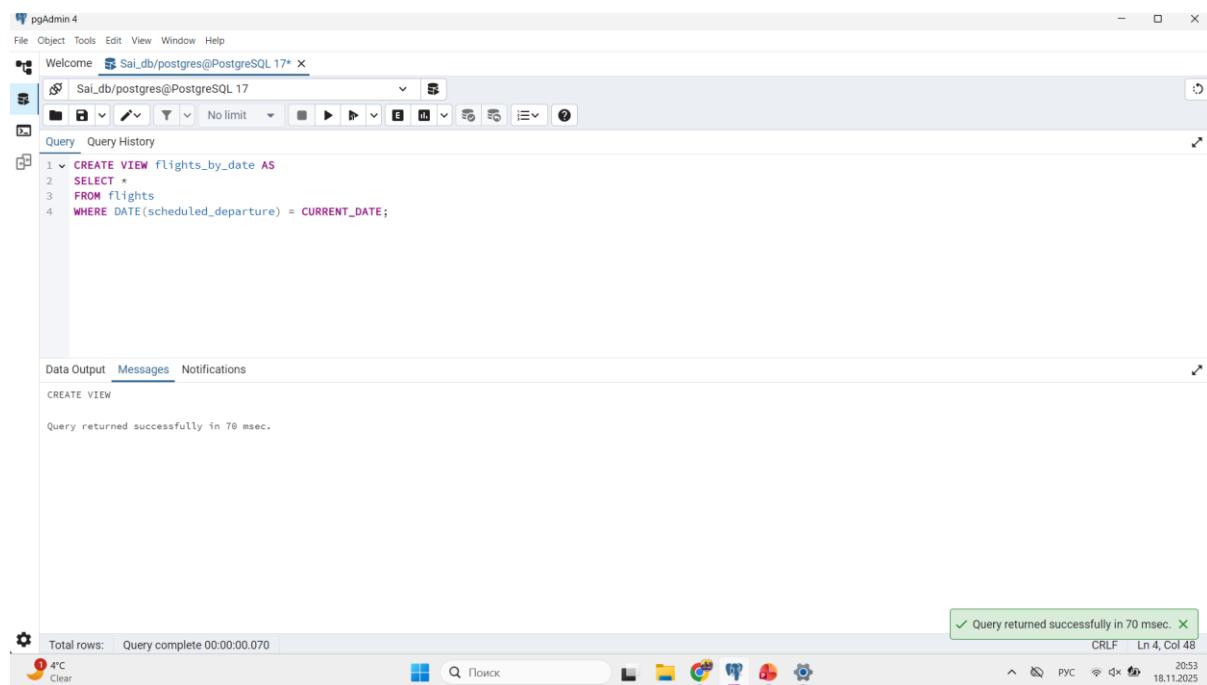


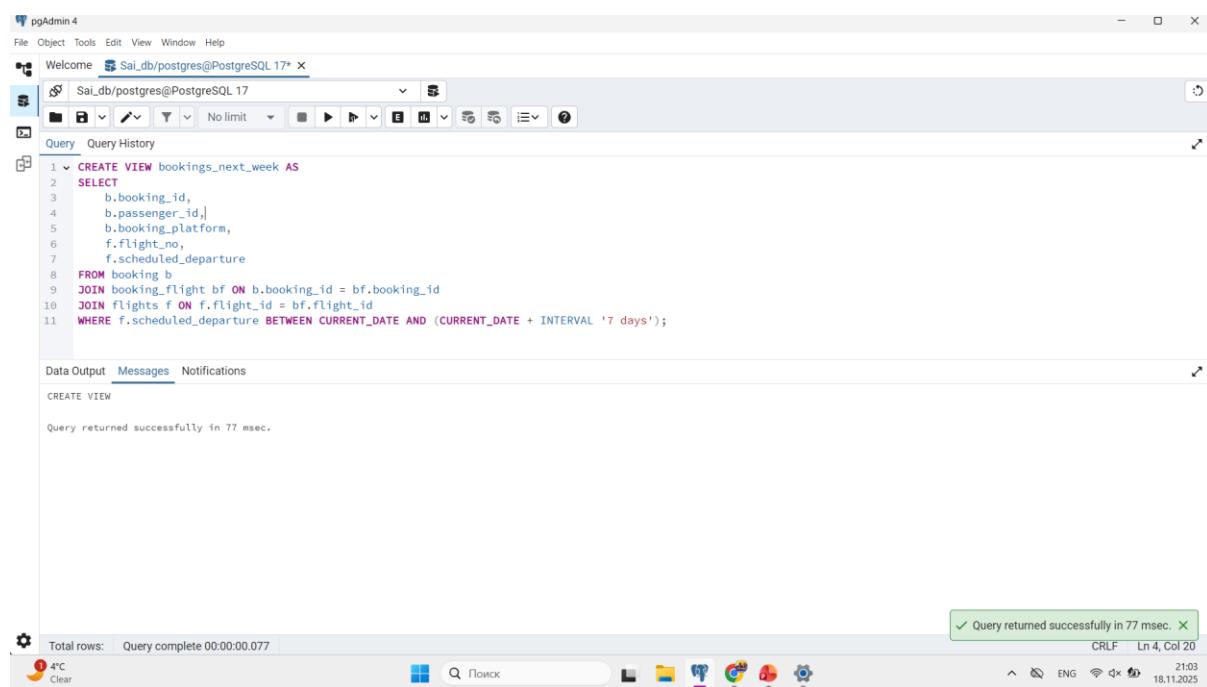
1. Create a view to show details of all flights that are departing on a specific date.



```
CREATE VIEW flights_by_date AS
SELECT *
FROM flights
WHERE DATE(scheduled_departure) = CURRENT_DATE;
```

The screenshot shows the pgAdmin 4 interface with a query editor window. The query is to create a view named 'flights_by_date' that selects all columns from the 'flights' table where the scheduled departure date matches the current date. The status bar at the bottom right indicates 'Query returned successfully in 70 msec.'

2. Create a view that shows bookings for flights scheduled to depart within the next week.



```
CREATE VIEW bookings_next_week AS
SELECT
    b.booking_id,
    b.passenger_id,
    b.booking_platform,
    f.flight_no,
    f.scheduled_departure
FROM booking b
JOIN booking_flight bf ON b.booking_id = bf.booking_id
JOIN flights f ON f.flight_id = bf.flight_id
WHERE f.scheduled_departure BETWEEN CURRENT_DATE AND (CURRENT_DATE + INTERVAL '7 days');
```

The screenshot shows the pgAdmin 4 interface with a query editor window. The query is to create a view named 'bookings_next_week' that selects specific columns from the 'booking', 'booking_flight', and 'flights' tables, filtering for bookings made within the next seven days. The status bar at the bottom right indicates 'Query returned successfully in 77 msec.'

3. Create a view to show the top 5 most popular flight routes based on the number of bookings.

The screenshot shows the pgAdmin 4 interface. In the top-left corner, it says "pgAdmin 4". The main window title is "Welcome Sai_db/postgres@PostgreSQL 17*". Below the title, there's a toolbar with various icons. The main area is titled "Query" and contains the following SQL code:

```
1 ✓ CREATE VIEW top_5_routes AS
2   SELECT
3     f.departure_airport_id,
4     f.arrival_airport_id,
5     COUNT(b.booking_id) AS total_bookings
6   FROM flights f
7   JOIN booking_flight bf ON f.flight_id = bf.flight_id
8   JOIN booking b ON b.booking_id = bf.booking_id
9   GROUP BY f.departure_airport_id, f.arrival_airport_id
10  ORDER BY total_bookings DESC
11  LIMIT 5;
```

Below the code, there are tabs for "Data Output", "Messages", and "Notifications". The "Messages" tab is selected, showing the message "CREATE VIEW". At the bottom of the pgAdmin window, there's a status bar with "Total rows: Query complete 00:00:00.110" and a timestamp "21:08 18.11.2025".

4. Create a view that lists all flights for a specific airline.

The screenshot shows the pgAdmin 4 interface. In the top-left corner, it says "pgAdmin 4". The main window title is "Welcome Sai_db/postgres@PostgreSQL 17*". Below the title, there's a toolbar with various icons. The main area is titled "Query" and contains the following SQL code:

```
1 ✓ CREATE VIEW flights_by_airline AS
2   SELECT flight_no, scheduled_departure, status
3   FROM flights
4   WHERE airline_id = 1;
```

Below the code, there are tabs for "Data Output", "Messages", and "Notifications". The "Messages" tab is selected, showing the message "CREATE VIEW". At the bottom of the pgAdmin window, there's a status bar with "Total rows: Query complete 00:00:00.070" and a timestamp "21:09 18.11.2025".

5. Modify the view created in task 4 to show only flights departing within the next 7 days for a specific airline.

The screenshot shows the pgAdmin 4 interface. In the top navigation bar, 'File', 'Object', 'Tools', 'Edit', 'View', 'Window', and 'Help' are visible. The title bar says 'pgAdmin 4' and 'Welcome Sai_db/postgres@PostgreSQL 17*'. The main window has a toolbar with various icons. Below the toolbar, there are tabs for 'Query' and 'Query History'. A code editor window displays the following SQL query:

```

1 ✓ CREATE OR REPLACE VIEW flights_by_airline AS
2   SELECT flight_no, scheduled_departure, status
3   FROM flights
4   WHERE airline_id = 1
5     AND scheduled_departure BETWEEN CURRENT_DATE AND (CURRENT_DATE + INTERVAL '7 days');

```

Below the code editor, there are three tabs: 'Data Output', 'Messages', and 'Notifications'. The 'Messages' tab is selected, showing the message 'CREATE VIEW'. At the bottom of the pgAdmin window, there is a status bar with 'Total rows: Query complete 00:00:00.080' and a system tray with icons for temperature, battery, and network.

6. Create a view to show flights that are delayed by more than 24 hours.

The screenshot shows the pgAdmin 4 interface. The top navigation bar and title bar are identical to the previous screenshot. The main window shows the creation of a new view. The code editor contains the following SQL query:

```

1 ✓ CREATE VIEW delayed_over_24h AS
2   SELECT
3     flight_no,
4     scheduled_departure,
5     actual_departure
6   FROM flights
7   WHERE actual_departure > (scheduled_departure + INTERVAL '24 hours');

```

The 'Messages' tab in the pgAdmin interface shows the message 'CREATE VIEW'. The status bar at the bottom indicates 'Total rows: Query complete 00:00:00.071'.

7. Create a view in which you can display the full name and country of origin of passengers who made bookings on Leffler-Thompson platform. Then show the list of that passengers.

pgAdmin 4

Welcome Sai_db/postgres@PostgreSQL 17*

```

1 CREATE VIEW lt_passengers AS
2   SELECT
3     p.first_name || ' ' || p.last_name AS full_name,
4     p.country_of_citizenship
5   FROM passengers p
6   JOIN booking b ON p.passenger_id = b.passenger_id
7   WHERE b.booking_platform = 'Leffler-Thompson';
8

```

Data Output Messages Notifications

CREATE VIEW

Query returned successfully in 77 msec.

Total rows: Query complete 00:00:00.077

4°C Clear

pgAdmin 4

Welcome Sai_db/postgres@PostgreSQL 17*

```

1 SELECT * FROM lt_passengers;

```

Data Output Messages Notifications

	fullName	country_of_citizenship
1	Philbert Shambroke	Colombia

Showing rows: 1 to 1 Page No: 1 of 1 14 <> >> |

✓ Successfully run. Total query runtime: 98 msec. 1 rows affected. CRLF Ln 1, Col 29

4°C Clear

8. Create a view that shows top 10 most visited countries.

The screenshot shows the pgAdmin 4 interface. In the top navigation bar, 'File', 'Object', 'Tools', 'Edit', 'View', 'Window', and 'Help' are visible. The title bar says 'pgAdmin 4'. Below the title bar, it says 'Welcome Sai_db/postgres@PostgreSQL 17*'. The main area has a toolbar with various icons. A 'Query' tab is selected, showing the following SQL code:

```
1 ✓ CREATE VIEW top_10_visited_countries AS
2   SELECT
3     arrival_airport_id AS destination,
4     COUNT(*) AS visits
5   FROM flights
6   GROUP BY arrival_airport_id
7   ORDER BY visits DESC
8   LIMIT 10;|
```

Below the code, there are tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Messages' tab is selected, showing 'CREATE VIEW'. Underneath, it says 'Query returned successfully in 94 msec.'

At the bottom of the window, there is a status bar with 'Total rows: Query complete 00:00:00.094', 'AW01 -1.41%', and a timestamp '21:17 18.11.2025'.

9. Update any of the created views by adding new information in the view table. Show results.

The screenshot shows the pgAdmin 4 interface. In the top navigation bar, 'File', 'Object', 'Tools', 'Edit', 'View', 'Window', and 'Help' are visible. The title bar says 'pgAdmin 4'. Below the title bar, it says 'Welcome Sai_db/postgres@PostgreSQL 17*'. The main area has a toolbar with various icons. A 'Query' tab is selected, showing the following SQL code:

```
1 ✓ CREATE VIEW flights_by_airline AS
2   SELECT
3     flight_no,
4     scheduled_departure,
5     scheduled_arrival,
6     departing_gate,
7     arriving_gate,
8     status
9   FROM flights
10  WHERE airline_id = 1;|
```

Below the code, there are tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Messages' tab is selected, showing 'CREATE VIEW'. Underneath, it says 'Query returned successfully in 70 msec.'

At the bottom of the window, there is a status bar with 'Total rows: Query complete 00:00:00.070', 'Tempt to drop Thursday', and a timestamp '21:21 18.11.2025'.

pgAdmin 4

Welcome Sai_db/postgres@PostgreSQL 17*

Query History

```
1  SELECT * FROM flights_by_airline;
```

Data Output Messages Notifications

	flight_no	scheduled_departure	scheduled_arrival	departing_gate	arriving_gate	status
1	BR-PE	2024-01-16	2023-06-02	1891	5429	Delayed
2	MZ-G	2023-09-21	2023-11-29	[null]	9380	Delayed
3	AU-NT	2023-03-29	2023-05-06	[null]	13	Delayed
4	FR-K	2023-12-26	2023-04-30	144	23	Delayed
5	US-VT	2023-08-28	2023-08-02	335	515	Delayed
6	PH-BUK	2023-08-20	2023-10-24	919	1330	Delayed
7	SD-01	2023-12-02	2023-12-17	897	1147	Delayed
8	PA-8	2023-03-19	2023-08-21	150	38	Boarding
9	NA-KU	2023-06-26	2023-12-04	2000	996	Delayed
10	US-CT	2023-11-07	2023-05-03	30	100	Boarding
11	RU-CU	2023-04-16	2023-09-15	558	33	Boarding
12	BE-VAN	2023-05-03	2023-06-01	53	57	Boarding
13	US-AK	2024-03-01	2023-11-27	[null]	31	Boarding
14	PG-NSB	2023-11-26	2023-12-05	230	13	Boarding
15	US-MS	2023-03-28	2023-10-25	298	16	Boarding

Showing rows: 1 to 32 | Page No: 1 of 1 | < << << >> >> > >>

Total rows: 32 Query complete 00:00:00.120

Successfully run. Total query runtime: 120 msec. 32 rows affected.

4°C Clear 21:21 18.11.2025

10. Drop all existing views.

pgAdmin 4

Welcome Sai_db/postgres@PostgreSQL 17*

Query History

```
1 ✓ DROP VIEW IF EXISTS
2   flights_by_date,
3   bookings_next_week,
4   top_5_routes,
5   flights_by_airline,
6   delayed_over_24h,
7   lt_passengers,
8   top_10_visited_countries
9 CASCADE;
```

Data Output Messages Notifications

DROP VIEW

Query returned successfully in 107 msec.

Total rows: 1 Query complete 00:00:00.107

✓ Query returned successfully in 107 msec.

4°C Clear 21:22 18.11.2025