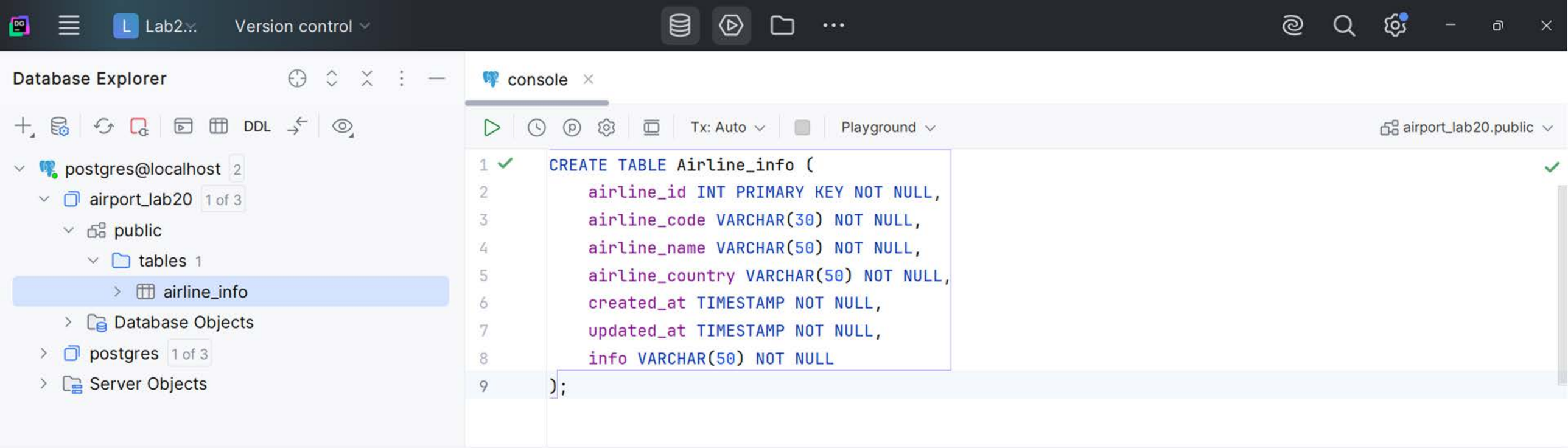
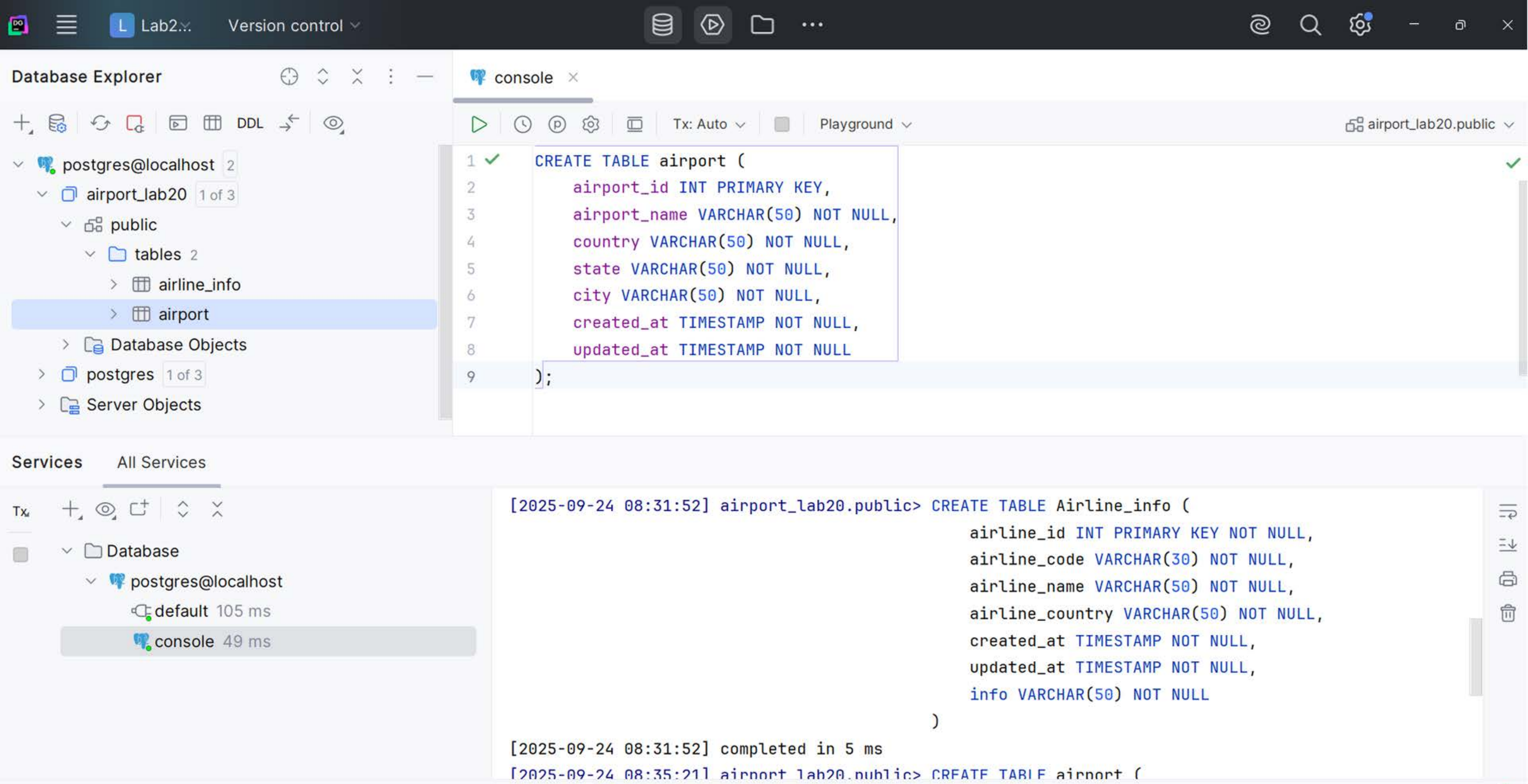


DDL

1. Create following tables with corresponding attributes:

Airline_info: airline_id (int), airline_code (varchar (30)),airline_name (varchar(50)),airline_country(varchar(50)), created_at(timestamp), updated_at(timestamp), info(varchar(50)) ;

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

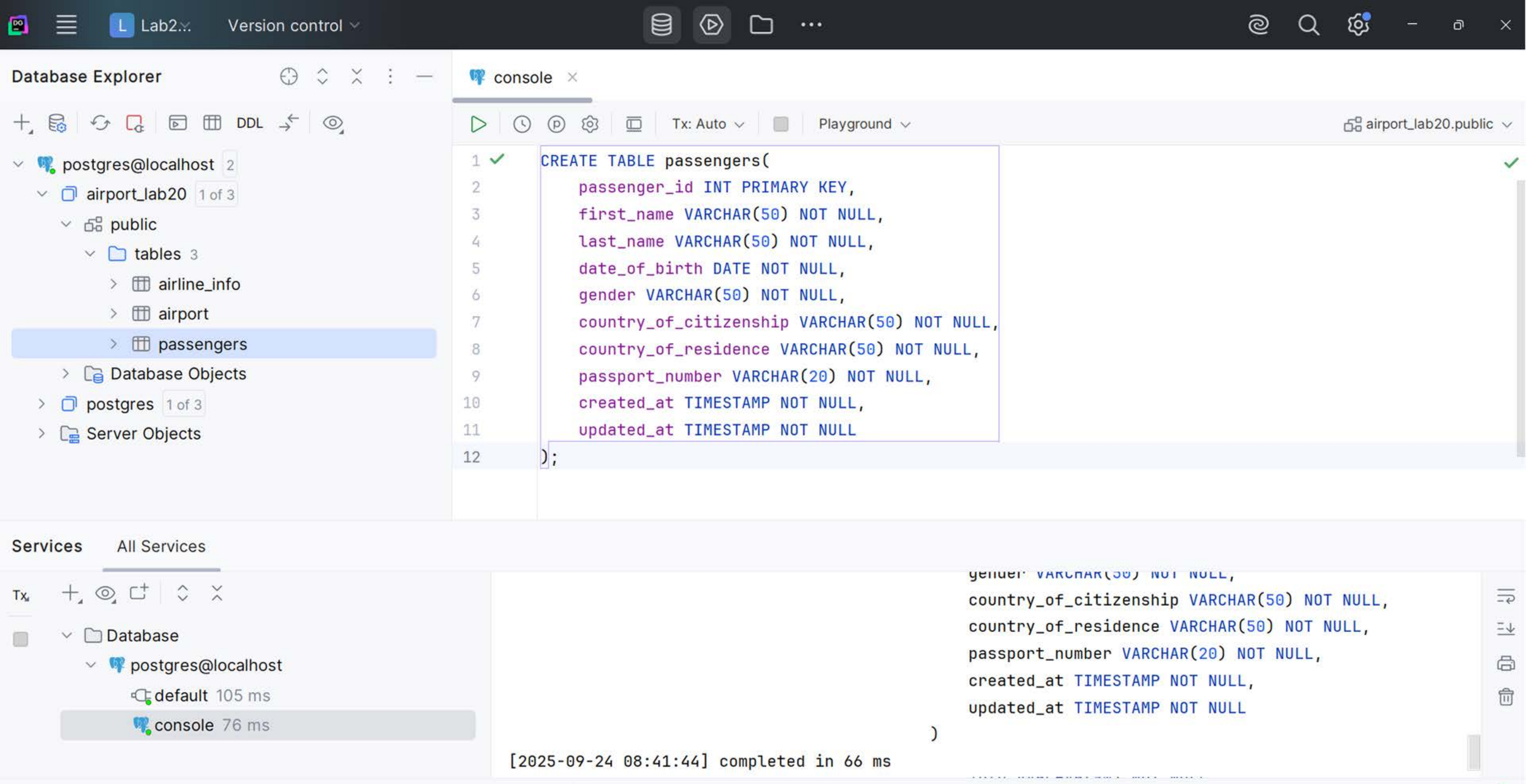


DDL

1. Create following tables with corresponding attributes:

Airport: airport_id(int), airport_name (varchar(50)),country (varchar(50)), state (varchar(50)), city(varchar(50)), created_at(timestamp), updated_at(timestamp) ;

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

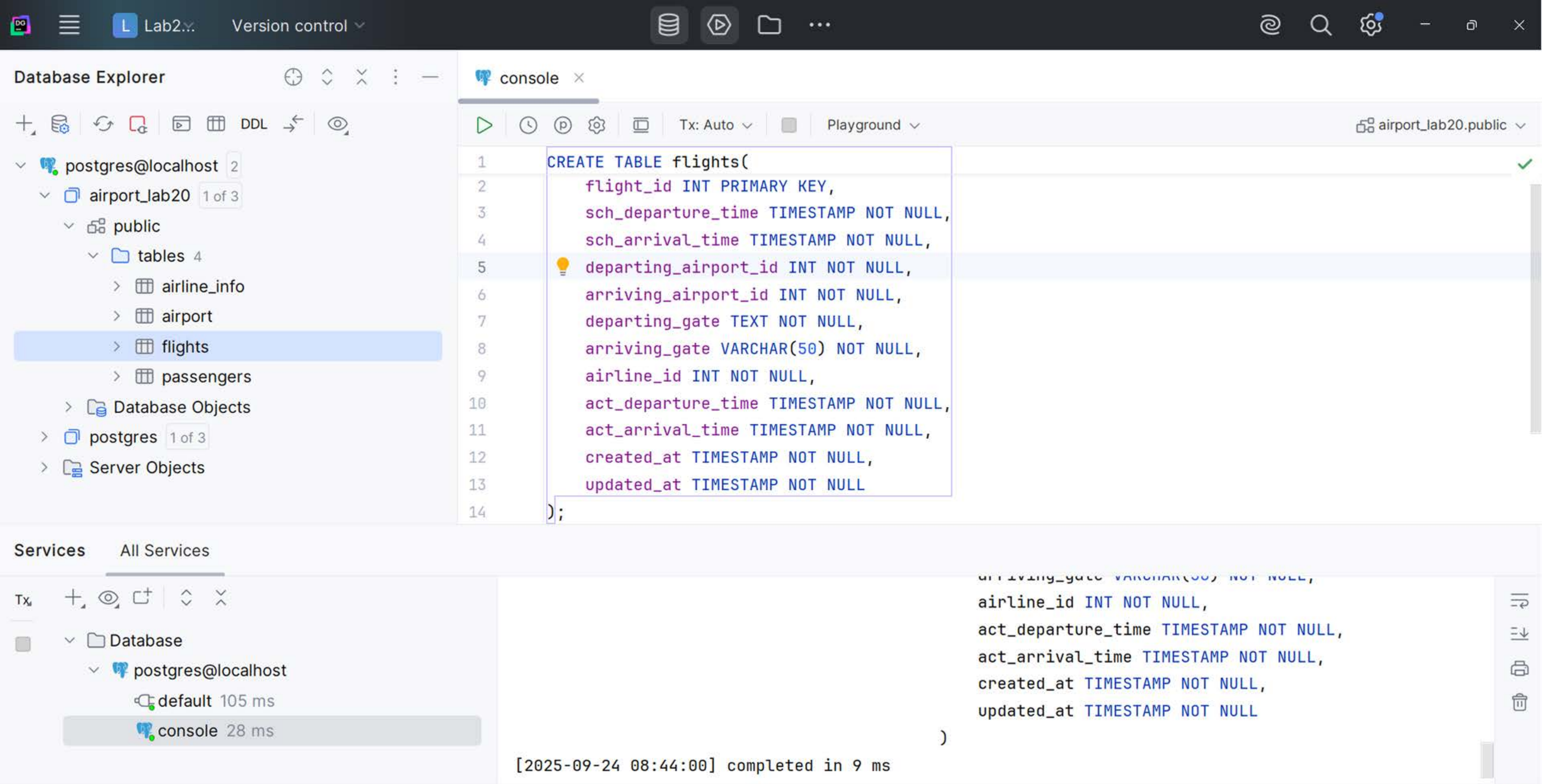


DDL

1. Create following tables with corresponding attributes:

Passengers: passenger_id(int), first_name(varchar(50)), last_name(varchar(50)), date_of_birth(date), gender(varchar(50)), country_of_citizenship(varchar(50)), country_of_residence(varchar(50)), passport_number(varchar(20)), created_at(timestamp), updated_at(timestamp) ;

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

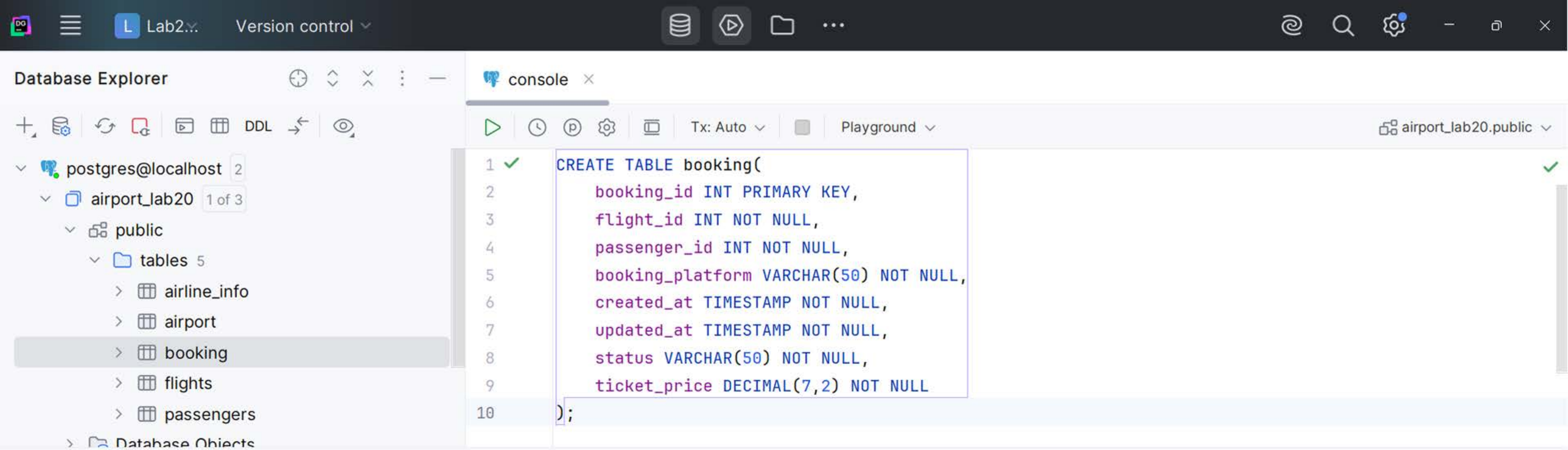


DDL

1. Create following tables with corresponding attributes:

Flights: flight_id(int), sch_departure_time(timestamp), sch_arrival_time(timestamp), departing_airport_id(int), arriving_airport_id(int), departing_gate(varchar(50)), arriving_gate(varchar(50)), airline_id(int), act_departure_time(timestamp), act_arrival_time(timestamp), created_at(timestamp), updated_at(timestamp) ;

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

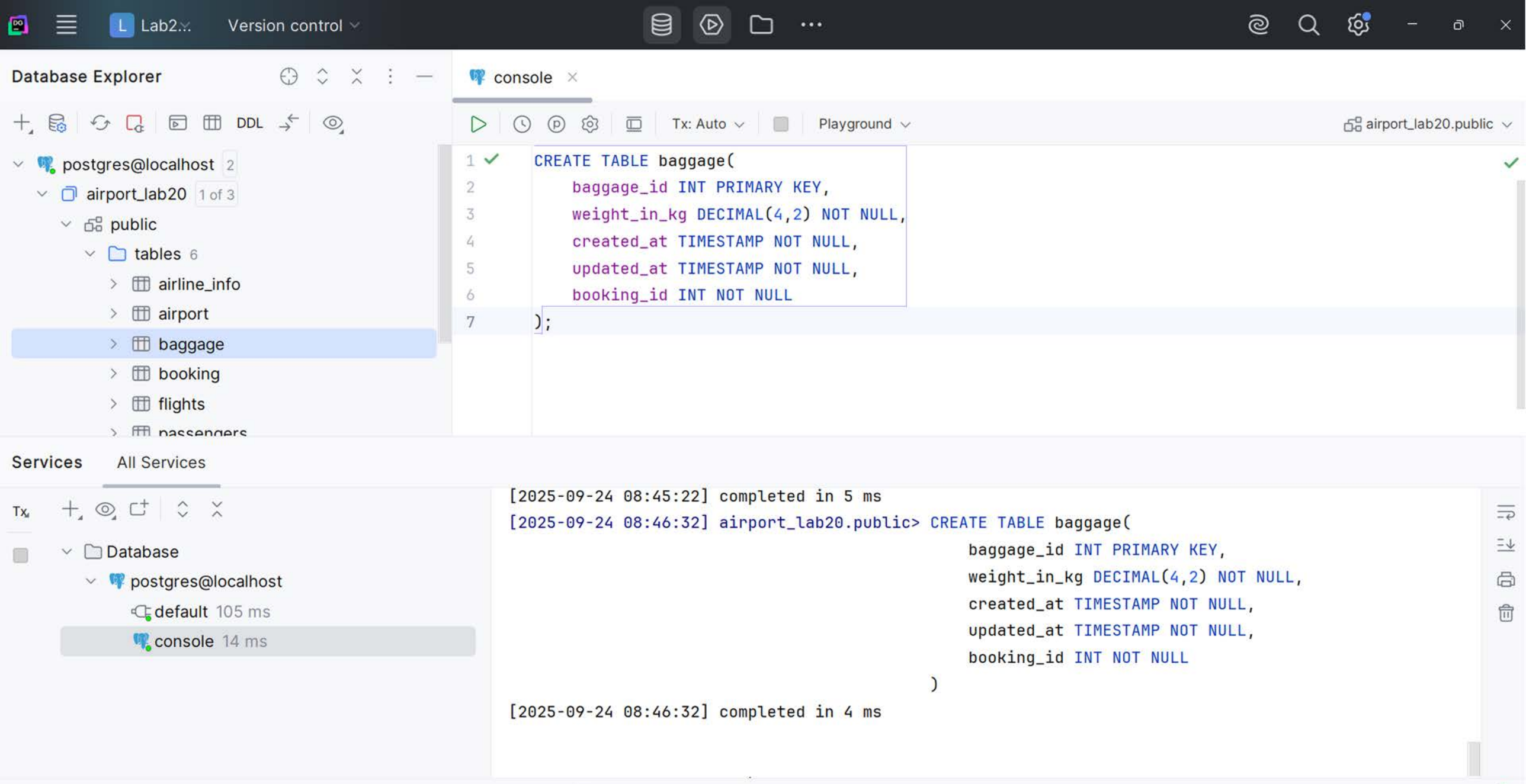


DDL

1. Create following tables with corresponding attributes:

Booking: booking_id(int), flight_id(int), passenger_id(int), booking_platform(varchar(50)), created_at(timestamp), updated_at(timestamp), status(varchar(50)), price(decimal(7,2));

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

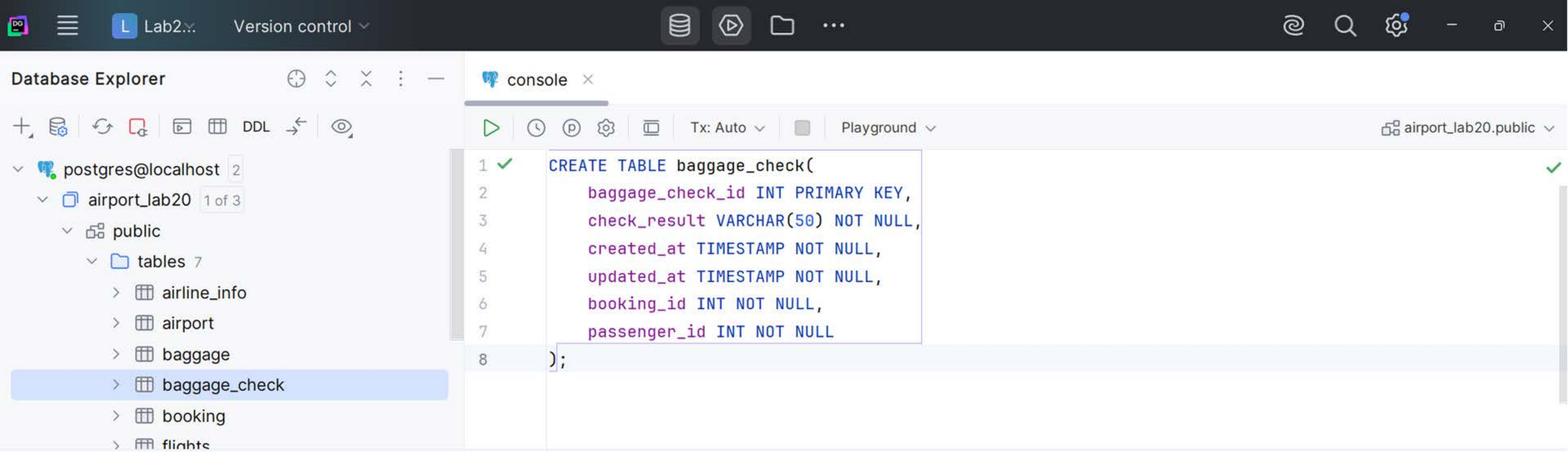


DDL

1. Create following tables with corresponding attributes:

Baggage: baggage_id(int), weight_in_kg (decimal(4,2)), created_at(timestamp, updated_at(timestamp) , booking_id(int);

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

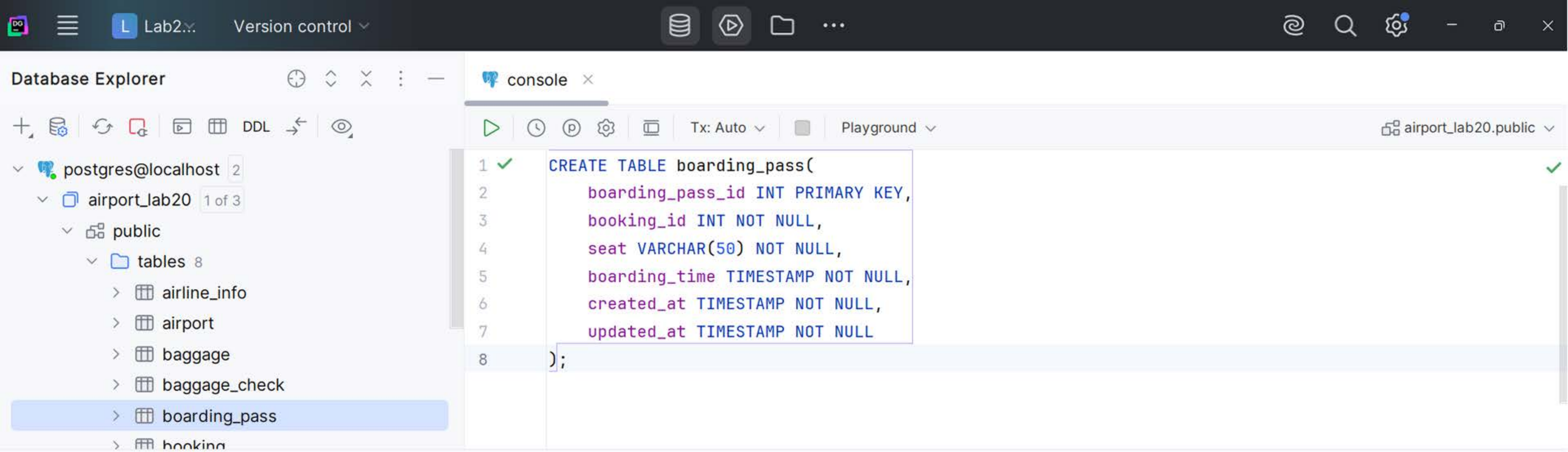


DDL

1. Create following tables with corresponding attributes:

Baggage_check: baggage_check_id(int), check_result (varchar(50)), created_at(timestamp), updated_at(timestamp), booking_id(int), passenger_id(int);

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

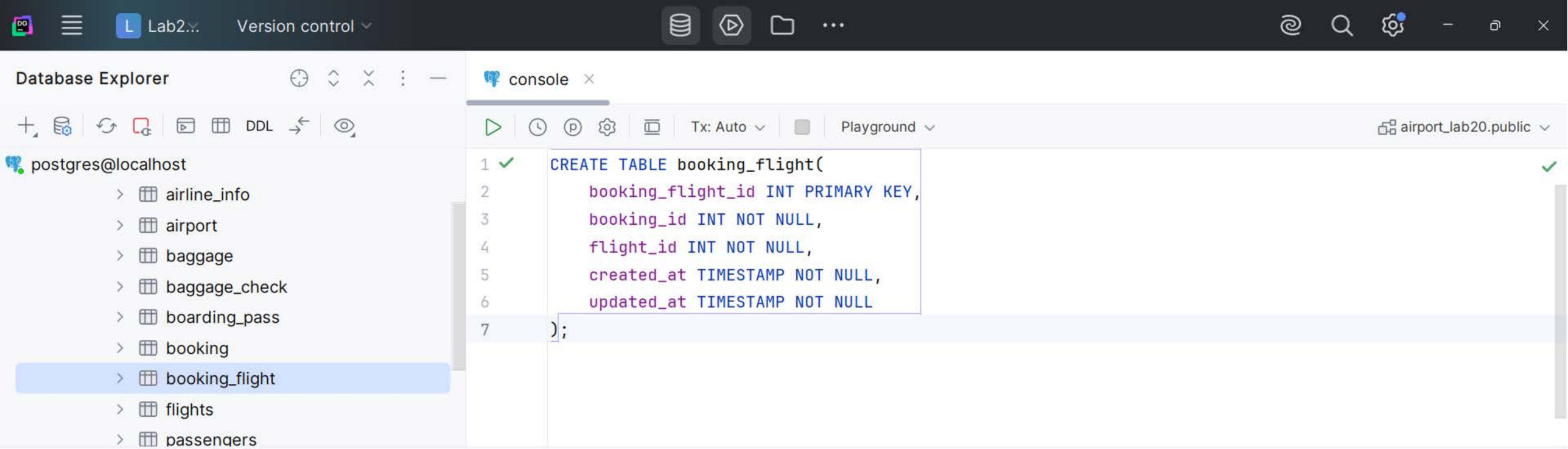


DDL

1. Create following tables with corresponding attributes:

Boarding_pass: boarding_pass_id(int), booking_id(int), seat (varchar(50)), boarding_time(timestamp), created_at(timestamp), updated_at(timestamp);

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

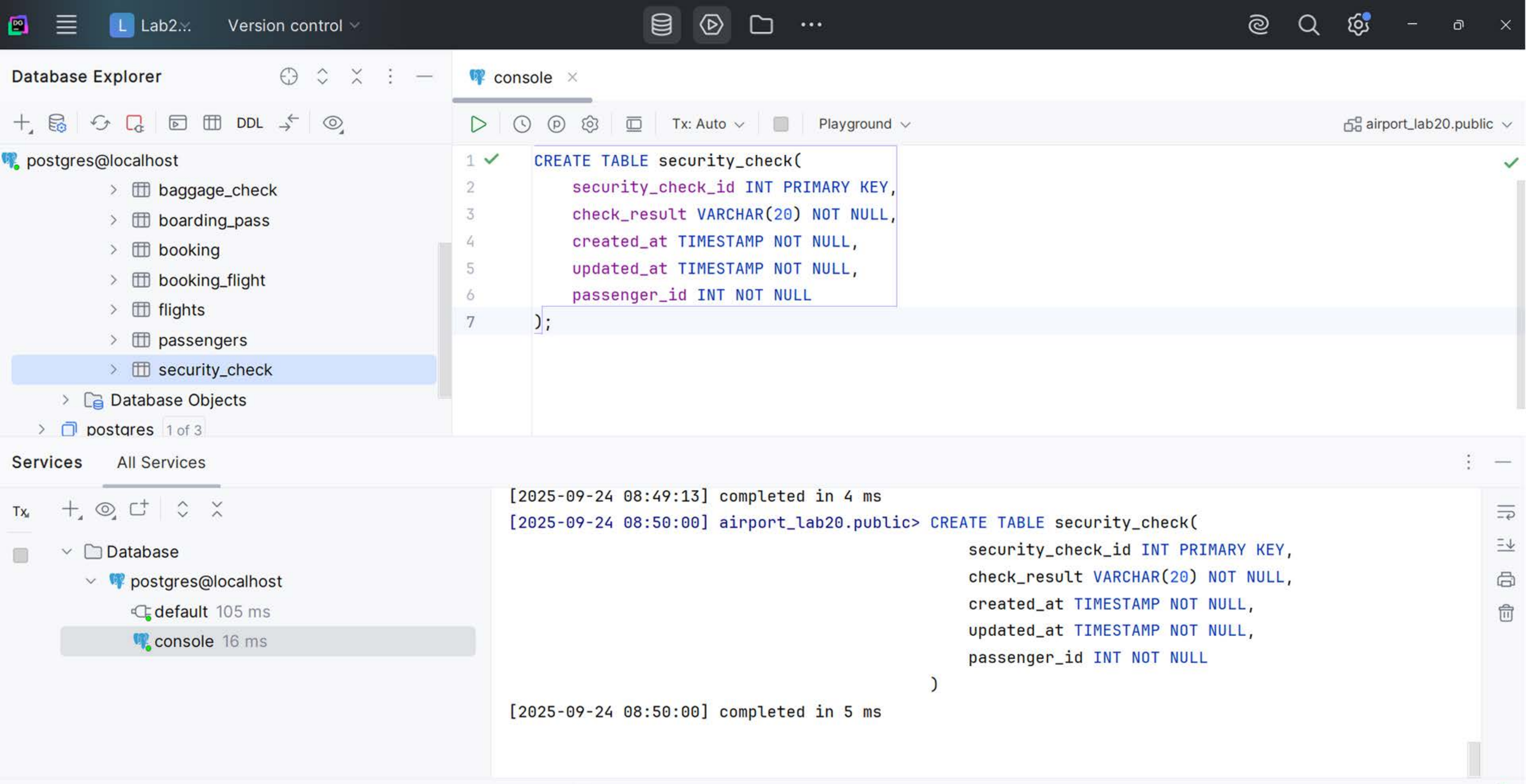


DDL

1. Create following tables with corresponding attributes:

Booking_flight: booking_flight_id(int), booking_id(int), flight_id(int), created_at(timestamp), updated_at(timestamp);

On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.

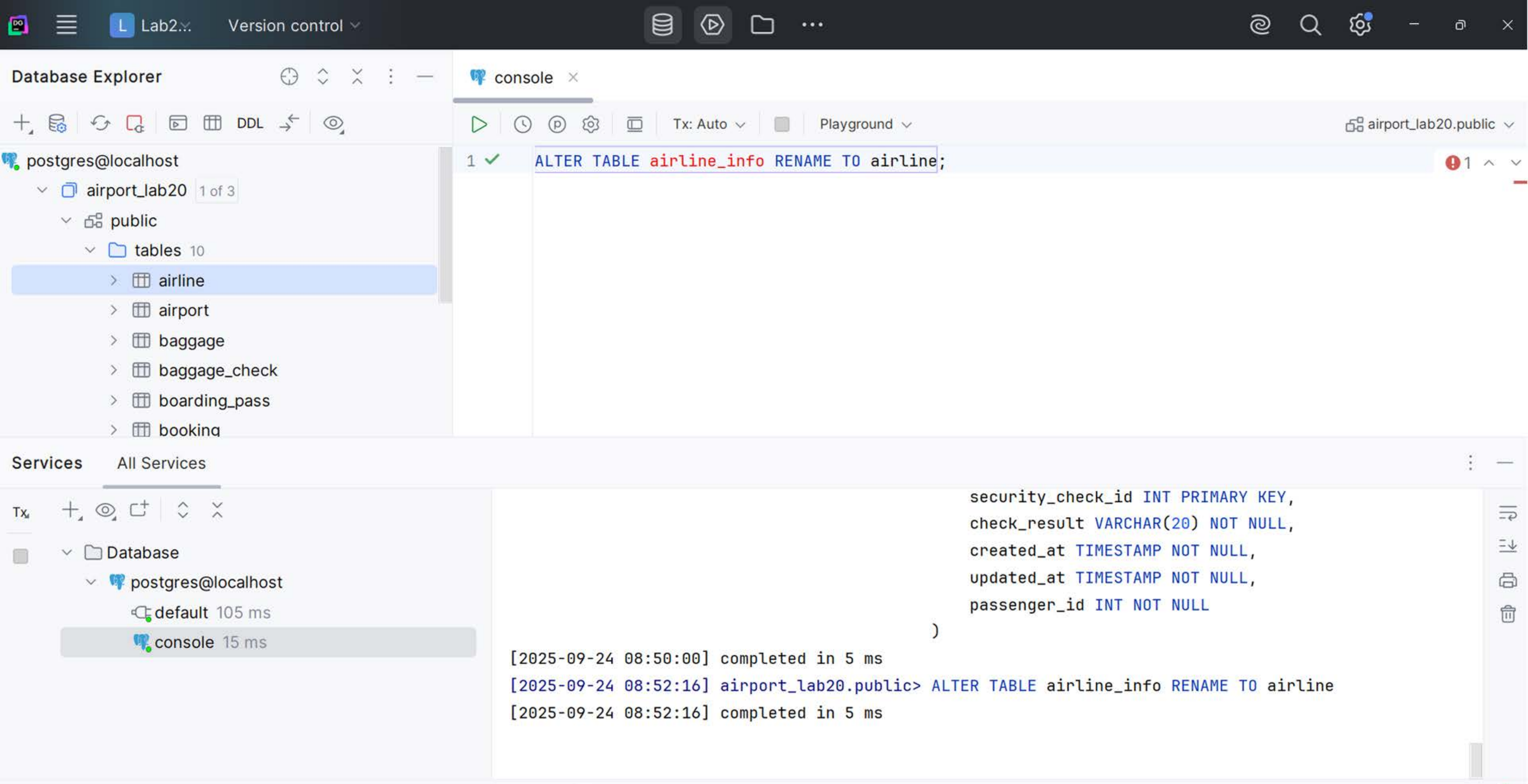


DDL

1. Create following tables with corresponding attributes:

Security_check: security_check_id(int), check_result(varchar(20)), created_at(timestamp), updated_at(timestamp), passenger_id(int);

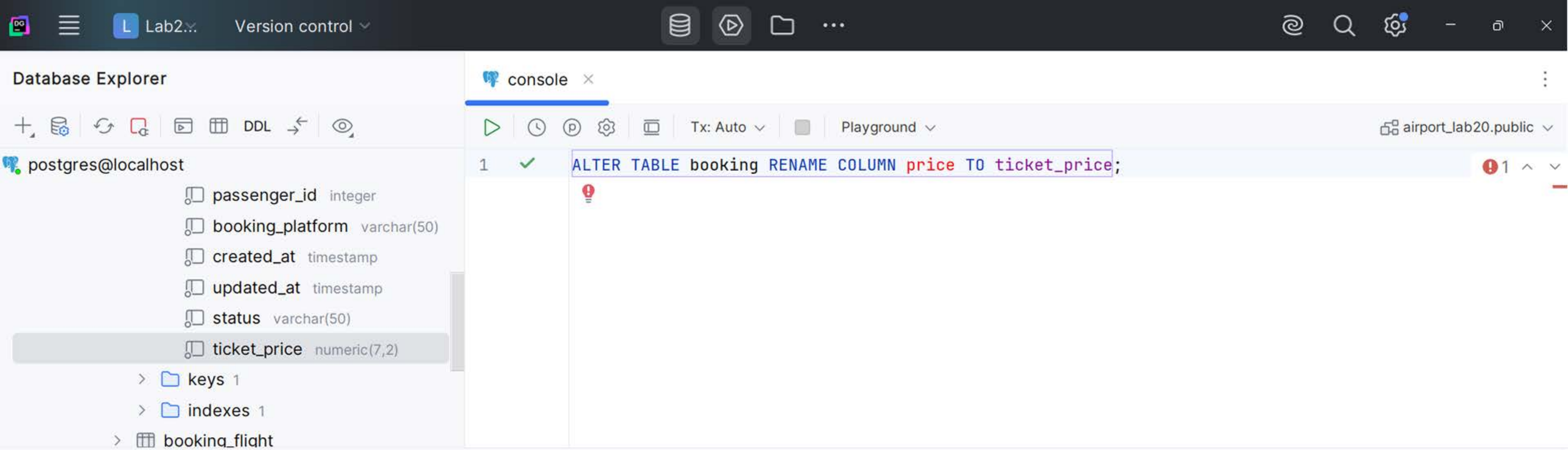
On this step I created all required tables such as airline_info, airport, passengers, flights, booking and others. For each table I defined primary keys and set all attributes with the NOT NULL constraint.



DDL

5. Rename airline_info table to airline;

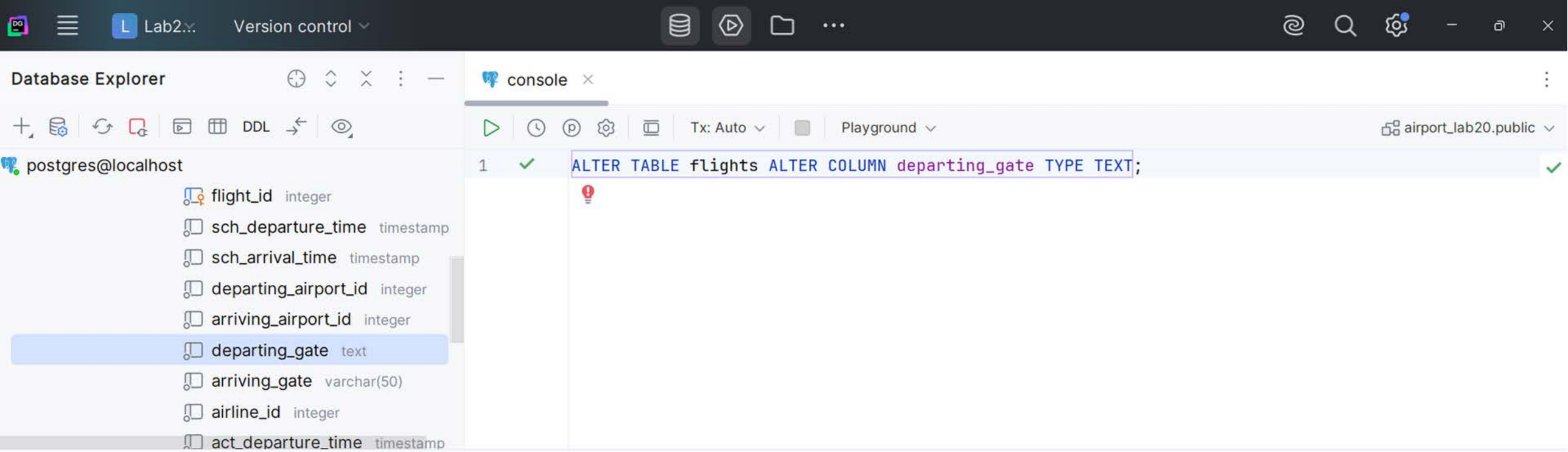
After creating the schema I applied modifications. I renamed airline_info to airline, changed column price to ticket_price in booking, altered departing_gate from varchar(50) to text in flights, and dropped the info column from airline.



DDL

6. Rename column price to ticket_price in booking table;

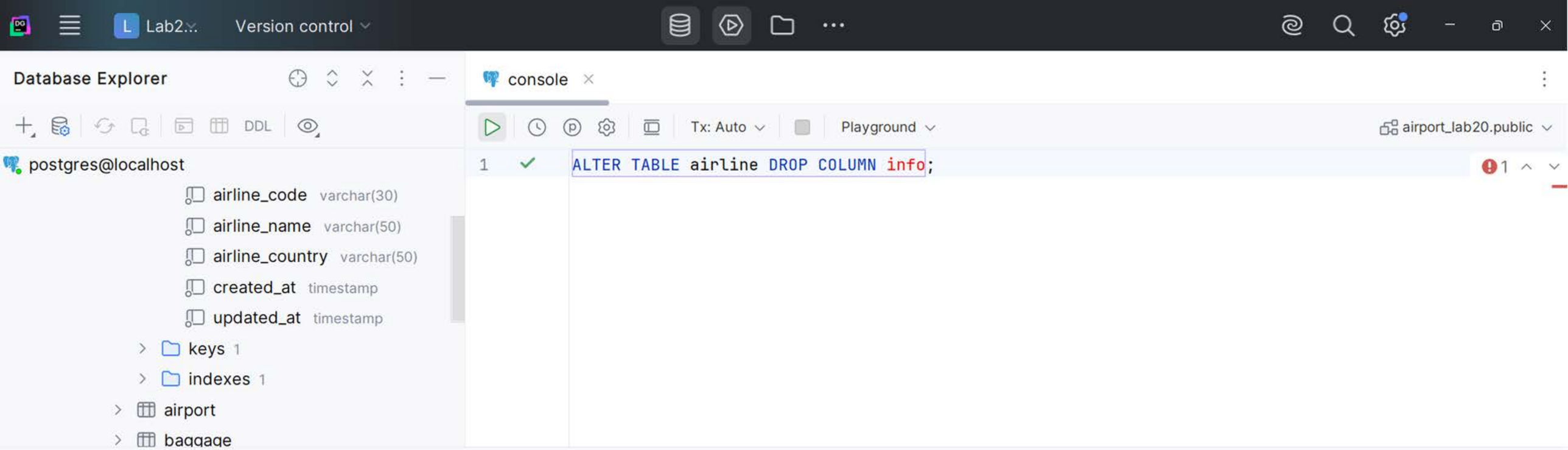
After creating the schema I applied modifications. I renamed airline_info to airline, changed column price to ticket_price in booking, altered departing_gate from varchar(50) to text in flights, and dropped the info column from airline.



DDL

7. Change data type of departing_gate from varchar(50) to text;

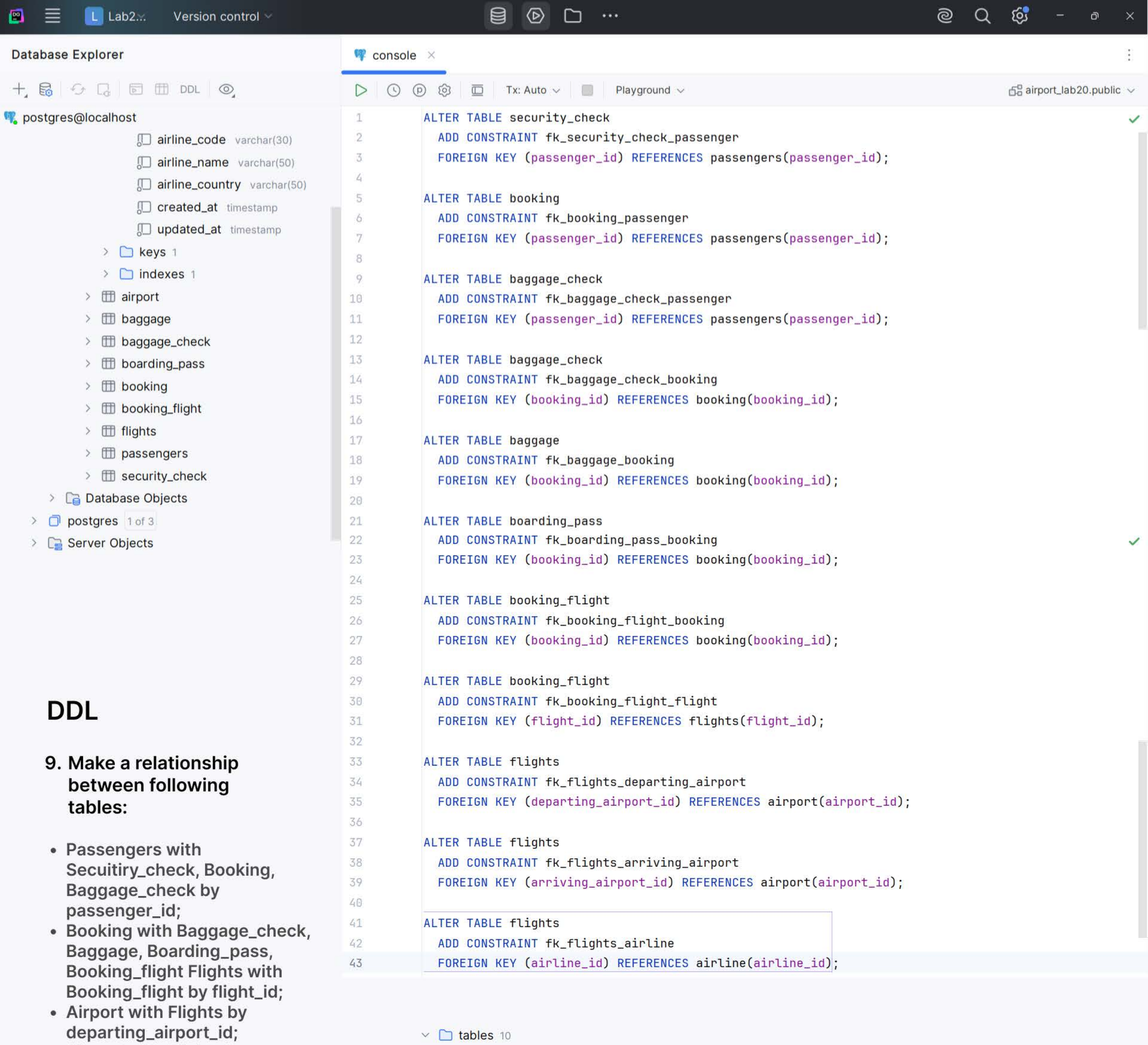
After creating the schema I applied modifications. I renamed airline_info to airline, changed column price to ticket_price in booking, altered departing_gate from varchar(50) to text in flights, and dropped the info column from airline.



DDL

7. Drop the column info(varchar(50)) from the airline table.

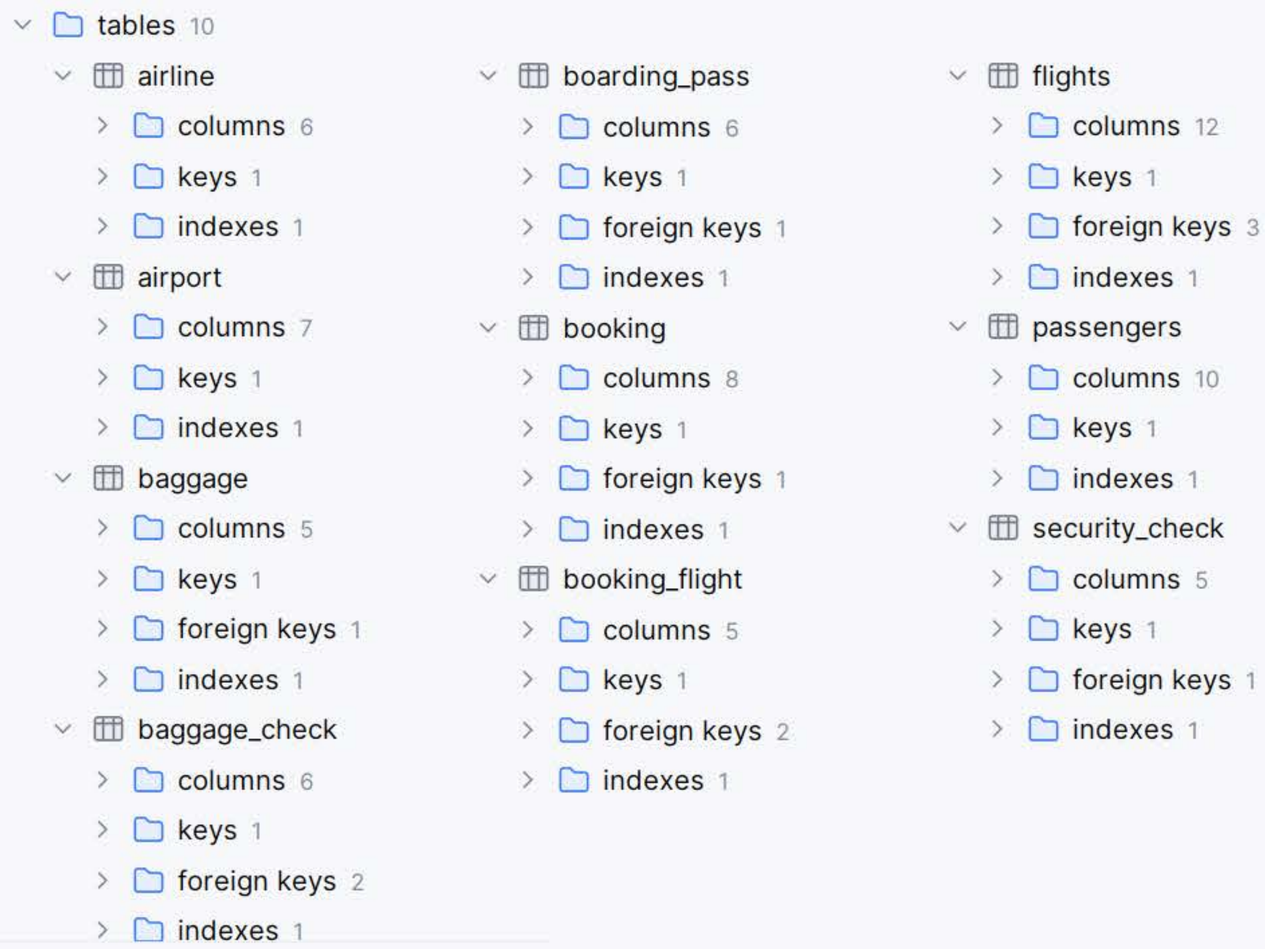
After creating the schema I applied modifications. I renamed airline_info to airline, changed column price to ticket_price in booking, altered departing_gate from varchar(50) to text in flights, and dropped the info column from airline.



DDL

9. Make a relationship between following tables:

- **Passengers with Secuitiry_check, Booking, Baggage_check by passenger_id;**
- **Booking with Baggage_check, Baggage, Boarding_pass, Booking_flight Flights with Booking_flight by flight_id;**
- **Airport with Flights by departing_airport_id;**
- **Airport with Flights by arriving_airport_id;**
- **Airline with Flights by airline_id;**



Next I established relationships between tables using foreign keys: passengers with security_check, booking, and baggage_check; booking with baggage, baggage_check, boarding_pass and booking_flight; flights with booking_flight; and airline and airport with flights.

Preview

```
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Carmen De Patagones Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-09-23', '2025-0
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Portoroz Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-05-14', '2025-02-25');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Deputado Luiz Eduardo Magalhães International Airport', 'Turkey', 'Manisa', 'Yenil
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('San Nicolas Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-04-01', '2025-05-19');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Fonte Boa Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-24', '2025-02-22');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Greater Binghamton/Edwin A Link field', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-06
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Atatürk International Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-10-22', '2024
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Senador Petrônio Portela Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-06-19', '2
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Columbia County Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-06-10', '2025-09-09
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Humbert River Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-28', '2024-09-25')
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Lusk Municipal Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-11-12', '2025-05-08'
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Bolling Air Force Base', 'Turkey', 'Manisa', 'Yeniköy', '2024-11-05', '2025-05-04'
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Woitape Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-01-05', '2025-09-23');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Ileg Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-10-18', '2025-01-07');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Chisasibi Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-02-14', '2024-09-25');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Lubang Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-10-23', '2025-09-04');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('L''Aquila-Preturo Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-05', '2024-11-
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Maintirano Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-06-28', '2025-04-04');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Kasos Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-05-06', '2025-09-17');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Piatã Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-10-20', '2025-07-12');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Sharjah International Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-12', '2025
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Malacca Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-07-17', '2025-03-30');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Helenvale Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-01-01', '2024-12-26');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Baie Comeau Airport', 'Turkey', 'Manisa', 'Yeniköy', '2025-06-01', '2025-05-17');
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Caribou Municipal Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-11-26', '2025-02-
insert into airport (airport_name, country, state, city, created_at, updated_at) values ('Rhinelander Oneida County Airport', 'Turkey', 'Manisa', 'Yeniköy', '2024-11-05', '
```

showing first 100 rows # Rows: 200 GENERATE DATA CLOSE

DML

1. **Generate and insert 200 random rows in your airport database.**

I generated and inserted 200 rows into the airport table. Then I added a new airline named KazAir in Kazakhstan, and later updated its country to Turkey. After that I inserted three airlines at once: AirEasy (France), FlyHigh (Brazil), and FlyFly (Poland).

Lab2 Version control

Database Explorer console

DDL Tx: Auto Playground airport_lab20.public

postgres@localhost 2
airport_lab20 1 of 3
postgres 1 of 3
Server Objects

```
1  SELECT count(*) FROM airport;
2  SELECT * FROM airport LIMIT 10;
3
```

Services All Services

Output count(*):bigint airport_lab20.public.airport

Tx: Auto DDL CSV

	airport_id	airport_name	country	state	city	created_at	upd
1	154	Tarempa Airport	Bolivia	<null>	Sotomayor	2025-07-04 00:00:00.000000	2024-
2	155	Unalakleet Airport	Egypt	<null>	Sumusţā as Sulţānī	2025-04-07 00:00:00.000000	2024-
3	156	Achutupo Airport	Russia	<null>	Krasnogvardeyets	2025-02-18 00:00:00.000000	2025-
4	157	Chistochina Airport	Czech Republic	<null>	Nová Role	2024-12-18 00:00:00.000000	2024-
5	158	Qaanaaq Airport	China	<null>	Matou	2025-04-01 00:00:00.000000	2025-
6	159	Blackall Airport	China	<null>	Pingpu	2025-06-22 00:00:00.000000	2025-
7	160	Kilkenny Airport	Vietnam		Phú Lộc	2025-01-31 00:00:00.000000	2024-
8	161	Majeed Bin Abdulaziz Airport	Portugal	São Miguel	Mosteiros	2025-02-11 00:00:00.000000	2025-

10 rows

Database Consoles > postgres@localhost > console 3:1 CRLF UTF-8 4 spaces

Lab2 Version control

Database Explorer console

DDL Tx: Auto Playground airport_lab20.public

postgres@localhost 2
airport_lab20 1 of 3
postgres 1 of 3
Server Objects

```
1  SELECT count(*) FROM airport;
2  SELECT * FROM airport LIMIT 10;
3
```

Services All Services

Output count(*):bigint airport_lab20.public.airport

SELECT count(*) FROM airport

CSV

	count
1	200

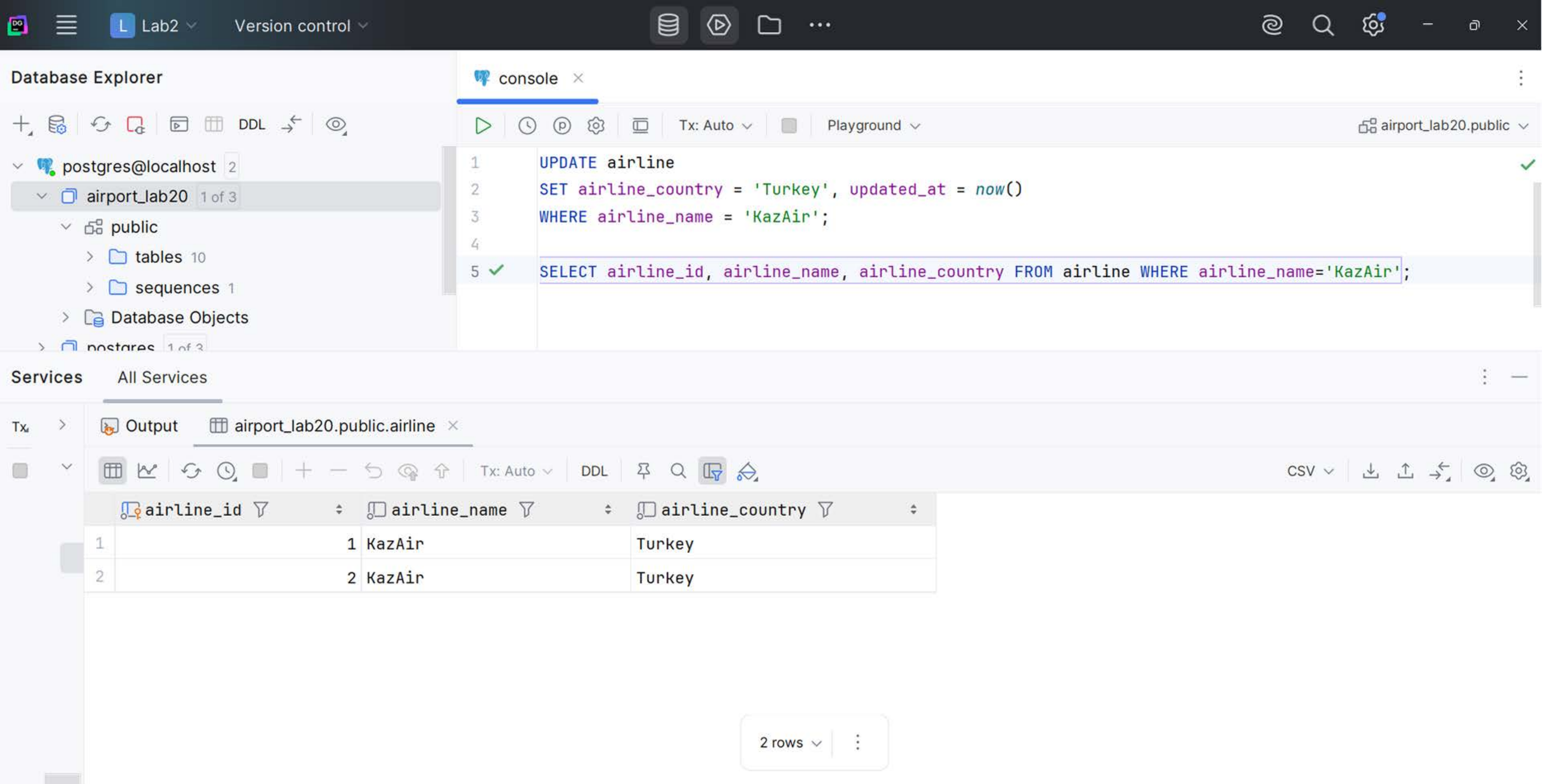
1 row

Database Consoles > postgres@localhost > console 3:1 CRLF UTF-8 4 spaces

DML

2. Add a new airline named "KazAir" based in "Kazakhstan" to the airline table.

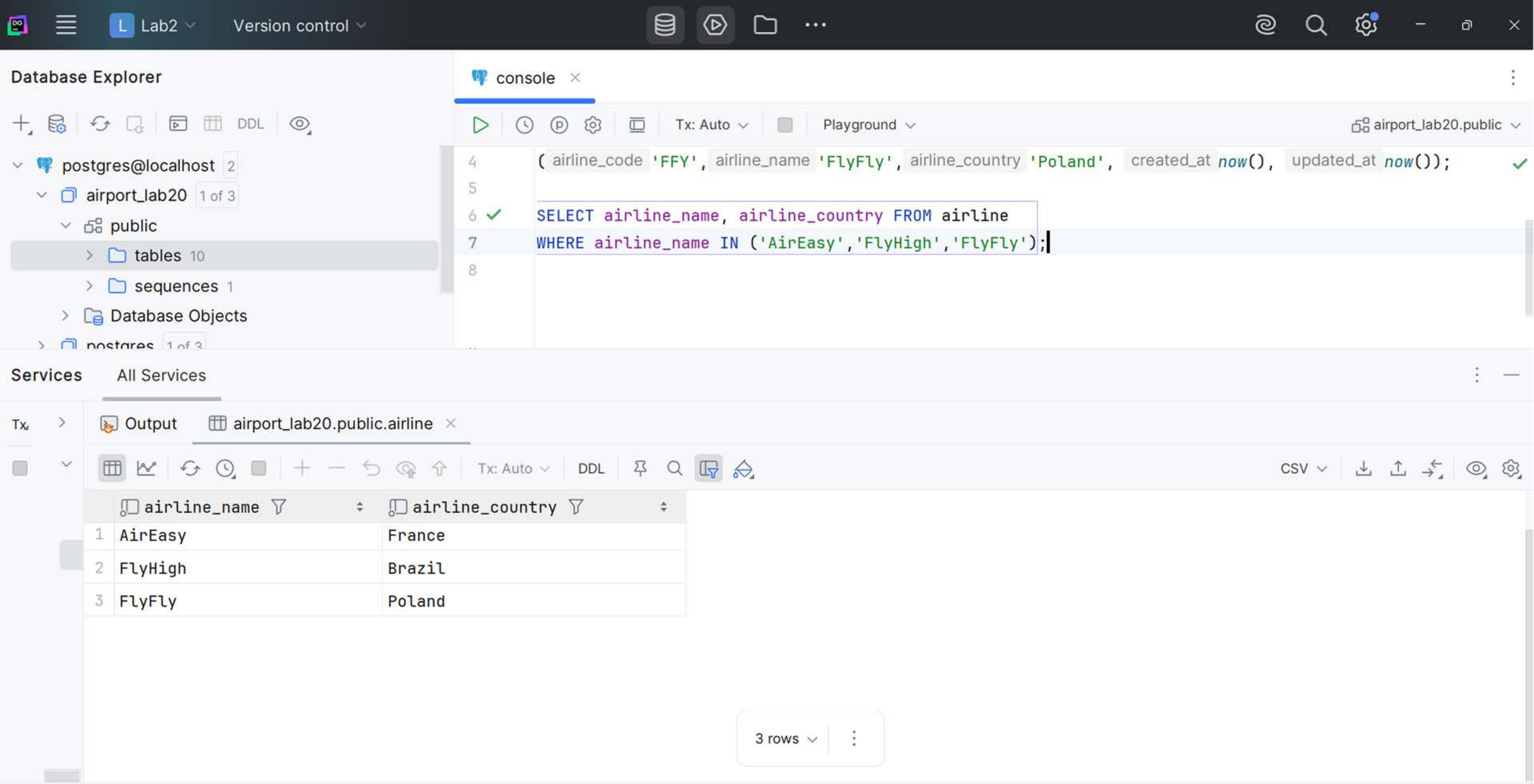
I generated and inserted 200 rows into the airport table. Then I added a new airline named KazAir in Kazakhstan, and later updated its country to Turkey. After that I inserted three airlines at once: AirEasy (France), FlyHigh (Brazil), and FlyFly (Poland).



DML

3. Update the airline country "KazAir" to "Turkey".

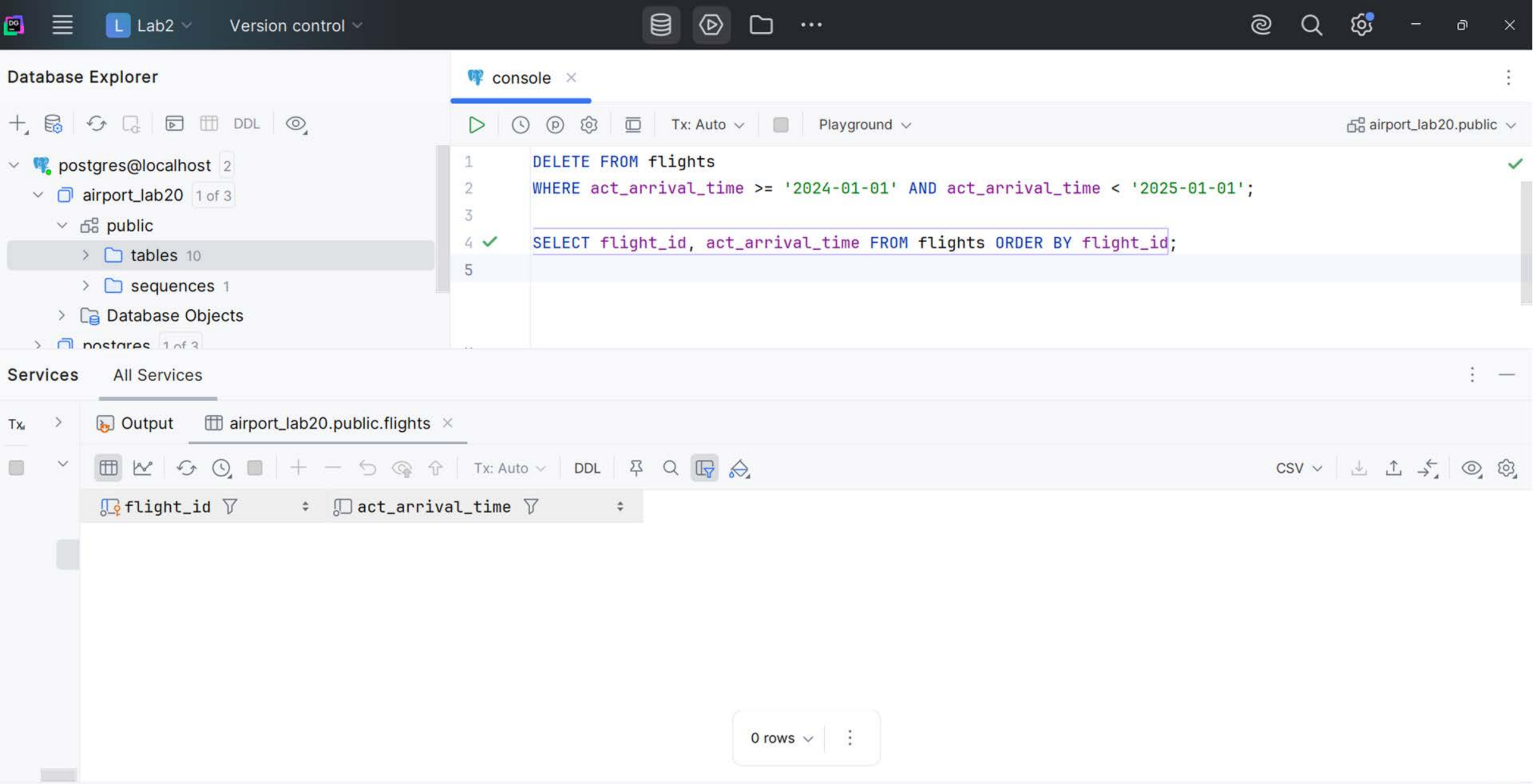
I generated and inserted 200 rows into the airport table. Then I added a new airline named KazAir in Kazakhstan, and later updated its country to Turkey. After that I inserted three airlines at once: AirEasy (France), FlyHigh (Brazil), and FlyFly (Poland).



DML

4. Add three airlines at once: "AirEasy" in "France", "FlyHigh" in "Brazil" and "FlyFly" in "Poland".

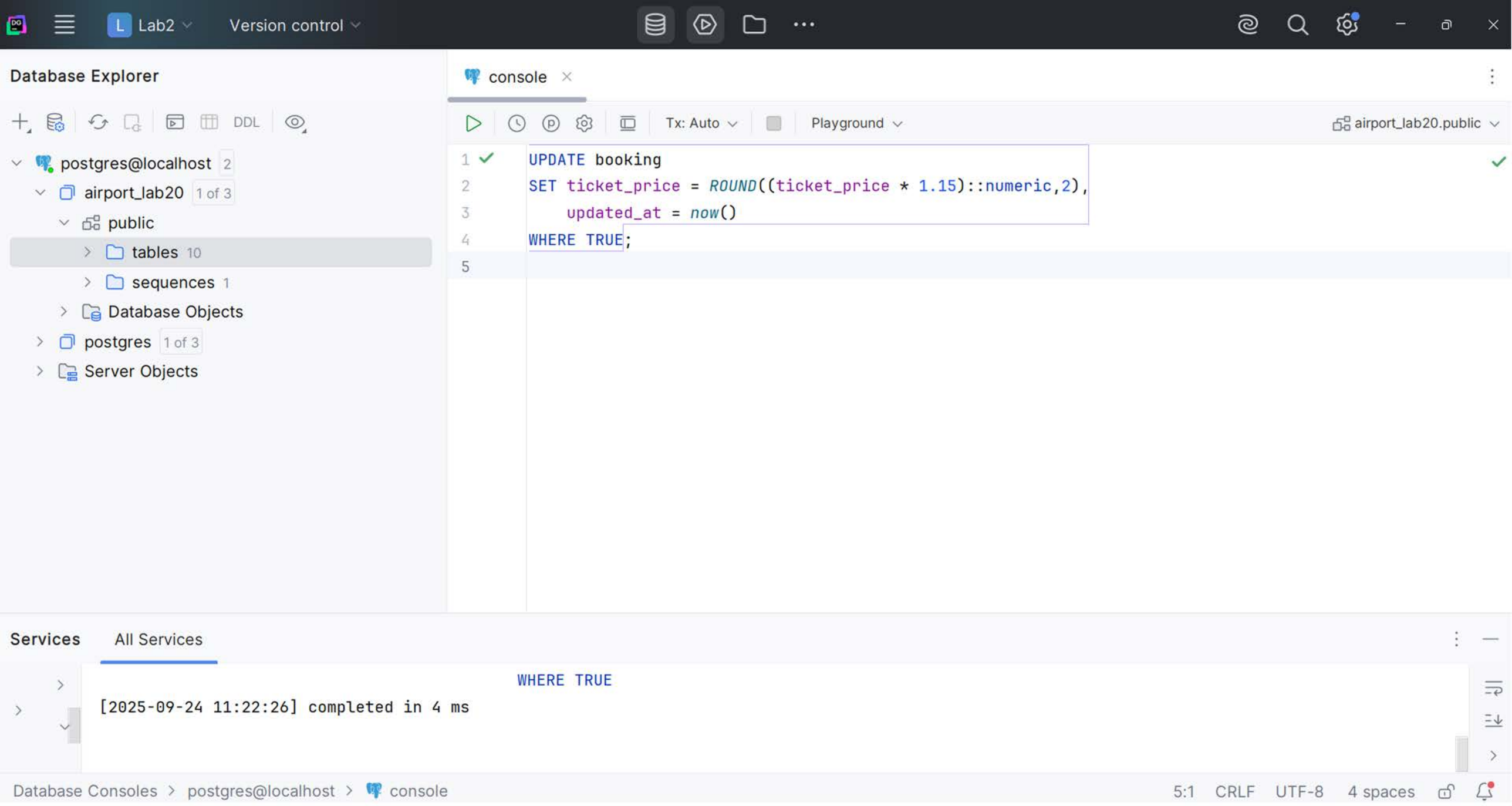
I generated and inserted 200 rows into the airport table. Then I added a new airline named KazAir in Kazakhstan, and later updated its country to Turkey. After that I inserted three airlines at once: AirEasy (France), FlyHigh (Brazil), and FlyFly (Poland).



DML

5. Delete all flights whose arrival in 2024 year.

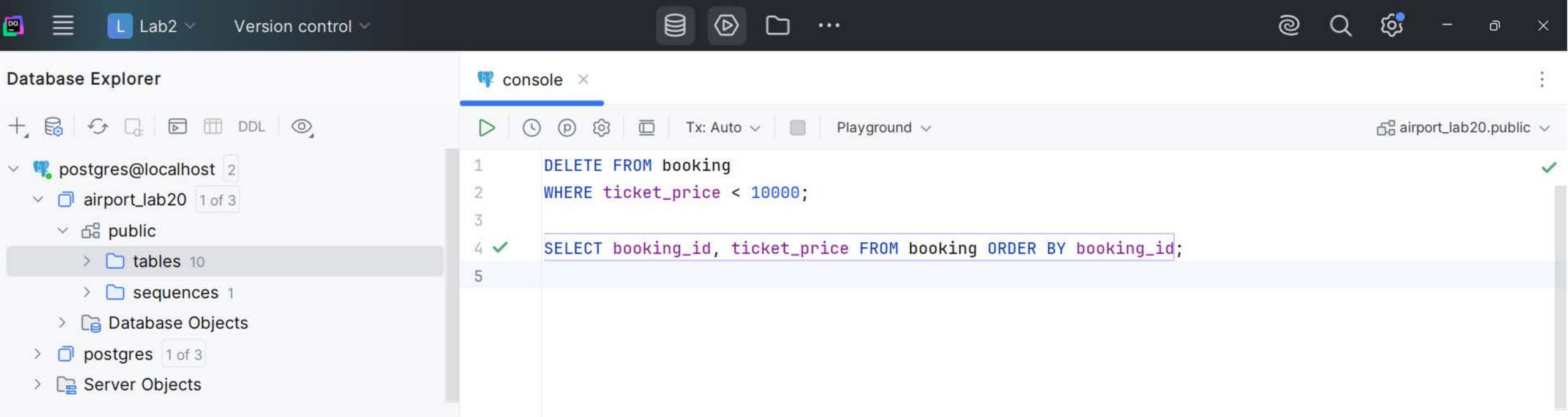
On this step I removed all flights that had their actual arrival time in the year 2024.
I used a DELETE statement with a date condition to filter the rows.
As a result, only the flights arriving outside of 2024 remained in the table.



DML

6. Increase the price of all tickets in booking table for flights by 15%.

On this step I updated the booking table to increase the price of all tickets by 15%.
I used an UPDATE statement that multiplied the value in the ticket_price column by 1.15.
As a result, every row in the booking table had its ticket price raised, while the update time column was also refreshed.



DML

7. Delete all tickets where price is less than 10000.

On this step I removed all records from the booking table where the ticket price was less than 10000.
I used a DELETE query with a condition on the ticket_price column.
As a result, only the tickets with higher prices remained in the table.