

Lab 07

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su... X

Query History

```
1 SELECT * FROM flights;
2
3
4
5 CREATE INDEX task1 ON flights(actual_departure);
6
7
8 CREATE UNIQUE INDEX task2 ON flights(flight_no, scheduled_departure);
9
10 CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
11
12
13
14
15 CREATE INDEX task4 ON flights(arriving_gate);
16 SELECT * FROM flights WHERE arriving_gate = '649';
17
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 58 msec.

Total rows: Query complete 00:00:00.058

CRLF Ln 6, Col 1

12:10 09.11.2025

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su... X

Query History

```
8 -- CREATE UNIQUE INDEX, выходит ошибка из за связей
9 CREATE INDEX task2 ON flights(flight_no, scheduled_departure);
10
11
12 CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
13
14
15
16 CREATE INDEX task4 ON flights(arriving_gate);
17 SELECT * FROM flights WHERE arriving_gate = '649';
18
19
20
21
22
23
24
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 61 msec.

Total rows: Query complete 00:00:00.061

CRLF Ln 9, Col 1

13:52 09.11.2025

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Edit View Window Help**: The top menu bar.
- Object Explorer Servers**: The left sidebar.
- Processes suppliers/postgres@PostgreSQL 17***: The top tab bar.
- public.booking/su...**: The second tab in the tab bar.
- Query History**: The tab selected in the toolbar.
- Scratch Pad**: A tab in the top right corner.
- Code Area:**

```
1  SELECT * FROM flights;
2
3
4
5  CREATE INDEX task1 ON flights(actual_departure);
6
7
8
9
10 CREATE UNIQUE INDEX task2 ON flights(flight_no, scheduled_departure);
11
12
13 CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
14
15
16
17 CREATE INDEX task4 ON flights(arriving_gate);
```
- Data Output**: The tab selected in the bottom navigation bar.
- Messages Notifications**: Other tabs in the bottom navigation bar.
- CREATE INDEX**: The message displayed in the Data Output area.
- Query returned successfully in 50 msec.**: The status message at the bottom.
- Total rows: Query complete 00:00:00.050**: The bottom status bar.
- CRLF Ln 13, Col 1**: The bottom right corner status.

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su... X

suppliers/postgres@PostgreSQL 17 No limit

Query History

```
1 SELECT * FROM flights;
2
3
4
5 CREATE INDEX task1 ON flights(actual_departure);
6
7
8 CREATE UNIQUE INDEX task2 ON flights(flight_no, scheduled_departure);
9
10
11 CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
12
13
14 SELECT * FROM flights WHERE arriving_gate = '649';
15
16
17
```

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1 of 1

	flight_id [PK] integer	flight_no character varying (50)	scheduled_departure_date	scheduled_arrival_date	departure_airport_id integer	arrival_airport_id integer	departing_gate character varying (50)	arriving_gate character varying (50)	airline_id integer
1	1	US-CT	2024-01-22	2023-09-08	12	15	9	649	31

Total rows: 1 Query complete 00:00:00.149

Successfully run. Total query runtime: 149 msec. 1 rows affected. X

CRLF Ln 15, Col 1

12:07 09.11.2025

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su...

suppliers/postgres@PostgreSQL 17

No limit

CREATE INDEX task1 ON flights(actual_departure);
CREATE UNIQUE INDEX task2 ON flights(flight_no, scheduled_departure);
CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
CREATE INDEX task4 ON flights(arriving_gate);
SELECT * FROM flights WHERE arriving_gate = '649';

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1 of 1

flight_id	flight_no	scheduled_departure	scheduled_arrival	departure_airport_id	arrival_airport_id	departing_gate	arriving_gate	airline_id
1	US-CT	2024-01-22	2023-09-08	12	15	9	649	35

Total rows: 1 Query complete 00:00:00.061

Successfully run. Total query runtime: 61 msec. 1 rows affected.

CRLF Ln 16, Col 1

12:08 09.11.2025

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su...

suppliers/postgres@PostgreSQL 17

No limit

CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
CREATE INDEX task4 ON flights(arriving_gate);
SELECT * FROM flights WHERE arriving_gate = '649';
DROP INDEX task3;
EXPLAIN ANALYZE
SELECT * FROM flights
WHERE departure_airport_id = 12 AND arrival_airport_id = 15;
SELECT * FROM flights;

Data Output Messages Notifications

Showing rows: 1 to 5 Page No: 1 of 1

flight_id	flight_no	scheduled_departure	scheduled_arrival	departure_airport_id	arrival_airport_id	departing_gate	arriving_gate	airline_id
1	US-CT	2024-01-22	2023-09-08	12	15	9	649	35
2	US-CA	2024-01-22	2023-09-08	12	15	9	649	35
3	US-DE	2024-01-22	2023-09-08	12	15	9	649	35
4	US-FR	2024-01-22	2023-09-08	12	15	9	649	35

QUERY PLAN

text:
1 Seq Scan on flights (cost=0.00..27.95 rows=3 width=61) (actual time=0.009..0.118 rows=2 loops=1)
2 Filter: ((departure_airport_id = 12) AND (arrival_airport_id = 15))
3 Rows Removed by Filter: 995
4 Planning Time: 0.777 ms
5 Execution Time: 0.131 ms

Total rows: 5 Query complete 00:00:00.052

Successfully run. Total query runtime: 52 msec. 5 rows affected.

CRLF Ln 21, Col 1

13:56 09.11.2025

Without index:

Query time = 61 ms

With index:

Query time = 52 ms

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su...

Query History

```

9 CREATE INDEX task2 ON flights(flight_no, scheduled_departure);
10
11
12 CREATE INDEX task3 ON flights(departure_airport_id, arrival_airport_id);
13
14
15
16 CREATE INDEX task4 ON flights(arriving_gate);
17 SELECT * FROM flights WHERE arriving_gate = '649';
18
19
20
21 EXPLAIN ANALYZE
22 SELECT * FROM flights
23 WHERE departure_airport_id = 12 AND arrival_airport_id = 15;
24
25

```

Data Output Messages Notifications

QUERY PLAN text

1	Bitmap Heap Scan on flights (cost=4.31..12.03 rows=3 width=61) (actual time=0.383..0.386 rows=2 loops=)
2	Recheck Cond: ((departure_airport_id = 12) AND (arrival_airport_id = 15))
3	Heap Blocks: exact=1
4	-> Bitmap Index Scan on task3 (cost=0.00..4.31 rows=3 width=0) (actual time=0.071..0.071 rows=2 loops=)
5	Index Cond: ((departure_airport_id = 12) AND (arrival_airport_id = 15))
6	Planning Time: 0.152 ms
7	Execution Time: 0.604 ms

Showing rows: 1 to 7 Page No: 1 of 1

Total rows: 7 Query complete 00:00:00.070

Successfully run. Total query runtime: 70 msec. 7 rows affected.

CRLF Ln 21, Col 1

13:55 09.11.2025

Index Scan using task3 — means PostgreSQL uses index Index Cond: — shows the conditions under which the index operates Execution Time: — total execution time

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer Servers

Processes suppliers/postgres@PostgreSQL 17* public.booking/su...

Query History

```

1 CREATE UNIQUE INDEX task6 ON passengers(passport_number);
2
3
4
5
6

```

Data Output Messages Notifications

indexname	indexdef
1 passengers_pkey	CREATE UNIQUE INDEX passengers_pkey ON public.passengers USING btree (passenger_id)
2 task6	CREATE UNIQUE INDEX task6 ON public.passengers USING btree (passport_number)

Showing rows: 1 to 2 Page No: 1 of 1

Total rows: 2 Query complete 00:00:00.067

CRLF Ln 6, Col 1

16:04 09.11.2025

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer tree, which includes sections for Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (10). Under Tables, 'passenger' is expanded, showing Constraints (4) which include 'age', 'date_of_birth_chkd', 'passenger_pkey', and 'task1'. The main window contains a query editor with the following SQL code:

```
1 CREATE UNIQUE INDEX task6 ON passengers(passport_number);
2
3 SELECT indexname, indexdef
4 FROM pg_indexes
5 WHERE tablename = 'passengers';
6
7
8 SELECT * FROM passengers;
9
10 INSERT INTO passengers VALUES(201, 'Alan', 'Discra', '2006-06-24', 'Male', 'KZ', '1-1', '20
11
12 INSERT INTO passengers VALUES(201, 'Alan', 'Discra', '2006-06-24', 'Male', 'KZ', '1-1', '20
13
14
```

Below the query editor is a Data Output tab showing error messages:

ERROR: повторяющееся значение ключа нарушает ограничение уникальности "passenger_pkey"
Ключ "(passenger_id)=(201)" уже существует.

ОШИБКА: повторяющееся значение ключа нарушает ограничение уникальности "passenger_pkey"
SQL state: 23505
Detail: Ключ "(passenger_id)=(201)" уже существует.

When I created the **unique index** on `passport_number`, PostgreSQL started enforcing the rule that every value in that column must be unique.

The first passenger inserted successfully because the passport number was new.

The second insert failed with an error, because the same `passport_number` already existed in the table.

This confirms that the **unique index works correctly** — it prevents duplicate passport numbers from being added.

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer > Collations

- > Domains
- > FTS Configurations
- > FTS Dictionaries
- > FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (10)
 - > airline
 - > airport
 - > baggage
 - > baggage_check
 - > boarding_pass
 - > booking
 - > booking_flight
 - > flights
 - > passengers
 - > Columns
 - > Constraints (4)
 - age
 - date_of_birth
 - passenger_pk
 - task1
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - > security_check
 - > Trigger Functions

Processes > suppliers/postgres@PostgreSQL 17* public.booking/su... >

Query Query History Execute script F5

```

1
2
3
4 CREATE INDEX task7 ON passengers(first_name, last_name, date_of_birth, country_of_citizenship);
5
6
7
8
9
  
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 48 msec.

Total rows: Query complete 00:00:00.048

✓ Query returned successfully in 48 msec. CRLF Ln 5, Col 1 16:15 09.11.2025

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer > Collations

- > Domains
- > FTS Configurations
- > FTS Dictionaries
- > FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (10)
 - > airline
 - > airport
 - > baggage
 - > baggage_check
 - > boarding_pass
 - > booking
 - > booking_flight
 - > flights
 - > passengers
 - > Columns
 - > Constraints (4)
 - age
 - date_of_birth
 - passenger_pk
 - task1
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - > security_check
 - > Trigger Functions

Processes > suppliers/postgres@PostgreSQL 17* public.booking/su... >

Query Query History

```

1
2
3
4 CREATE INDEX task7 ON passengers(first_name, last_name, date_of_birth, country_of_citizenship);
5
6
7 EXPLAIN ANALYZE
8 SELECT * FROM passengers WHERE country_of_citizenship = 'Philippines' AND date_of_birth BETWEEN '1984-01-01' AND '1984-12-31';
9
10
11
12
13
  
```

Data Output Messages Notifications

Showing rows: 1 to 5 Page No: 1 of 1 16:20 09.11.2025

QUERY PLAN	
	text
1	Seq Scan on passengers (cost=0.00..6.52 rows=1 width=64) (actual time=0.059..0.102 rows=1 loops=1)
2	Filter: ((date_of_birth >= '1984-01-01'::date) AND (date_of_birth <= '1984-12-31'::date) AND ((country_of_citizenship)::text = 'Philippines'::text))
3	Rows Removed by Filter: 200
4	Planning Time: 0.233 ms
5	Execution Time: 0.122 ms

Total rows: 5 Query complete 00:00:00.044

CRLF Ln 8, Col 127

Although I created a composite index on (first_name, last_name, date_of_birth, country_of_citizenship), PostgreSQL did not use it because my query filters only on the last two columns.

For composite indexes, PostgreSQL can only use them effectively if the filtering starts with the leftmost

columns.

The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables' section, the 'passenger' table is selected. In the main pane, a query is run:

```
1 SELECT
2     indexname AS index_name,
3     indexdef AS definition
4 FROM
5     pg_indexes
6 WHERE
7     tablename = 'passengers';
```

The results show three indexes defined on the 'passenger' table:

index_name	definition
passenger_pkey	CREATE UNIQUE INDEX passenger_pkey ON public.passenger USING btree (passenger_id)
task6	CREATE UNIQUE INDEX task6 ON public.passenger USING btree (passport_number)
task7	CREATE INDEX task7 ON public.passenger USING btree (first_name, last_name, date_of_birth, country_of_citizenship)

Total rows: 3 Query complete 00:00:00.066

The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables' section, the 'passenger' table is selected. In the main pane, two index drop statements are run:

```
1 DROP INDEX task6;
2
3 DROP INDEX task7;
```

After running the queries, the results show only one index remains:

index_name	definition
passenger_pkey	CREATE UNIQUE INDEX passenger_pkey ON public.passenger USING btree (passenger_id)

Total rows: 1 Query complete 00:00:00.107

Successfully run. Total query runtime: 107 msec. 1 rows affected.