

1)

Query History

```
1 CREATE INDEX idx_actual_departure
2 ON flights (actual_departure);
3
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 117 msec.

Total rows: Query complete 00:00:00.117 CRLF Ln 3, Col 1

2)

Query History

```
1 CREATE UNIQUE INDEX idx_unique_flight_no_sched_dep
2 ON flights (flight_no, scheduled_departure);
3
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 61 msec.

Total rows: Query complete 00:00:00.061 ✓ CRLF Ln 3, Col 1

3)

Query Query History

```
1 CREATE INDEX idx_dep_arr_airports
2 ON flights (departure_airport_id, arrival_airport_id);
3
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 97 msec.

Total rows: 0 Query complete 00:00:00.097

✓ Query returned successfully in 97 msec. ✘

CRLF Ln 3, Col 1

4)

Query Query History

```
1 EXPLAIN ANALYZE
2 SELECT *
3 FROM passengers
4 WHERE country_of_citizenship = 'Philippines'
5   AND date_of_birth BETWEEN '1984-01-01' AND '1984-12-31';
```

Data Output Messages Notifications

≡+ 📁 ↻ 🔍 ↴ ↵ 🗑️ ↲ ↳ SQL

Showing rows: 1 to 5 🖊 Page No: 1 of 1 ⏪ ⏪ ⏫ ⏬ ⏭

QUERY PLAN	
text	
1	Seq Scan on passengers (cost=0.00..2877.00 rows=1 width=57) (actual time=0.028..18.195 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: 99999
4	Planning Time: 3.046 ms
5	Execution Time: 18.230 ms

Total rows: 5 Query complete 00:00:00.084

✓ Successfully run. Total query runtime: 84 msec. 5 rows affected. ✘

CRLF Ln 1, Col 9

Query    Query History

```
1 CREATE INDEX idx_passengers_citizenship_dob ON passengers (country_of_citizenship, date_of_birth);
```

Data Output    Messages    Notifications

```
CREATE INDEX
```

Query returned successfully in 471 msec.

Total rows: 4    Query complete 00:00:00.471    CRLF    Ln 1, Col 99

```
Query    Query History
```

```
1 EXPLAIN ANALYZE
2 SELECT *
3 FROM passengers
4 WHERE country_of_citizenship = 'Philippines'
5   AND date_of_birth BETWEEN '1984-01-01' AND '1984-12-31';
```

Data Output    Messages    Notifications

Showing rows: 1 to 4    Page No: 1 of 1    |< << <> >> >|

	QUERY PLAN
1	text Index Scan using idx_passengers_citizenship_dob on passengers (cost=0.29..8.32 rows=1 width=57) (actual time=0.087..0.089 rows=1 loops=1)
2	Index Cond: (((country_of_citizenship)::text = 'Philippines'::text) AND (date_of_birth >= '1984-01-01'::date) AND (date_of_birth <= '1984-12-31'::date))
3	Planning Time: 1.934 ms
4	Execution Time: 0.113 ms

Total rows: 4    Query complete 00:00:00.107    CRLF    Ln 1, Col 9

5)

Query History

```
1 EXPLAIN ANALYZE
2 SELECT * FROM flights
3 WHERE departure_airport_id = 1 AND arrival_airport_id = 2;
4
```

Data Output Messages Notifications

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

QUERY PLAN
text
1 Bitmap Heap Scan on flights (cost=4.30..9.97 rows=2 width=61) (actual time=0.073..0.079 rows=2 loops=1)
2 Recheck Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 2))
3 Heap Blocks: exact=2
4 -> Bitmap Index Scan on idx_dep_arr_airports (cost=0.00..4.29 rows=2 width=0) (actual time=0.048..0.048 rows=2 loops=1)
5 Index Cond: ((departure_airport_id = 1) AND (arrival_airport_id = 2))
6 Planning Time: 0.264 ms
7 Execution Time: 0.130 ms

Total rows: 7 Query complete 00:00:00.008 CRLF Ln 4 Col 1

6)

Query History

```
1 CREATE UNIQUE INDEX idx_unique_passport
2 ON passengers (passport_number);
3
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 208 msec.

Total rows: 0 Query complete 00:00:00.208 ✓ CRLF Ln 2. Col 31

## Query Query History

```

1  INSERT INTO passengers (
2      passenger_id, first_name, last_name, date_of_birth, gender,
3      country_of_citizenship, country_of_residence, passport_number,
4      created_at, update_at
5  ) VALUES (
6      201 , 'Айгерим', 'Каримова', '1995-03-15'::DATE, 'Female',
7      'KZ', 'KZ', 'N1234567',
8      CURRENT_DATE, CURRENT_DATE
9  );

```

## Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 62 msec.

Total rows: 1 Query complete 00:00:00.062

CRLF Ln 6, Col 10

```

1  INSERT INTO passengers (
2      passenger_id, first_name, last_name, date_of_birth, gender,
3      country_of_citizenship, country_of_residence, passport_number,
4      created_at, update_at
5  ) VALUES (
6      202 , 'Айгерим', 'Каримова', '1995-03-15'::DATE, 'Female',
7      'KZ', 'KZ', 'N1234567',
8      CURRENT_DATE, CURRENT_DATE
9  );

```

## Data Output Messages Notifications

ERROR: повторяющееся значение ключа нарушает ограничение уникальности "unique\_name"  
 Ключ "(last\_name, first\_name, date\_of\_birth)=(Каримова, Айгерим, 1995-03-15)" уже существует.

ОШИБКА: повторяющееся значение ключа нарушает ограничение уникальности "unique\_name"  
 SQL state: 23505  
 Detail: Ключ "(last\_name, first\_name, date\_of\_birth)=(Каримова, Айгерим, 1995-03-15)" уже существует.

Total rows: 1 Query complete 00:00:00.068

CRLF Ln 6, Col 9

7)

Query Query History

```
1 CREATE INDEX idx_passenger_details
2 ON passengers (first_name, last_name, date_of_birth, country_of_citizenship);
3
```

Data Output Messages Notifications

CREATE INDEX

Query returned successfully in 73 msec.

Total rows: 1 Query complete 00:00:00.073

✓ Query returned successfully in 73 msec. ✎  
CRLF Ln 3, Col 1

Query Query History

```
1 SET enable_seqscan = OFF;
2
3 EXPLAIN ANALYZE
4 SELECT * FROM passengers
5 WHERE country_of_citizenship = 'PH'
6     AND date_of_birth BETWEEN '1984-01-01' AND '1984-12-31';
```

Data Output Messages Notifications

	QUERY PLAN	Sh
1	Bitmap Heap Scan on passengers (cost=10.16..13.28 rows=1 width=64) (actual time=0.053..0.054 rows=0 loops=1)	🔒
2	Recheck Cond: ((date_of_birth >= '1984-01-01'::date) AND (date_of_birth <= '1984-12-31'::date))	
3	Filter: ((country_of_citizenship)::text = 'PH'::text)	
4	Rows Removed by Filter: 6	
5	Heap Blocks: exact=2	
6	-> Bitmap Index Scan on unique_name (cost=0.00..10.15 rows=7 width=0) (actual time=0.030..0.030 rows=6 loop...	
7	Index Cond: ((date_of_birth >= '1984-01-01'::date) AND (date_of_birth <= '1984-12-31'::date))	
8	Planning Time: 0.318 ms	
9	Execution Time: 0.080 ms	

8)

Query History

```
1 SELECT indexname, indexdef
2   FROM pg_indexes
3 WHERE tablename = 'passengers';
4
```

Data Output Messages Notifications

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

indexname name	indexdef text
1 passengers_pkey	CREATE UNIQUE INDEX passengers_pkey ON public.passengers USING btree (passenger_id)
2 unique_passenger_email	CREATE UNIQUE INDEX unique_passenger_email ON public.passengers USING btree (passenger_id)
3 unique_name	CREATE UNIQUE INDEX unique_name ON public.passengers USING btree (last_name, first_name, date_of_birth)
4 idx_unique_passport	CREATE UNIQUE INDEX idx_unique_passport ON public.passengers USING btree (passport_number)
5 idx_passenger_details	CREATE INDEX idx_passenger_details ON public.passengers USING btree (first_name, last_name, date_of_birth, country_of_citizenship)

Total rows: 5 Query complete 00:00:00.200 ✓ Successfully run. Total query runtime: 200 msec. 5 rows affected. CRLF Ln 4, Col 1

Query History

```
1 DROP INDEX IF EXISTS idx_actual_departure;
2 DROP INDEX IF EXISTS idx_unique_flight_no_sched_dep;
3 DROP INDEX IF EXISTS idx_dep_arr_airports;
4 DROP INDEX IF EXISTS idx_unique_passport;
5 DROP INDEX IF EXISTS idx_passenger_details;
6
```

Data Output Messages Notifications

DROP INDEX

Query returned successfully in 83 msec.

Total rows: 5 Query complete 00:00:00.083 CRLF Ln 6, Col 1