

The screenshot shows a PostgreSQL database management interface. In the Database Explorer, the `postgres` database is selected, and the `public` schema is expanded to show tables like `airline`, `airport`, `baggage`, `baggage_check`, `boarding_pass`, `booking`, `booking_flight`, and `flights`. In the console tab, a command is being run to create an index:

```
CREATE INDEX idx_passenger_search
ON Passengers (first_name, last_name, date_of_birth, country_of_citizenship);
```

The command fails because it tries to create an index that already exists. The error message is:

```
[2025-11-12 00:07:47] [42P07] ERROR: relation "idx_unique_passport" already exists
```

Below the console, the Services panel shows the transaction history. It lists several failed attempts to create the same index, each followed by the error message and the completion time:

- [2025-11-12 00:07:47] [42P07] ERROR: relation "idx\_unique\_passport" already exists
- [2025-11-12 00:07:47] postgres.public> CREATE UNIQUE INDEX idx\_unique\_passport  
ON Passengers (passport\_number)
- [2025-11-12 00:07:47] [42P07] ERROR: relation "idx\_unique\_passport" already exists
- [2025-11-12 00:07:47] postgres.public> CREATE UNIQUE INDEX idx\_unique\_passport  
ON Passengers (passport\_number)
- [2025-11-12 00:07:47] [42P07] ERROR: relation "idx\_unique\_passport" already exists
- [2025-11-12 00:08:25] postgres.public> CREATE INDEX idx\_passenger\_search  
ON Passengers (first\_name, last\_name, date\_of\_birth, country\_of\_citizenship)
- [2025-11-12 00:08:25] completed in 6 ms.

The bottom status bar indicates the session is at row 3.1, using CRLF line endings, UTF-8 encoding, and 4 spaces. The system tray shows the date and time as 11/12/2025, 12:08 AM.

The screenshot shows a PostgreSQL database management interface with the following details:

- Database Explorer:** Shows the `postgres` database with 4 objects, including 3 tables: `airline`, `airport`, `baggage`, `baggage_check`, `boarding_pass`, `booking`, `booking_flight`, and `flights`.
- Console:** The current tab is `console`. A command is being run:

```
CREATE UNIQUE INDEX idx_unique_passport  
ON Passengers (passport_number);
```
- Result:** An error message is displayed: `[42P07] ERROR: relation "idx_unique_passport" already exists`.
- Services:** Shows a transaction (`Tx`) and an output window with the following logs:

```
[2025-11-12 00:05:43] [22P02] ERROR: invalid input syntax for type integer: "DME"  
[2025-11-12 00:06:43] Позиция: 90  
[2025-11-12 00:06:55] postgres.public> EXPLAIN ANALYZE  
SELECT flight_id, flight_no  
FROM flights  
WHERE departure_airport_id = 1  
AND arrival_airport_id = 3  
[2025-11-12 00:06:55] 7 rows retrieved starting from 1 in 394 ms (execution: 13 ms, fetching: 381 ms)  
[2025-11-12 00:07:43] postgres.public> CREATE UNIQUE INDEX idx_unique_passport  
ON Passengers (passport_number)  
[2025-11-12 00:07:43] completed in 7 ms  
[2025-11-12 00:07:45] postgres.public> CREATE UNIQUE INDEX idx_unique_passport  
ON Passengers (passport_number)
```
- Bottom Status:** Shows the console path as `Database Consoles > postgres > console`, and the terminal settings as `1:1 CRLF UTF-8 4 spaces`.

The screenshot shows a PostgreSQL database console interface. In the top-left, the Database Explorer sidebar lists the 'postgres' database with its tables: airline, airport, baggage, baggage\_check, boarding\_pass, booking, booking\_flight, and flights. The 'flights' table is currently selected. The main area displays the following SQL query:

```
EXPLAIN ANALYZE
SELECT flight_id, flight_no
FROM flights
WHERE departure_airport_id = 1
    AND arrival_airport_id = 3;
```

Below the query, the 'Output' tab shows the detailed EXPLAIN ANALYZE results:

```
1 Bitmap Heap Scan on flights  (cost=4.30..10.33 rows=2 width=10) (actual time=0.025..0.030 rows=4 loops=1)
  2  Recheck Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 3))
  3  Heap Blocks: exact=3
  4 -> Bitmap Index Scan on idx_flights_route  (cost=0.00..4.29 rows=2 width=0) (actual time=0.018..0.018 rows=4 loops=1)
  5   Index Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 3))
  6 Planning Time: 0.098 ms
  7 Execution Time: 0.047 ms
```

The bottom status bar indicates the session details: Database Consoles > postgres > console, along with the timestamp 4:31, CRLF, UTF-8, and 4 spaces.

The screenshot shows a PostgreSQL database management interface with the following details:

- Database Explorer:** Shows the `postgres` database with 4 total tables: `airline`, `airport`, `baggage`, `baggage_check`, `boarding_pass`, `booking`, `booking_flight`, and `flights`.
- Console:** A query window titled "console" containing the following SQL code:

```
EXPLAIN ANALYZE
SELECT * FROM flights
WHERE departure_airport_id = 1 AND arrival_airport_id = 3;
```
- Services:** A results window showing the query execution plan:

```
1 Bitmap Heap Scan on flights  (cost=4.30..10.33 rows=2 width=189) (actual time=0.164..0.168 rows=4 loops=1)
  2  Recheck Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 3))
  3  Heap Blocks: exact=3
  4 -> Bitmap Index Scan on idx_flights_route  (cost=0.00..4.29 rows=2 width=0) (actual time=0.160..0.160 rows=4 loops=1)
      Index Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 3))
  5  Planning Time: 0.332 ms
  6  Execution Time: 0.379 ms
```
- System:** A taskbar at the bottom with various icons and status information.

The screenshot shows a PostgreSQL database management interface with the following details:

- Database Explorer:** Shows the schema structure of the `postgres` database, specifically the `public` schema which contains tables like `airline`, `airport`, `baggage`, `baggage_check`, `boarding_pass`, `booking`, `booking_flight`, and `flights`.
- Console:** A tab labeled `console` is active, displaying SQL commands and their execution results.
- SQL Query:** The command shown is:

```
CREATE INDEX idx_flights_route
ON flights (departure_airport_id, arrival_airport_id);
```
- Execution Log:** The output window shows the execution log for the query:

```
FROM flights
GROUP BY flight_no, scheduled_departure
HAVING COUNT(*) > 1
)
[2025-11-12 00:01:36] 14 rows retrieved starting from 1 in 462 ms (execution: 21 ms, fetching: 441 ms)
[2025-11-12 00:02:27] postgres.public> CREATE INDEX idx_flight_sched
ON flights (flight_no, scheduled_departure)
[2025-11-12 00:02:27] completed in 7 ms
[2025-11-12 00:03:01] postgres.public> CREATE INDEX idx_flights_route
ON flights (departure_airport_id, arrival_airport_id)
[2025-11-12 00:03:01] completed in 7 ms.
```
- Services:** A sidebar showing the status of the database service.
- Bottom Status Bar:** Displays the current session path, encoding (3:1 CRLF), character set (UTF-8), and code style (4 spaces).

The screenshot shows the PostgreSQL Database Explorer in the VS Code interface. The left sidebar displays the database schema, including the `flights` table and its columns: `flight_no`, `scheduled_departure`, and `actual_departure`. The right pane shows the `console` tab where the command `CREATE INDEX idx_flight_sched ON flights (flight_no, scheduled_departure);` has been run and successfully executed.

Database Explorer

postgres | 1 of 4

postgres | 1 of 3

public

tables | 10

- airline
- airport
- baggage
- baggage\_check
- boarding\_pass
- booking
- booking\_flight
- flights

Services

Tx

Output

postgres.public.flights

Database

postgres

console

```
FROM flights
WHERE (flight_no, scheduled_departure) IN (
    SELECT flight_no, scheduled_departure
    FROM flights
    GROUP BY flight_no, scheduled_departure
    HAVING COUNT(*) > 1
)
[2025-11-12 00:01:36] 14 rows retrieved starting from 1 in 462 ms (execution: 21 ms, fetching: 441 ms)
[2025-11-12 00:02:27] postgres> CREATE INDEX idx_flight_sched
ON flights (flight_no, scheduled_departure)
[2025-11-12 00:02:27] completed in 7 ms
```

Database Consoles > postgres > console

3:1 CRLF UTF-8 4 spaces

The screenshot shows the DataGrip IDE interface. At the top, there's a toolbar with various icons. Below it is a header bar with tabs for 'llab-2' and 'Version control'. The main area is divided into several panes:

- Database Explorer** (left pane): Shows a tree view of the 'postgres' database schema. Under 'public' schema, there are 10 tables: airline, airport, baggage, baggage\_check, boarding\_pass, booking, booking\_flight, and flights.
- console** (center pane): A code editor window showing the execution of a SQL command to create an index:

```
CREATE INDEX idx_flights_actual_departure  
ON flights (actual_departure);
```
- Services** (bottom-left pane): Displays a transaction log with the following entries:
  - [2025-11-11 23:56:12] Connected
  - [2025-11-11 23:56:12] postgres> CREATE INDEX idx\_flights\_actual\_departure  
ON flights (actual\_departure)
  - [2025-11-11 23:56:12] completed in 47 ms
- Bottom Status Bar**: Shows the path 'Database Consoles > postgres > console', file encoding '3.1 CRLF', character set 'UTF-8', and other system information like 'USD/INR -0.29%', battery level, and network status.