

The screenshot shows the pgAdmin 4 interface. At the top, there's a menu bar with File, Object, Tools, Edit, View, Window, Help. Below the menu is a toolbar with various icons for database management. The main area has two tabs: 'Welcome' and 'postgres/postgres@PostgreSQL 17\*'. The current tab is 'postgres/postgres@PostgreSQL 17'. In the 'Query' tab, there is a SQL script:

```
begin
delete from Booking
where booking_id = 105;
delete from Booking
where booking_id = 999999999;
commit;
select * from Booking
where booking_id = 105;
```

Below the query editor is a 'Data Output' section which is currently empty. At the bottom, there's a status bar showing 'Total rows: 0' and 'Query complete 00:00:00.096'.

On the right side of the interface, there's a table definition for the 'Booking' table:

	booking_id	flight_id	passenger_id	booking_platform	created_at	updated_at	status	ticket_price	ticket_discount
[PK]	integer	integer	integer	character varying (50)	timestamp without time zone	timestamp without time zone	character varying (50)	numeric (7,2)	numeric (3,2)

1. A passenger cancels their booking. You need to remove the booking for the flight. Ensure the 'booking' table no longer contains the booking. Simulate an error to test rollback (for example, invalid booking\_id).

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* × postgres/postgres@PostgreSQL 17

No limit E

Query History

Query Scratch Pad

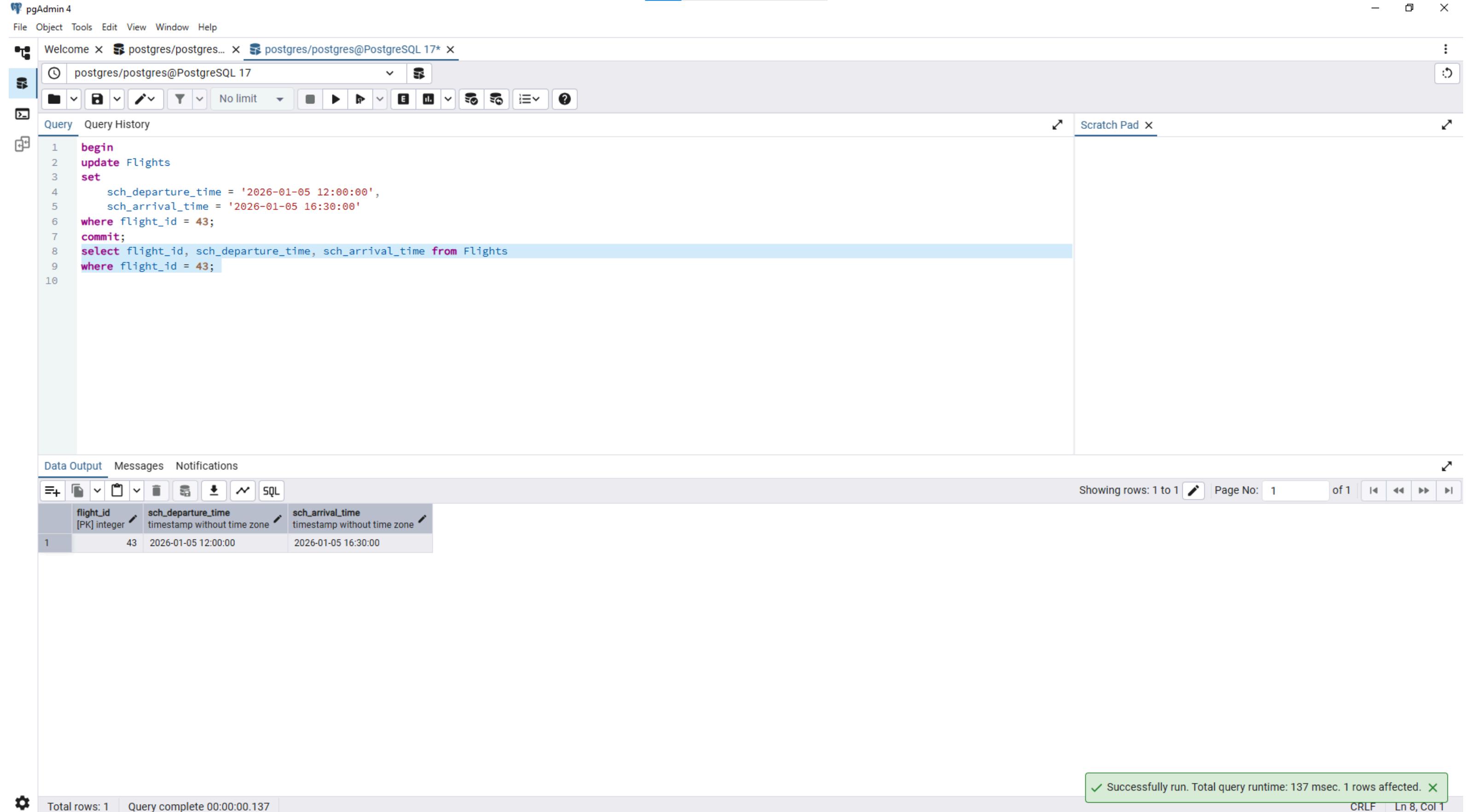
```
begin
update Flights
set
    sch_departure_time = '2026-01-05 12:00:00',
    sch_arrival_time = '2026-01-05 16:30:00'
where flight_id = 43;
commit;
select flight_id, sch_departure_time, sch_arrival_time from Flights
where flight_id = 43;
```

Data Output Messages Notifications

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

	flight_id [PK] integer	sch_departure_time timestamp without time zone	sch_arrival_time timestamp without time zone
1	43	2026-01-05 12:00:00	2026-01-05 16:30:00

Total rows: 1 Query complete 00:00:00.137 ✓ Successfully run. Total query runtime: 137 msec. 1 rows affected. CRLF Ln 8, Col 1



2. Rescheduling a flight. You need to reschedule a flight. Verify the 'flights' table reflects the new departure time. Simulate an error to test rollback (for example, invalid flight\_id).

The screenshot shows the pgAdmin 4 interface. At the top, there are two tabs: "Welcome" and "postgres/postgres@PostgreSQL 17\*". The second tab is active, showing a SQL query window with the following code:

```
begin
update Booking
set
    ticket_price = ticket_price * 0.99
where flight_id = 136;
commit;
select booking_id, flight_id, ticket_price from Booking
where flight_id = 136;
```

Below the query window is a "Data Output" window displaying the results of the query. The results are as follows:

	booking_id [PK] integer	flight_id integer	ticket_price numeric (7,2)
1	81	136	926.27
2	82	136	703.61
3	85	136	810.83
4	86	136	668.29
5	88	136	876.96
6	89	136	810.97
7	92	136	678.52
8	93	136	965.43
9	95	136	653.32
10	96	136	940.94
11	97	136	741.38
12	99	136	662.85
13	100	136	1001.50
14	103	136	938.67

At the bottom of the pgAdmin window, it says "Total rows: 131 Query complete 00:00:00.124".

3. Updating ticket prices. You need to decrease the ticket price for a specific flight for all existing bookings. If an error occurs, no changes should be applied.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17

No limit

Query History

Scratch Pad ×

```
begin
update Passengers
set
    first_name = 'Misha',
    last_name = 'Litvin',
    country_of_citizenship = 'France',
    updated_at = NOW()
where passenger_id = 50;

select passenger_id, first_name, last_name, country_of_citizenship
from Passengers
where passenger_id = 50;

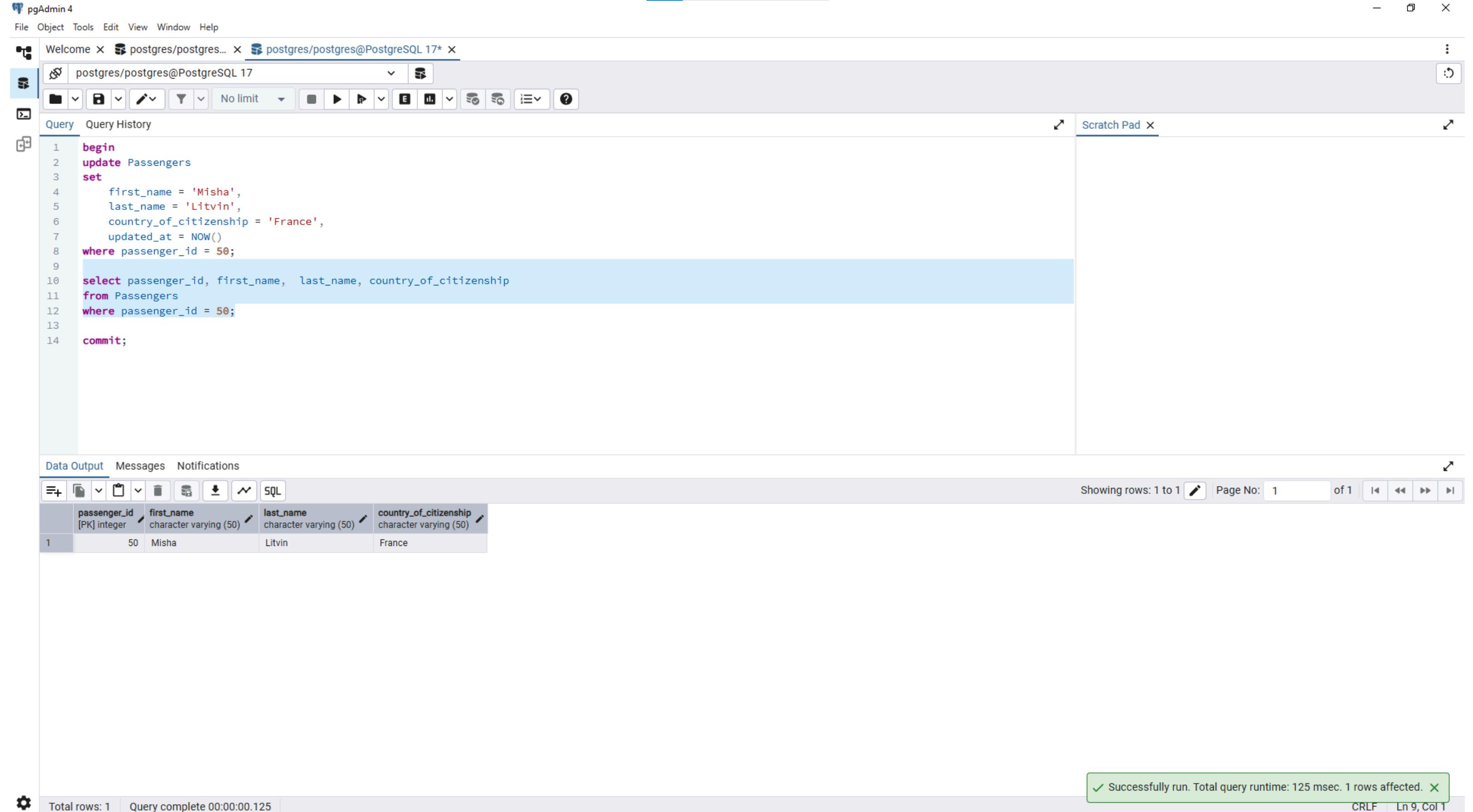
commit;
```

Data Output Messages Notifications

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

	passenger_id [PK] integer	first_name character varying (50)	last_name character varying (50)	country_of_citizenship character varying (50)
1	50	Misha	Litvin	France

Total rows: 1 Query complete 00:00:00.125 ✓ Successfully run. Total query runtime: 125 msec. 1 rows affected. CRLF Ln 9, Col 1



4. A passenger updates their details. Ensure the update is reflected across all associated records, including bookings.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17 No limit E

Query History

Query Scratch Pad

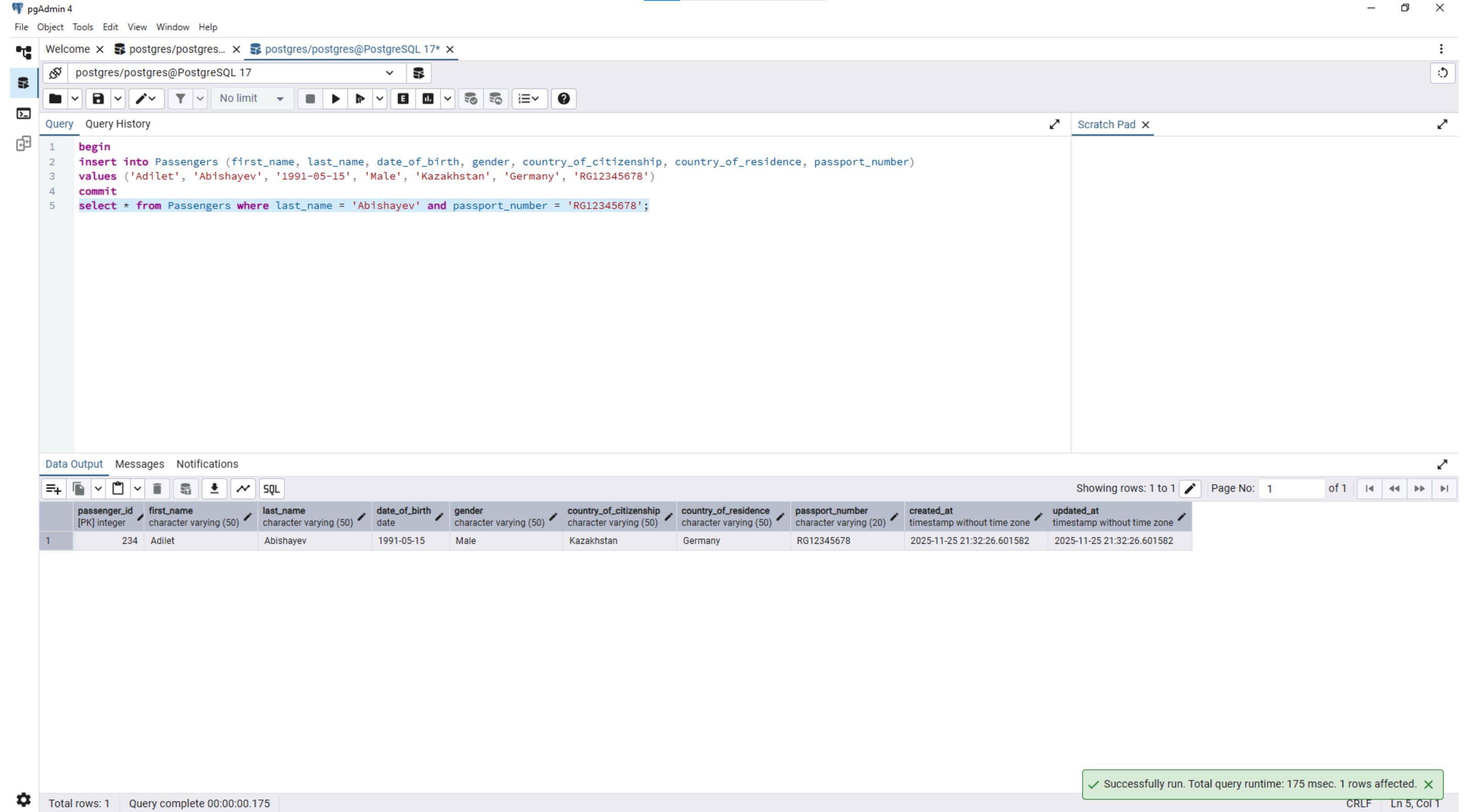
```
begin
insert into Passengers (first_name, last_name, date_of_birth, gender, country_of_citizenship, country_of_residence, passport_number)
values ('Adilet', 'Abishayev', '1991-05-15', 'Male', 'Kazakhstan', 'Germany', 'RG12345678')
commit
select * from Passengers where last_name = 'Abishayev' and passport_number = 'RG12345678';
```

Data Output Messages Notifications

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

	passenger_id [PK] integer	first_name character varying (50)	last_name character varying (50)	date_of_birth date	gender character varying (50)	country_of_citizenship character varying (50)	country_of_residence character varying (50)	passport_number character varying (20)	created_at timestamp without time zone	updated_at timestamp without time zone
1	234	Adilet	Abishayev	1991-05-15	Male	Kazakhstan	Germany	RG12345678	2025-11-25 21:32:26.601582	2025-11-25 21:32:26.601582

Total rows: 1 Query complete 00:00:00.175 ✓ Successfully run. Total query runtime: 175 msec. 1 rows affected. CRLF Ln 5, Col 1



5. A new passenger is registered, and a booking is created.  
Ensure the new passenger is added and the booking succeeds.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17

No limit

Query History

Scratch Pad ×

```
begin
update Booking
set
    ticket_price = ticket_price + 50.00,
    updated_at = NOW()
where flight_id = 136;
commit;
select booking_id, flight_id, ticket_price from Booking
where flight_id = 136;
```

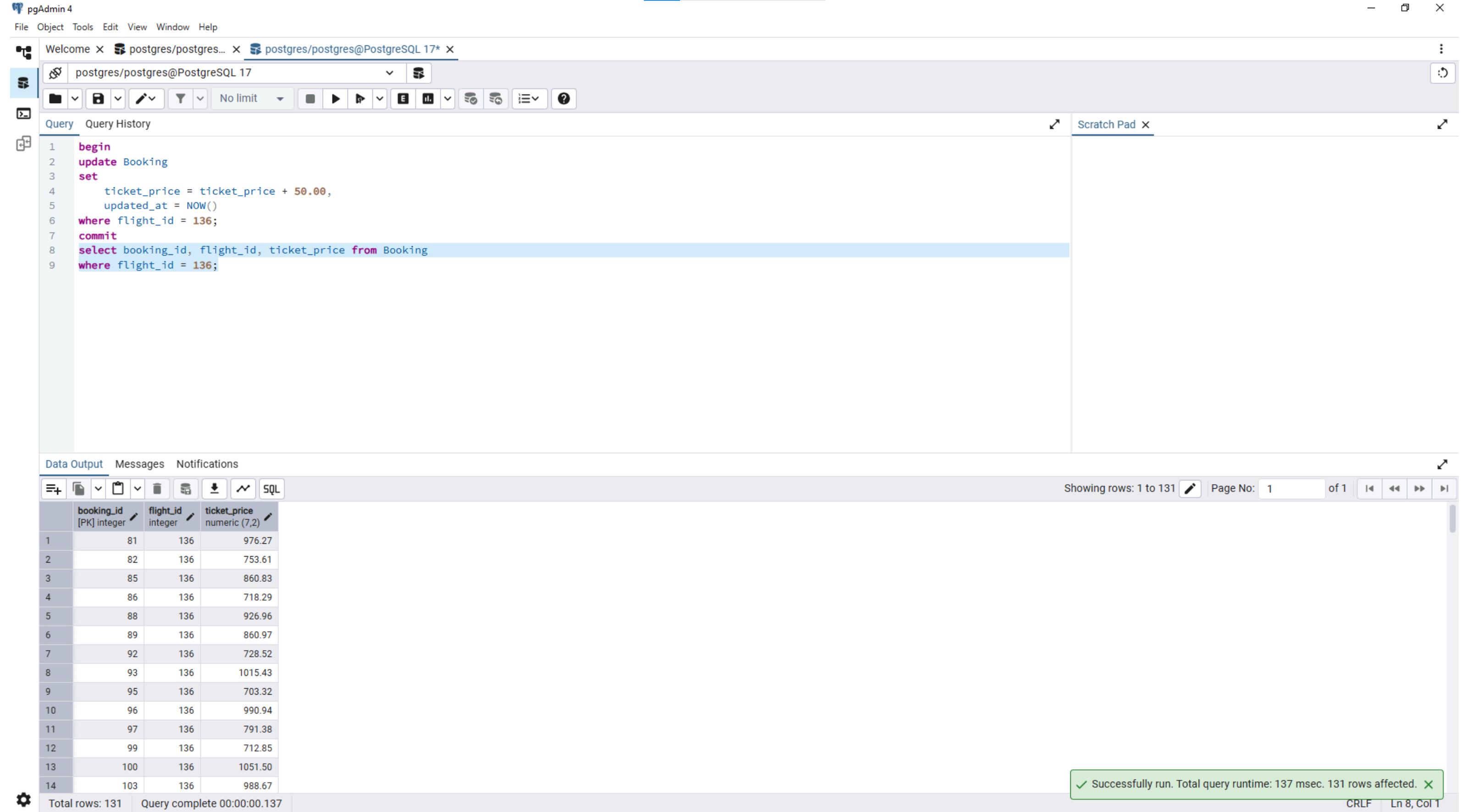
Data Output Messages Notifications

Showing rows: 1 to 131 Page No: 1 of 1

	booking_id [PK] integer	flight_id integer	ticket_price numeric (7,2)
1	81	136	976.27
2	82	136	753.61
3	85	136	860.83
4	86	136	718.29
5	88	136	926.96
6	89	136	860.97
7	92	136	728.52
8	93	136	1015.43
9	95	136	703.32
10	96	136	990.94
11	97	136	791.38
12	99	136	712.85
13	100	136	1051.50
14	103	136	988.67

Successfully run. Total query runtime: 137 msec. 131 rows affected.

Total rows: 131 Query complete 00:00:00.137 CRLF Ln 8, Col 1



6. Increase the ticket price for all bookings on a specific flight by a fixed amount.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17

No limit

Query History

Query Scratch Pad

```
begin
update Baggage
set
    weight_in_kg = 19.80,
    updated_at = NOW()
where baggage_id = 103;
commit;
select baggage_id, weight_in_kg from Baggage
where baggage_id = 103;
```

Data Output Messages Notifications

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

	baggage_id [PK] integer	weight_in_kg numeric (4,2)
1	103	19.80

Successfully run. Total query runtime: 166 msec. 1 rows affected.

Total rows: 1 Query complete 00:00:00.166 CRLF Ln 8, Col 1

A screenshot of the pgAdmin 4 interface. The top menu bar includes File, Object, Tools, Edit, View, Window, and Help. The title bar shows 'Welcome × postgres/postgres@PostgreSQL 17\* ×'. The main window has two tabs: 'Query' (selected) and 'Scratch Pad'. The 'Query' tab contains a SQL script with a transaction block, an update statement, and a select statement. The 'Scratch Pad' tab is empty. Below the query editor is a 'Data Output' section with tabs for 'Messages' and 'Notifications'. It shows a table with one row: baggage\_id 103 and weight\_in\_kg 19.80. A status message at the bottom right says 'Successfully run. Total query runtime: 166 msec. 1 rows affected.' The bottom status bar shows 'Total rows: 1' and 'Query complete 00:00:00.166'.

7. Update a baggage weight. A passenger updates the declared weight of their baggage. Ensure that the change is correctly reflected in the database.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17

No limit

Query History

Scratch Pad ×

```
begin
update Booking
set
    ticket_price = ticket_price * 0.80,
    updated_at = NOW()
where booking_id = 125;
commit
select booking_id, ticket_price from Booking where booking_id = 125;
```

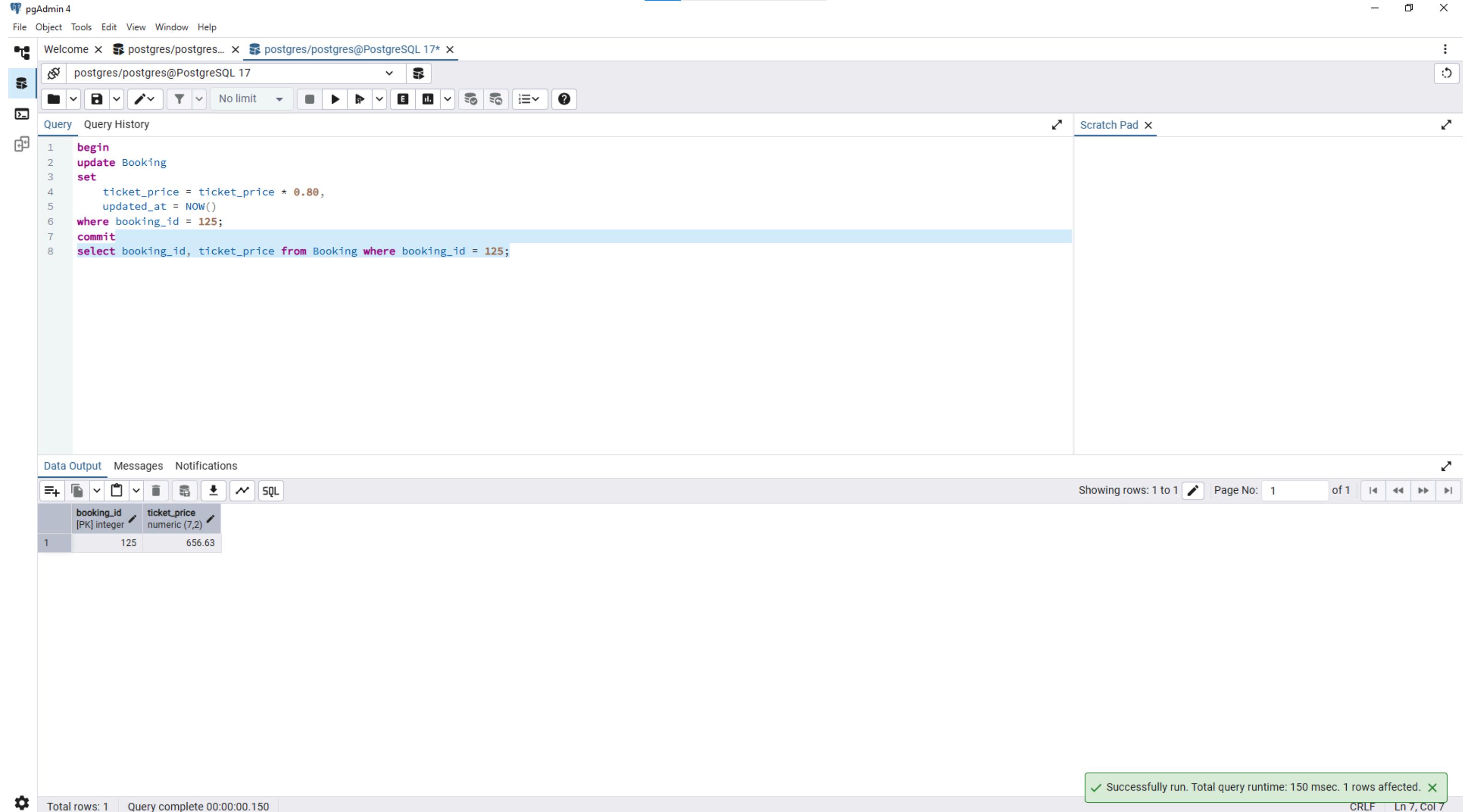
Data Output Messages Notifications

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

	booking_id	ticket_price
1	125	656.63

Successfully run. Total query runtime: 150 msec. 1 rows affected.

Total rows: 1 Query complete 00:00:00.150 CRLF Ln 7, Col 7



8. Apply a discount to a booking for a specific passenger. If any error occurs, roll back.

pgAdmin 4

File Object Tools Edit View Window Help

Welcome × postgres/postgres@PostgreSQL 17\* ×

postgres/postgres@PostgreSQL 17

No limit

Query History

Scratch Pad ×

```
begin
update Booking
set
  flight_id = 201,
  updated_at = NOW(),
  status = 'Rescheduled'
where flight_id = 101;
commit
select booking_id, flight_id, status from Booking
where booking_id = 167;
```

Data Output Messages Notifications

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

	booking_id	flight_id	status
1	167	136	Rescheduled

Total rows: 1 Query complete 00:00:00.147 ✓ Successfully run. Total query runtime: 147 msec. 1 rows affected. CRLF Ln 9, Col 1

The screenshot shows the pgAdmin 4 interface with a single active connection tab labeled 'postgres/postgres@PostgreSQL 17\*'. The main area displays a SQL script for updating a 'Booking' table. The script begins with a 'begin' statement, followed by an 'update' statement setting 'flight\_id' to 201, 'updated\_at' to the current time, and 'status' to 'Rescheduled' for rows where 'flight\_id' is 101. It then includes a 'commit' statement and a final 'select' statement to retrieve the updated row where 'booking\_id' is 167. Below the script, the 'Data Output' tab is selected, showing a single row in a table with columns 'booking\_id', 'flight\_id', and 'status'. The row has values 167, 136, and 'Rescheduled' respectively. A green success message at the bottom right indicates the query was successfully run with a runtime of 147 msec and 1 row affected.

9. Reschedule all bookings for a flight to a new flight.