

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

- Top Bar:** Includes tabs for "Lab 2 DB" and "Version control", and icons for database management.
- Database Explorer:** Shows the schema of the "postgres" database, including tables like airport, baggage, baggage_check, boarding_pass, booking, booking_flight, flights, passengers, and security_check, along with their columns, keys, and indexes.
- Console:** A "console_5" tab is active, displaying the command: `create index idx_flights_act_departure on flights (act_departure_time);`. The "indexes" section of the schema browser also lists this index.
- Services:** A "Tx" panel shows the transaction history with the following log:
 - [2025-11-10 18:34:43] postgres.lab2> create index idx_flights_act_departure on flights (act_departure_time)
 - [2025-11-10 18:34:43] completed in 3 ms
- Bottom Status:** Shows the current session details: "Database Consoles > postgres@localhost > console_5", and system status: "1:72 LF UTF-8 4 spaces".

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

Database Explorer pane (left):

- Connected to **postgres@localhost**.
- Selected schema: **flights**.
- Shows tables: **airport**, **baggage**, **baggage_check**, **boarding_pass**, **booking**, **booking_flight**, **columns**, **keys**, **foreign keys**, **indexes**, and **passengers**.

Console_5 pane (top right):

- Query: `create unique index uq_flights_id_sch on flights(flight_id, sch_departure_time);`
- Status: 1 ✓ (Success)
- Execution details: `iu flights_pkey (flight_id) UNIQUE`, `iu uq_flights_id_sch (flight_id, sch_departure_time) UNIQUE`, `i idx_flights_act_departure (act_departure_time)`.

Services pane (bottom left):

- Shows a transaction history with the following logs:
 - [2025-11-10 18:34:43] postgres.lab2> create index idx_flights_act_departure on flights (act_departure_time)
 - [2025-11-10 18:34:43] completed in 3 ms
 - [2025-11-10 18:40:18] postgres.lab2> create unique index uq_flights_id_sch on flights(flight_id, sch_departure_time)
 - [2025-11-10 18:40:18] completed in 3 ms
- Database tree:
 - Database
 - postgres@localhost
 - flights (selected)
 - flights
 - flights
 - flights
 - flights

Bottom Status Bar:

- Database Consoles > postgres@localhost > console_5
- 1:1 LF UTF

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

Top Bar: LD Lab 2 DB, Version control, Database Explorer, Database Explorer (with tabs like +, DDL, etc.), console_5, flights, Tx: Auto, Playground, postgres.lab2, and a gear icon.

Database Explorer: Shows the schema of the 'flights' database. Under 'flights', the 'indexes' folder is expanded, showing four indexes: flights_pkey, uq_flights_id_sch, idx_flights_act_departure, and idx_flights_dep_arriv.

Console: The current tab is 'console_5' under the 'flights' database. A query is being run:

```
create index| idx_flights_dep_arriv ON flights (departing_airport_id, arriving_airport_id);
```

The status bar indicates the query is in progress (1 ✓).

Services: Shows a transaction history with the following log entries:

- [2025-11-10 18:34:43] postgres.lab2> create index idx_flights_act_departure on flights (act_departure_time)
- [2025-11-10 18:34:43] completed in 3 ms
- [2025-11-10 18:40:18] postgres.lab2> create unique index uq_flights_id_sch on flights(flight_id, sch_departure_time)
- [2025-11-10 18:40:18] completed in 3 ms
- [2025-11-10 18:58:14] postgres.lab2> create index idx_flights_dep_arriv ON flights (departing_airport_id, arriving_airport_id)
- [2025-11-10 18:58:14] completed in 7 ms

Bottom Status Bar: Database Consoles > postgres@localhost > console_5, 1:13, LF, UTF-8, 4 spaces, and a file icon.

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

Database Explorer (Left Panel):

- Connected to **postgres@localhost**.
- Selected schema: **flights**.
- Tables listed: boarding_pass, booking, booking_flight, flights, passengers, security_check, public.
- Indexes listed: flights_pkey, uq_flights_id_sch, idx_flights_act_departure.

Console 5 (Top Right):

- Query: `--without index | explain analyse select flight_id from flights where departing_airport_id = 1 and arriving_airport_id = 2;`
- Status: 6 rows.

Services (Bottom Left):

- Transactions (Tx):
- Database: **postgres@localhost** (selected).
- Flights (342 ms):
 - flights (342 ms)
 - flights (342 ms)
 - flights (342 ms)
 - flights (342 ms)

Output (Bottom Right):

- Result: 299 rows.
- Query Plan:
 - Seq Scan on flights (cost=0.00..6.01 rows=1 width=4) (actual time=0.077..0.078 rows=1 loops=1)
 - Filter: ((departing_airport_id = 1) AND (arriving_airport_id = 2))
 - Rows Removed by Filter: 200
 - Planning Time: 0.135 ms
 - Execution Time: 0.100 ms

Bottom Status Bar:

- Database Consoles > postgres@localhost > console_5
- 1:17 LF UTF-8 4 spaces

The screenshot shows a PostgreSQL database management interface with two main panes. The left pane is the Database Explorer, displaying the schema of the 'flights' table. The right pane is the Query Editor, showing a query and its execution plan.

Database Explorer:

- Selected database: postgres@localhost
- Selected schema: flights
- Tables listed: boarding_pass, booking, booking_flight, columns (12), keys (1), foreign keys (3), indexes (4).

Query Editor:

```
--with index
create index idx_flights_dep_arriv on flights (departing_airport_id, arriving_airport_id);
set enable_seqscan = off;
explain analyse
select flight_id
from flights
where departing_airport_id = 1
    and arriving_airport_id = 2;
set enable_seqscan = on;
```

Services:

- Selected transaction: console_5 (12 ms)
- Selected table: flights (342 ms)

Output:

Step	Details
1	Index Scan using idx_flights_dep_arriv on flights (cost=0.14..8.16 rows=1 width=4) (actual time=0.021..0.022 rows=1 loops=1)
2	Index Cond: ((departing_airport_id = 1) AND (arriving_airport_id = 2))
3	Planning Time: 0.361 ms
4	Execution Time: 0.033 ms

Bottom status bar: Database Consoles > postgres@localhost > console_5 | 4 rows | ... | 9:25 LF UTF-8 4 spaces

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

Top Bar: LD Lab 2 DB, Version control, Database Explorer, Database Explorer (flights), Tx: Auto, Playground, postgres.lab2, and a green checkmark icon.

Database Explorer: Shows the database structure for `postgres@localhost`. The `flights` schema is selected. Other schemas listed are `boarding_pass`, `booking`, `booking_flight`, `passengers`, `security_check`, and `public`.

Console: The current tab is `console_5`. The query being run is:

```
1 ✓ create unique index uq_passengers_passport on passengers(passport_number);
2
3 ✓ select indexname, indexdef
4   from pg_indexes
5   where tablename = 'passengers';
6
```

Services: Shows the transaction history. The most recent transaction is in the `flights` schema and took 342 ms.

Output: Shows the results of the query run in the console. The table `postgres.pg_catalog.pg_indexes` contains two rows:

indexname	indexdef
passengers_pkey	CREATE UNIQUE INDEX passengers_pkey ON lab2.passengers USING btree (passenger_id)
uq_passengers_passport	CREATE UNIQUE INDEX uq_passengers_passport ON lab2.passengers USING btree (passport_number)

Bottom Status: Database Consoles > postgres@localhost > console_5, 6:1 LF UTF-8 4 spaces, and a status bar icon.

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

Database Explorer pane (left):

- Selected database: `postgres@localhost`
- Tables listed under `flights` and `passengers`.
- Selected table: `passengers`
- Columns, keys, indexes, security checks, and public objects are also listed.

Console pane (right):

- Session: `console_5`
- Current tab: `passengers`
- Playground tab is active.
- Code editor content:

```
1 create unique index uq_passengers_passport on passengers(passport_number);
2
3 ✓ INSERT INTO passengers (passenger_id, first_name, last_name, date_of_birth, gender,
4                               country_of_citizenship, country_of_residence, passport_number,
5                               created_at, updated_at)
6 VALUES (passenger_id 203, first_name 'Илюха', last_name 'Монеси', date_of_birth '2010-05-15', gender 'Male',
7         country_of_citizenship 'Russia', country_of_residence 'Russia', passport_number 'UP203', created_at now(), updated_at now());
8
9 ! INSERT INTO passengers (passenger_id, first_name, last_name, date_of_birth, gender,
10                            country_of_citizenship, country_of_residence, passport_number,
11                            created_at, updated_at)
12 VALUES (passenger_id 204, first_name 'Габен', last_name 'Отцов', date_of_birth '2012-08-25', gender 'Male',
13         country_of_citizenship 'Russia', country_of_residence 'Russia', passport_number 'UP203', created_at now(), updated_at now());
14
15 --вставка падает с ошибкой duplicate key value violates unique constraint ... , потому что уникальный индекс
16 --гарантирует, что passport_number не повторяется в таблице. Это и есть проверка – индекс работает.
```

Services pane (bottom):

- Transactions (Tx) list.
- Selected transaction: `console_5` (37 ms).
- Logs:

 - [2025-11-11 10:19:45] `postgres.lab2> INSERT INTO passengers (passenger_id, first_name, last_name, date_of_birth, gender, country_of_citizenship, country_of_residence, passport_number, created_at, updated_at)`
 - [2025-11-11 10:19:45] `VALUES (203, 'Илюха', 'Монеси', '2010-05-15', 'Male', 'Russia', 'Russia', 'UP203', now(), now())`
 - [2025-11-11 10:19:45] `1 row affected in 6 ms`
 - [2025-11-11 10:19:45] `postgres.lab2> INSERT INTO passengers (passenger_id, first_name, last_name, date_of_birth, gender, country_of_citizenship, country_of_residence, passport_number, created_at, updated_at)`
 - [2025-11-11 10:19:45] `VALUES (204, 'Габен', 'Отцов', '2012-08-25', 'Male', 'Russia', 'Russia', 'UP203', now(), now())`
 - [2025-11-11 10:19:45] `[23505] ERROR: duplicate key value violates unique constraint "uq_passengers_passport"`
 - [2025-11-11 10:19:45] `Подробности: Key (passport_number)=(UP203) already exists.`

Lab 2 DB Version control

Database Explorer

postgres@localhost

- > passengers
- > flights
- > boarding_pass
- > baggage
- > baggage_check
- > airport
- > booking
- > booking_flight
- > console_5

passengers

- > columns 10
- > keys 1
- > indexes 3
 - i_u passengers_pkey (passenger_id)
 - i_u uq_passengers_passport (passport)
 - i_idx_passengers_country_dob (country_of_citizenship, date_of_birth)

Security check

console_5

passengers

flights

Playground

```
1 ✓ create index idx_passengers_country_dob_name on passengers (country_of_citizenship, date_of_birth, last_name, first_name);
2 --changed the condition values, because i dont have that country and dob in passenger table
3 ✓ explain analyse
4 select passenger_id, first_name, last_name, date_of_birth
5 from passengers
6 where country_of_citizenship = 'USA'
7 and extract(year from date_of_birth) = 1990;
8
9 --в таблице очень мало строк, планировщик может выбрать Seq Scan, так как он считает быстрее на маленьких таблицах
```

Services

Tx

Output changed the condition in passenger table

Database

postgres@localhost

- passengers 440 ms
- console_5 385 ms
- flights 342 ms
- flights
- flights
- flights
- flights 342 ms

QUERY PLAN

```
1 Seq Scan on passengers  (cost=0.00..6.61 rows=1 width=27) (actual time=0.213..0.214 rows=1 loops=1)
2   Filter: ((country_of_citizenship)::text = 'USA'::text) AND (EXTRACT(year FROM date_of_birth) = '1990'::numeric)
3   Rows Removed by Filter: 205
4 Planning Time: 2.422 ms
5 Execution Time: 0.291 ms
```

5 rows

Database Consoles > postgres@localhost > console_5

9:15 LF UTF-8 4 spaces

Database Explorer

postgres@localhost

- > **passengers**
- > columns 10
- > keys 1
- > **indexes 3**
 - i_u passengers_pkey (passenger_
 - i_u uq_passengers_passport (pa
 - i_idx_passengers_country_dob

Services

Output postgres.pg_catalog.pg_indexes

indexname
1 passengers_pkey
2 uq_passengers_passport
3 idx_passengers_country_dob_name

indexname
1 passengers_pkey
2 uq_passengers_passport
3 idx_passengers_country_dob_name

Lab 2 DB Version control

Database Explorer

postgres@localhost

- tables 10
 - airline
 - airport
 - baggage
 - baggage_check
 - boarding_pass
 - booking
 - booking_flight
 - flights
 - columns 12
 - keys 1
 - foreign keys 3
 - indexes 1
 - passengers
 - security_check

console_5

```
1 ✓ drop index uq_passengers_passport;
2 ✓ drop index idx_passengers_country_dob_name;
3 ✓ drop index idx_flights_dep_arriv;
4 ✓ drop index idx_flights_act_departure;
5 ✓ drop index uq_flights_id_sch;
```

postres.lab2

Services

Tx + ⌂ ^ ×

Database

postgres@localhost

console_5 13 ms

flights 411 ms

flights

flights

flights

```
[2025-11-11 22:29:39] 5 rows retrieved starting from 1 in 366 ms (execution: 28 ms, fetching: 338 ms)
[2025-11-11 22:40:53] postgres.lab2> select indexname
   from pg_indexes
   where tablename = 'passengers'
[2025-11-11 22:40:53] 3 rows retrieved starting from 1 in 359 ms (execution: 8 ms, fetching: 351 ms)
[2025-11-11 22:46:07] postgres.lab2> drop index uq_passengers_passport
[2025-11-11 22:46:07] completed in 10 ms
[2025-11-11 22:46:07] postgres.lab2> drop index idx_passengers_country_dob_name
[2025-11-11 22:46:07] completed in 3 ms
[2025-11-11 22:46:07] postgres.lab2> drop index idx_flights_dep_arriv
[2025-11-11 22:46:07] completed in 3 ms
[2025-11-11 22:46:07] postgres.lab2> drop index idx_flights_act_departure
[2025-11-11 22:46:07] completed in 2 ms
[2025-11-11 22:46:07] postgres.lab2> drop index uq_flights_id_sch
[2025-11-11 22:46:07] completed in 2 ms
```

Database Consoles > postgres@localhost > console_5

5:30 LF UTF