

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

- Object Explorer:** On the left, it lists database objects for the schema "public.flights/postgres@localhost". Under the "flights" table, the "Columns (12)" section is selected, showing columns like flight_id, sch_departure_time, and act_departure_time.
- Query Editor:** The main area contains a query window with the following SQL code:

```
1 CREATE INDEX idx_acr_dep
2 ON flights(act_departure_time)
```
- Data Output:** Below the query window, the "Data Output" tab shows the results of the query execution:
 - CREATE INDEX
 - Query returned successfully in 55 msec.
- Messages:** A green message bar at the bottom right indicates: "✓ Query returned successfully in 55 msec." (with a close button), "✓ Table rows counted: 200" (with a close button), and "✓ Table rows counted: 200" (with a close button).
- Bottom Status:** The status bar at the bottom shows "Total rows: 200" and "Query complete 00:00:00.055".

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

- Object Explorer:** On the left, it lists database objects under the schema "public.flights/postgres@localhost". It includes tables like "flights" (12 columns), "passengers" (10 columns), and "booking_flight" (2 columns). Other objects like "Columns", "Constraints", "Indexes", "RLS Policies", "Rules", and "Triggers" are also listed.
- Query Editor:** The main area contains a query window titled "public.flights/postgres@localhost". The query is:

```
1 CREATE UNIQUE INDEX idx_un_flights
2 ON flights(flight_no,sch_departure_time);
```
- Data Output:** Below the query editor, the "Data Output" tab shows the results of the query execution:
 - CREATE INDEX
 - Query returned successfully in 36 msec.
- Messages:** A green message bar at the bottom right indicates: "✓ Query returned successfully in 36 msec.", "✓ Table rows counted: 200", and "✓ Table rows counted: 200".
- Status Bar:** At the bottom, it shows "Total rows: 200" and "Query complete 00:00:00.036".

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

- Object Explorer:** On the left, it shows the database schema. Under the "flights" table, the "Columns (12)" section is selected, listing columns such as flight_id, sch_departure_time, and airline_id.
- Query Editor:** The main area contains the following SQL code:

```
1 CREATE INDEX idx_dep_arr_id
2 ON flights(departing_airport_id, arriving_airport_id);
```
- Data Output:** Below the query editor, the output shows:
 - CREATE INDEX
 - Query returned successfully in 34 msec.
- Messages:** A green message box indicates: ✓ Query returned successfully in 34 msec.
- Notifications:** Three green notification boxes at the bottom right indicate:
 - ✓ Table rows counted: 200
 - ✓ Table rows counted: 200
 - ✓ Table rows counted: 200
- Bottom Status:** Total rows: | Query complete 00:00:00.034 | LF Ln 2, Col 55

Object Explorer public.booking/postgres@localhost* public.flights/postgres@localhost* public.passengers/postgres@localhost* public.flights/postgres@localhost* public.flights/postgres@localhost*

Columns (9)

- booking_id
- flight_id
- passenger_id
- booking_platform
- created_at
- updated_at
- status
- ticket_price
- ticket_discount

Constraints

Indexes

RLS Policies

Rules

Triggers

booking_flight

flights

Columns (12)

- flight_id
- sch_departure_time
- sch_arrival_time
- departing_airport_id
- arriving_airport_id
- departing_gate
- arriving_gate
- airline_id
- act_departure_time
- act_arrival_time
- created_at
- updated_at

Constraints

Indexes

RLS Policies

Rules

Triggers

passengers

Columns (10)

Query Query History Scratch Pad

```
EXPLAIN ANALYZE
SELECT * FROM flights
WHERE departing_airport_id = 1 AND arriving_airport_id = 3;

CREATE INDEX idx_t_speed
ON flights(departing_airport_id, arriving_airport_id);

EXPLAIN ANALYZE
SELECT * FROM flights
WHERE departing_airport_id = 1 AND arriving_airport_id = 3;
```

Data Output Messages Notifications

SQL

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

QUERY PLAN

text
1 Seq Scan on flights (cost=0.00..6.00 rows=1 width=108) (actual time=0.064..0.064 rows=0 loops=1)
2 Filter: ((departing_airport_id = 1) AND (arriving_airport_id = 3))
3 Rows Removed by Filter: 200
4 Planning Time: 0.417 ms
5 Execution Time: 0.079 ms

Total rows: 5 Query complete 00:00:00.040

✓ Successfully run. Total query runtime: 40 msec. 5 rows affected.

✓ Table rows counted: 200

✓ Table rows counted: 200

LF Ln 10, Col 16

Object Explorer public.booking/postgres@localhost* public.flights/postgres@localhost* public.passengers/postgres@localhost* public.flights/postgres@localhost* public.flights/postgres@localhost*

Columns (9)

- booking_id
- flight_id
- passenger_id
- booking_platform
- created_at
- updated_at
- status
- ticket_price
- ticket_discount

Constraints

Indexes

RLS Policies

Rules

Triggers

booking_flight

flights

Columns (12)

- flight_id
- sch_departure_time
- sch_arrival_time
- departing_airport_id
- arriving_airport_id
- departing_gate
- arriving_gate
- airline_id
- act_departure_time
- act_arrival_time
- created_at
- updated_at

Constraints

Indexes

RLS Policies

Rules

Triggers

passengers

Columns (10)

Query Query History

```
EXPLAIN ANALYZE
SELECT * FROM flights
WHERE departing_airport_id = 1 AND arriving_airport_id = 2;
```

Data Output Messages Notifications

SQL

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

QUERY PLAN

text

```
1 Seq Scan on flights  (cost=0.00..6.00 rows=1 width=108) (actual time=0.089..0.090 rows=0 loops=1)
2   Filter: ((departing_airport_id = 1) AND (arriving_airport_id = 2))
3   Rows Removed by Filter: 200
4 Planning Time: 0.264 ms
5 Execution Time: 0.119 ms
```

Total rows: 5 Query complete 00:00:00.043 Rows selected: 5

✓ Table rows counted: 200 ✓ Table rows counted: 200

LF Ln 3, Col 60

Object Explorer

public.flights/postgres@localhost public.passengers/postgres@localhost public.flights/postgres@localhost public.flights/postgres@localhost public.passengers/postgres@localhost*

Query History

```
CREATE UNIQUE INDEX idx_un_p_number
ON passengers(passport_number);

INSERT INTO passengers(first_name,last_name,passport_number, date_of_birth,country_of_citizenship)
VALUES('Aigerym', 'Kalaubaeva', 'p12345', '1990-11-23', 'Kazakhstan');

INSERT INTO passengers(first_name,last_name,passport_number, date_of_birth,country_of_citizenship)
VALUES('Dias', 'Serikbay', 'p12345', '1990-01-13', 'Kazakhstan');
```

Data Output

Messages Notifications

ERROR: null value in column "passenger_id" of relation "passengers" violates not-null constraint
Failing row contains (null, Aigerym, Kalaubaeva, 1990-11-23, null, Kazakhstan, null, p12345, null, null).
SQL state: 23502
Detail: Failing row contains (null, Aigerym, Kalaubaeva, 1990-11-23, null, Kazakhstan, null, p12345, null, null).

Table rows counted: 200 ✓

Table rows counted: 200 ✓

Total rows: 200 | Query complete 00:00:00.043 | LF Ln 8, Col 66

Object Explorer

public.passengers... public.flights/postgres@localhost public.flights/postgres@localhost public.passengers... public.passengers/postgres@localhost*

public.passengers/postgres@localhost

No limit

Query History

Scratch Pad

Query

```
1 CREATE UNIQUE INDEX idx_un_pass
2 ON passengers(passport_number);
3
4 INSERT INTO passengers(first_name, last_name, date_of_birth, gender, country_of_citizenship, country_of_residence)
5 VALUES ('Aigerym', 'Kalaubaeva', '1990-01-12', 'Female', 'Kazakhstan', 'UK', 'p12367');
6
7 INSERT INTO passengers(first_name, last_name, date_of_birth, gender, country_of_citizenship, country_of_residence)
8 VALUES ('Aigerym', 'Kalaubaeva', '1990-01-12', 'Female', 'Kazakhstan', 'UK', 'p12367')
```

Data Output

Messages

Notifications

ERROR: column "pasport_number" does not exist

SQL state: 42703

Total rows: 200 | Query complete 00:00:00.060 | ✓ Table rows counted: 200 | ✓ Table rows counted: 200 | LF | Ln 8, Col 17

Object Explorer

public.passengers... public.flights/postgres@localhost public.flights/postgres@localhost public.passengers... public.passengers/postgres@localhost*

public.passengers/postgres@localhost

No limit

Query History

Scratch Pad

Query

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

Columns (10)

- arriving_airport_id
- departing_gate
- arriving_gate
- airline_id
- act_departure_time
- act_arrival_time
- created_at
- updated_at

passenger_id

first_name

last_name

date_of_birth

gender

country_of_citizenship

country_of_residence

passport_number

created_at

updated_at

passenger_id

first_name

last_name

date_of_birth

gender

country_of_citizenship

country_of_residence

passport_number

created_at

updated_at

Constraints

Indexes

RLS Policies

Rules

Triggers

passengers

security_check

Trigger Functions

Types

Views

Subscriptions

Login/Group Roles

Tablespaces

Data Output

Messages

Notifications

Showing rows: 1 to 5

Page No: 1 of 1

QUERY PLAN

text

1 Seq Scan on passengers (cost=0.00..9.50 rows=1 width=73) (actual time=0.033..0.034 rows=0 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: 0.265 ms

5 Execution Time: 0.048 ms

Total rows: 5 | Query complete 00:00:00.045 | LF | Ln 2, Col 6

✓ Table rows counted: 200

✓ Table rows counted: 200

Object Explorer

public.passengers... public.flights/postgres@localhost public.flights/postgres@localhost public.passengers... public.passengers/postgres@localhost*

public.passengers/postgres@localhost

No limit

Query History

Scratch Pad

Query

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

Data Output

Messages

Notifications

Showing rows: 1 to 5

Page No: 1 of 1

SQL

QUERY PLAN

text

1 Seq Scan on passengers (cost=0.00..9.50 rows=1 width=73) (actual time=0.033..0.034 rows=0 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: 0.265 ms
5 Execution Time: 0.048 ms

✓ Table rows counted: 200

✓ Table rows counted: 200

Total rows: 5 | Query complete 00:00:00.045 | LF | Ln 2, Col 6

Object Explorer

public.passengers... public.flights/post... public.flights/post... public.passengers... public.passengers/postgres@localhost*

public.passengers/postgres@localhost

No limit

Query History

Scratch Pad

Query

```
1 SELECT indexname, indexdef
2 FROM pg_indexes
3 WHERE tablename = 'passengers';
4
5 DROP INDEX IF EXISTS idx_passenger_info;
6 DROP INDEX IF EXISTS idx_unique_passport;
```

Data Output

Messages

Notifications

NOTICE: index "idx_passenger_info" does not exist, skipping
NOTICE: index "idx_unique_passport" does not exist, skipping
DROP INDEX

Query returned successfully in 47 msec.

Total rows: Query complete 00:00:00.047

✓ Query returned successfully in 47 msec. ✓
✓ Table rows counted: 200 ✓
✓ Table rows counted: 200 ✓

LF Ln 6, Col 40