

Kazakh-British Technical University

DATABASES
Laboratory work №8

Kislitsin Alexandr
23B031861

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

The screenshot shows the DBeaver IDE interface with the following details:

- Database Explorer:** Shows the database structure with 'lab_db' selected.
- SQL Editor:** Contains the SQL code for creating a view:


```

CREATE OR REPLACE VIEW Flights_On_Specific_Date AS
SELECT
    flight_id,
    scheduled_departure,
    scheduled_arrival,
    departure_airport_id,
    arrival_airport_id,
    departing_gate,
    arriving_gate,
    airline_id,
    actual_departure,
    actual_arrival,
    status
FROM Flights
WHERE scheduled_departure = '2024-01-15';
      
```
- Services:** Shows the transaction history with entries for connecting to the database and executing the query.
- Notifications:** Displays a timeline of recent notifications.
- Bottom Status:** Shows the completion message '[2025-11-18 22:05:58] completed in 21 ms'.

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

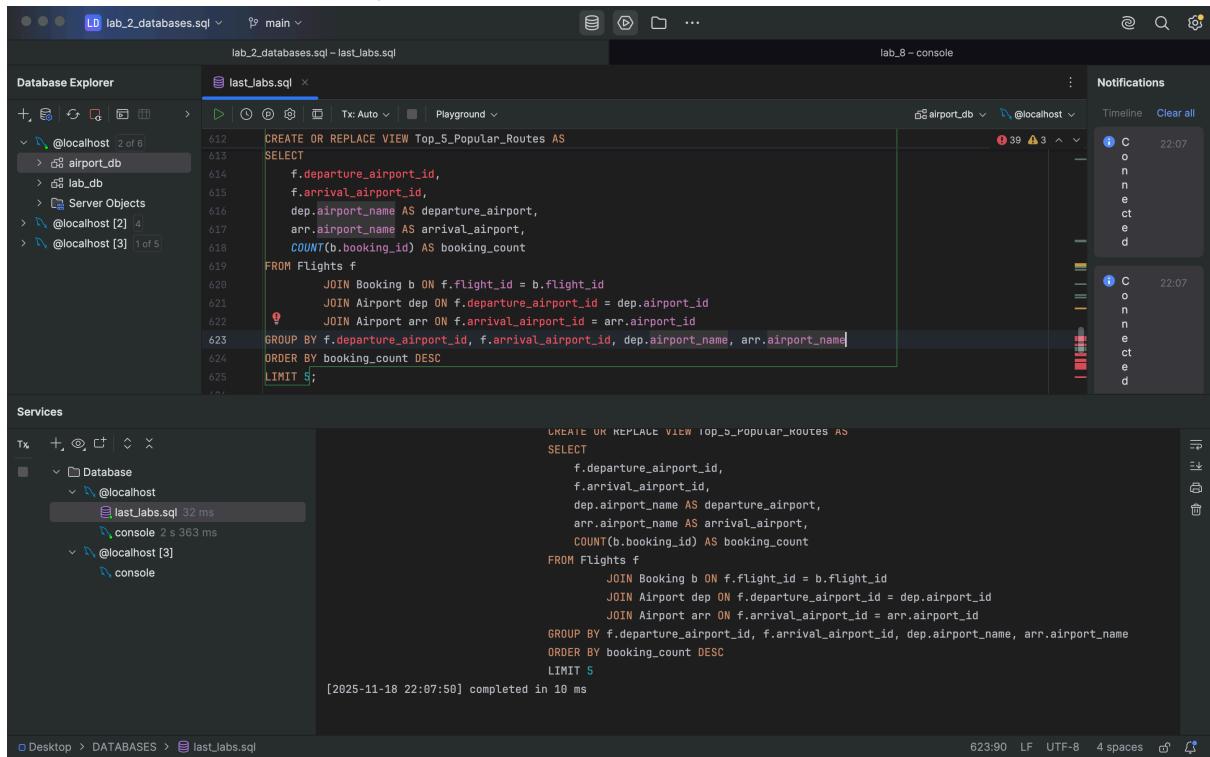
The screenshot shows the DBeaver IDE interface with the following details:

- Database Explorer:** Shows the database structure with 'lab_db' selected.
- SQL Editor:** Contains the SQL code for creating a view:


```

CREATE OR REPLACE VIEW Bookings_Next_Week AS
SELECT
    b.booking_id,
    b.flight_id,
    b.passenger_id,
    b.booking_platform,
    b.status,
    b.ticket_price,
    f.scheduled_departure,
    f.scheduled_arrival
FROM Booking b
JOIN Flights f ON b.flight_id = f.flight_id
WHERE f.scheduled_departure BETWEEN CURDATE() AND DATE_ADD(CURDATE(), INTERVAL 7 DAY);
      
```
- Services:** Shows the transaction history with entries for connecting to the database and executing the query.
- Notifications:** Displays a timeline of recent notifications.
- Bottom Status:** Shows the completion message '[2025-11-18 22:07:22] completed in 10 ms'.

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

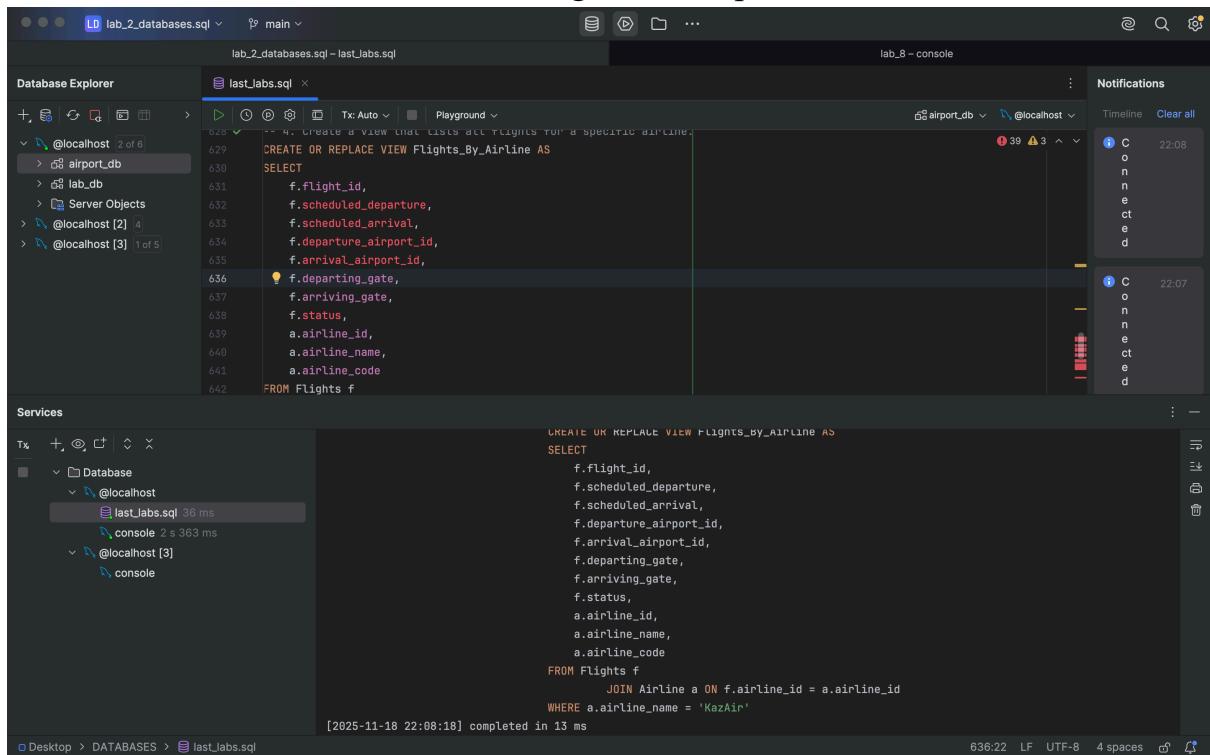


```

CREATE OR REPLACE VIEW Top_5_Popular_Routes AS
SELECT
    f.departure_airport_id,
    f.arrival_airport_id,
    dep.airport_name AS departure_airport,
    arr.airport_name AS arrival_airport,
    COUNT(b.booking_id) AS booking_count
FROM Flights f
    JOIN Booking b ON f.flight_id = b.flight_id
    JOIN Airport dep ON f.departure_airport_id = dep.airport_id
    JOIN Airport arr ON f.arrival_airport_id = arr.airport_id
GROUP BY f.departure_airport_id, f.arrival_airport_id, dep.airport_name, arr.airport_name
ORDER BY booking_count DESC
LIMIT 5;

```

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



```

-- 4. Create a view that lists all flights for a specific airline.
CREATE OR REPLACE VIEW Flights_By_Airline AS
SELECT
    f.flight_id,
    f.scheduled_departure,
    f.scheduled_arrival,
    f.departure_airport_id,
    f.arrival_airport_id,
    f.departing_gate,
    f.arriving_gate,
    f.status,
    a.airline_id,
    a.airline_name,
    a.airline_code
FROM Flights f
    JOIN Airline a ON f.airline_id = a.airline_id
WHERE a.airline_name = 'KazAir';

```

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

```

CREATE OR REPLACE VIEW Flights_By_Airline_Next_7_Days AS
SELECT
    f.flight_id,
    f.scheduled_departure,
    f.scheduled_arrival,
    f.departure_airport_id,
    f.arrival_airport_id,
    f.departing_gate,
    f.arriving_gate,
    f.status,
    a.airline_id,
    a.airline_name,
    a.airline_code
FROM Flights f
    INNER JOIN Airline a ON f.airline_id = a.airline_id
WHERE a.airline_name = 'KazAir'

```

The screenshot shows the MySQL Workbench interface with the 'lab_2_databases.sql' file open. In the Database Explorer, there is a connection to 'localhost' with database 'lab_db'. A new view named 'last_labs.sql' is being created in the 'last_labs.sql' tab. The code above defines a view named 'Flights_By_Airline_Next_7_Days' that selects flight details and joins with the 'Airline' table to filter by 'KazAir'. The execution completed in 13 ms on 2025-11-18 at 22:08:18.

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

```

CREATE OR REPLACE VIEW Delayed_Flights_24h AS
SELECT
    f.flight_id,
    f.scheduled_departure,
    f.actual_departure,
    f.scheduled_arrival,
    f.actual_arrival,
    DATEDIFF(f.actual_arrival, f.scheduled_arrival) AS delay_days,
    TIMESTAMPDIFF(HOUR, f.scheduled_arrival, f.actual_arrival) AS delay_hours,
    a.airline_name
FROM Flights f
    JOIN Airline a ON f.airline_id = a.airline_id
WHERE TIMESTAMPDIFF(HOUR, f.scheduled_arrival, f.actual_arrival) > 24;

```

The screenshot shows the MySQL Workbench interface with the 'lab_2_databases.sql' file open. In the Database Explorer, there is a connection to 'localhost' with database 'lab_db'. A new view named 'last_labs.sql' is being created in the 'last_labs.sql' tab. The code above defines a view named 'Delayed_Flights_24h' that calculates the difference between scheduled and actual arrival times to find flights delayed by more than 24 hours. The execution completed in 9 ms on 2025-11-18 at 22:09:48.

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.

The screenshot shows a SQL development environment with the following details:

- Database Explorer:** Shows the connection to `@localhost [2 of 6]` and the database `airport_db`.
- Console:** The code being run is:


```

CREATE OR REPLACE VIEW Leffler_Thompson_Passengers AS
SELECT DISTINCT
    p.passenger_id,
    CONCAT(p.first_name, ' ', p.last_name) AS full_name,
    p.country_of_citizenship AS country_of_origin,
    p.passport_number,
    b.booking_id,
    b.booking_platform
FROM Passengers p
JOIN Booking b ON p.passenger_id = b.passenger_id
WHERE b.booking_platform = 'Leffler-Thompson';
      
```
- Services:** Shows a transaction history with the last entry being the execution of the view creation query.
- Output:** Displays the results of the `SELECT * FROM Leffler_Thompson_Passengers;` query, showing one row with passenger_id 157, full_name "Philbert Shambroke", country_of_origin "Colombia", passport_number "251255686-7", booking_id 15, and booking_platform "Leffler-Thompson".

The screenshot shows a SQL development environment with the following details:

- Database Explorer:** Shows the connection to `@localhost [2 of 6]` and the database `airport_db`.
- Console:** The code being run is:


```

CREATE OR REPLACE VIEW Leffler_Thompson_Passengers AS
SELECT DISTINCT
    p.passenger_id,
    p.first_name,
    p.last_name,
    p.country_of_citizenship AS country_of_origin,
    p.passport_number,
    b.booking_id,
    b.booking_platform
FROM Passengers p
JOIN Booking b ON p.passenger_id = b.passenger_id
WHERE b.booking_platform = 'Leffler-Thompson';
      
```
- Services:** Shows a transaction history with the last entry being the execution of the view creation query.
- Output:** Displays the results of the `SELECT * FROM Leffler_Thompson_Passengers;` query, showing one row with passenger_id 157, full_name "Philbert Shambroke", country_of_origin "Colombia", passport_number "251255686-7", booking_id 15, and booking_platform "Leffler-Thompson".

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

The screenshot shows a SQL development environment with the following interface elements:

- Database Explorer**: Shows connections to @localhost (2 of 6), including airport_db, lab_db, and Server Objects.
- last_labs.sql**: The current file being edited, containing the SQL code for creating a view.
- Notifications**: A sidebar showing three notifications: "Connected" at 22:11, "Connected" at 22:10, and "Connected" at 22:09.
- Services**: Shows a transaction (Tx) and a database connection to @localhost.
- Output**: Shows the execution results of the SQL query, including the creation of the view and its definition.
- Result 46**: Shows the results of the query, which is empty as it has just been created.
- Bottom Status Bar**: Displays the date (2025-11-18), time (22:11:41), file type (LF), encoding (UTF-8), and code style settings (4 spaces).

```
-- 8. Create a view that shows top 10 most visited countries.
CREATE OR REPLACE VIEW Top_10_Most_Visited_Countries AS
SELECT
    ap.country,
    COUNT(b.booking_id) AS visit_count
FROM Airport ap
    JOIN Flights f ON ap.airport_id = f.arrival_airport_id
    JOIN Booking b ON f.Flight_id = b.flight_id
GROUP BY ap.country
ORDER BY visit_count DESC
LIMIT 10;
```

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

The screenshot shows a SQL development environment with the following interface elements:

- Database Explorer**: Shows connections to @localhost (2 of 6), including airport_db, lab_db, and Server Objects.
- last_labs.sql**: The current file being edited, containing the SQL code for updating a view.
- Notifications**: A sidebar showing three notifications: "Connected" at 22:12, "Connected" at 22:11, and "Connected" at 22:10.
- Services**: Shows a transaction (Tx) and a database connection to @localhost.
- Output**: Shows the execution results of the SQL query, including the update of the view and its definition.
- Result 46**: Shows the results of the query, which is empty as it has just been updated.
- Bottom Status Bar**: Displays the date (2025-11-18), time (22:12:08), file type (LF), encoding (UTF-8), and code style settings (4 spaces).

```
CREATE OR REPLACE VIEW Leffler_Thompson_Passengers AS
SELECT DISTINCT
    p.passenger_id,
    CONCAT(p.first_name, ' ', p.last_name) AS full_name,
    p.country_of_citizenship AS country_of_origin,
    p.passport_number,
    b.booking_id,
    b.booking_platform,
    b.ticket_price,
    b.status AS booking_status,
    b.created_at AS booking_date
FROM Passengers p
    JOIN Booking b ON p.passenger_id = b.passenger_id
WHERE b.booking_platform = 'Leffler-Thompson';
```

10. Drop all existing views.

The screenshot shows a SQL development environment with the following interface elements:

- Database Explorer:** Shows connections to three hosts: @localhost [2 of 6], @localhost [2] [4], and @localhost [3] [1 of 5].
- last_labs.sql:** The current file being edited, containing a series of SQL commands to drop various views.
- Notifications:** A sidebar showing two notifications: "Connected" at 22:12 and "Connected" at 22:12.
- Services:** A sidebar showing database connections and their status.
- Output:** A tab showing the results of the executed SQL commands.
- Result 46:** The output of the last command, which failed due to an unknown table.

```
735 DROP VIEW IF EXISTS Flights_On_Specific_Date;
736 DROP VIEW IF EXISTS Bookings_Next_Week;
737 DROP VIEW IF EXISTS Top_5_Popular_Routes;
738 DROP VIEW IF EXISTS Flights_By_Airline;
739 DROP VIEW IF EXISTS Flights_By_Airline_Next_7_Days;
740 DROP VIEW IF EXISTS Delayed_Flights_24h;
741 DROP VIEW IF EXISTS Leffler_Thompson_Passengers;
742 DROP VIEW IF EXISTS Top_10_Most_Visited_Countries;
743 ✓ DROP VIEW IF EXISTS Flights_On_Date;

[2025-11-18 22:12:08] completed in 8 ms
[2025-11-18 22:12:30] airport_db> DROP VIEW IF EXISTS Flights_On_Date
[2025-11-18 22:12:30] [42S02][1051] Unknown table 'airport_db.flights_on_date'
[2025-11-18 22:12:30] completed in 18 ms
```