**DDL**

1. Create following tables with corresponding attributes:

CREATE TABLE airline\_info (

airline\_id INT NOT NULL PRIMARY KEY,

airline\_code VARCHAR(30) NOT NULL,

airline\_name VARCHAR(50) NOT NULL,

airline\_country VARCHAR(50) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

info VARCHAR(50) NOT NULL

);

CREATE TABLE airport (

airport\_id INT NOT NULL PRIMARY KEY,

airport\_name VARCHAR(50) NOT NULL,

country VARCHAR(50) NOT NULL,

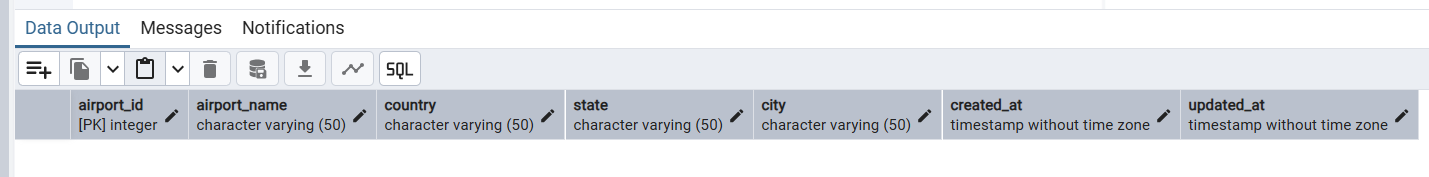
state VARCHAR(50) NOT NULL,

city VARCHAR(50) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL

);



CREATE TABLE passengers (

passenger\_id INT NOT NULL PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

date\_of\_birth DATE NOT NULL,

gender VARCHAR(50) NOT NULL,

country\_of\_citizenship VARCHAR(50) NOT NULL,

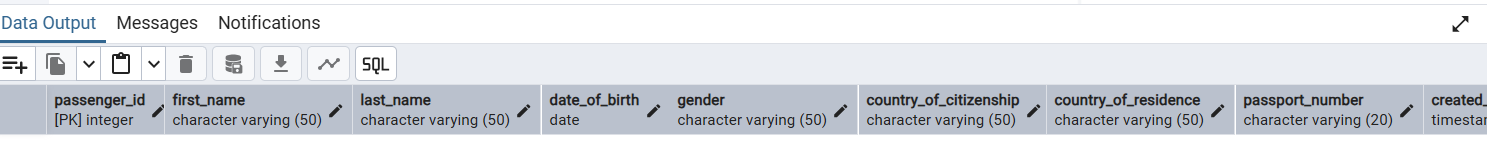
country\_of\_residence VARCHAR(50) NOT NULL,

passport\_number VARCHAR(20) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL

);



CREATE TABLE security\_check (

security\_check\_id INT NOT NULL PRIMARY KEY,

check\_result VARCHAR(20) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

passenger\_id INT NOT NULL,

CONSTRAINT fk\_security\_passenger FOREIGN KEY (passenger\_id) REFERENCES passengers (passenger\_id)

);



CREATE TABLE booking (

booking\_id INT NOT NULL PRIMARY KEY,

flight\_id INT NOT NULL,

passenger\_id INT NOT NULL,

booking\_platform VARCHAR(50) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

status VARCHAR(50) NOT NULL,

price DECIMAL(7,2) NOT NULL

);



CREATE TABLE baggage\_check (

baggage\_check\_id INT NOT NULL PRIMARY KEY,

check\_result VARCHAR(50) NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

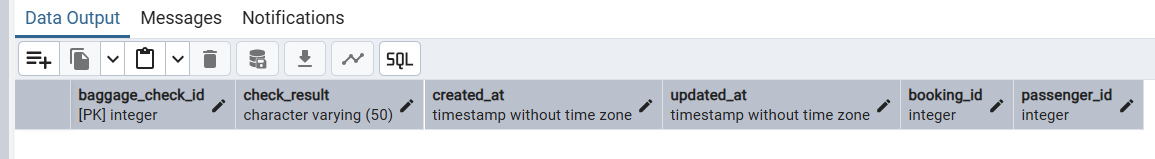
booking\_id INT NOT NULL,

passenger\_id INT NOT NULL,

CONSTRAINT fk\_baggagecheck\_booking FOREIGN KEY (booking\_id) REFERENCES booking (booking\_id),

CONSTRAINT fk\_baggagecheck\_passenger FOREIGN KEY (passenger\_id) REFERENCES passengers (passenger\_id)

);



CREATE TABLE baggage (

baggage\_id INT NOT NULL PRIMARY KEY,

weight\_in\_kg DECIMAL(4,2) NOT NULL,

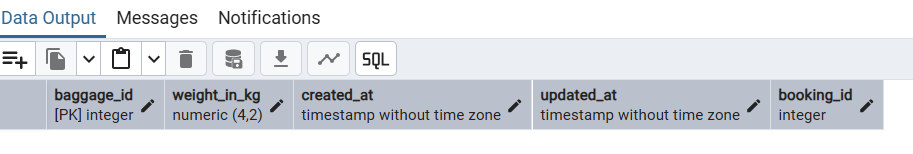
created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

booking\_id INT NOT NULL,

CONSTRAINT fk\_baggage\_booking FOREIGN KEY (booking\_id) REFERENCES booking (booking\_id)

);



CREATE TABLE boarding\_pass (

boarding\_pass\_id INT NOT NULL PRIMARY KEY,

booking\_id INT NOT NULL,

seat VARCHAR(50) NOT NULL,

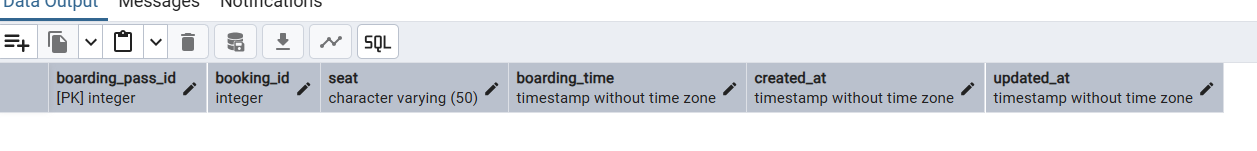
boarding\_time TIMESTAMP NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

CONSTRAINT fk\_boardingpass\_booking FOREIGN KEY (booking\_id) REFERENCES booking (booking\_id)

);



CREATE TABLE flights (

flight\_id INT NOT NULL PRIMARY KEY,

sch\_departure\_time TIMESTAMP NOT NULL,

sch\_arrival\_time TIMESTAMP NOT NULL,

departing\_airport\_id INT NOT NULL,

arriving\_airport\_id INT NOT NULL,

departing\_gate VARCHAR(50) NOT NULL, -- изменим на TEXT ниже, как в требовании

arriving\_gate VARCHAR(50) NOT NULL,

airline\_id INT NOT NULL,

act\_departure\_time TIMESTAMP NOT NULL,

act\_arrival\_time TIMESTAMP NOT NULL,

created\_at TIMESTAMP NOT NULL,

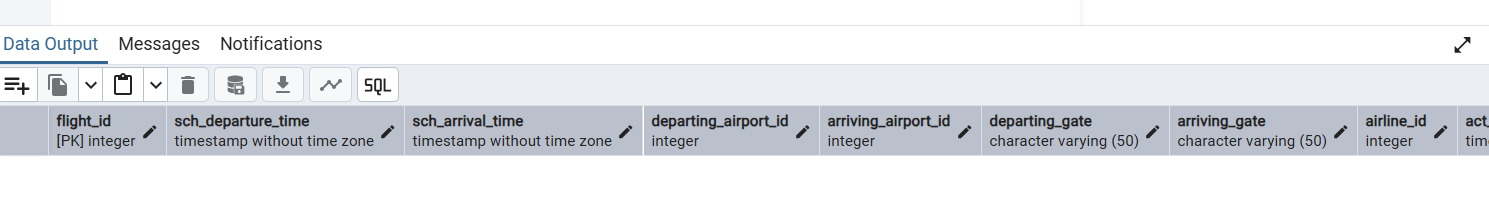
updated\_at TIMESTAMP NOT NULL,

CONSTRAINT fk\_flights\_departure\_airport FOREIGN KEY (departing\_airport\_id) REFERENCES airport (airport\_id),

CONSTRAINT fk\_flights\_arrival\_airport FOREIGN KEY (arriving\_airport\_id) REFERENCES airport (airport\_id),

CONSTRAINT fk\_flights\_airline FOREIGN KEY (airline\_id) REFERENCES airline\_info (airline\_id)

);



CREATE TABLE booking\_flight (

booking\_flight\_id INT NOT NULL PRIMARY KEY,

booking\_id INT NOT NULL,

flight\_id INT NOT NULL,

created\_at TIMESTAMP NOT NULL,

updated\_at TIMESTAMP NOT NULL,

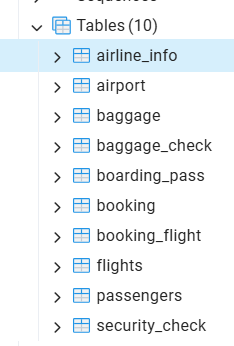
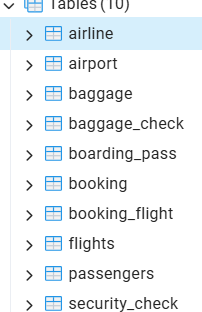
CONSTRAINT fk\_bookingflight\_booking FOREIGN KEY (booking\_id) REFERENCES booking (booking\_id),

CONSTRAINT fk\_bookingflight\_flight FOREIGN KEY (flight\_id) REFERENCES flights (flight\_id)

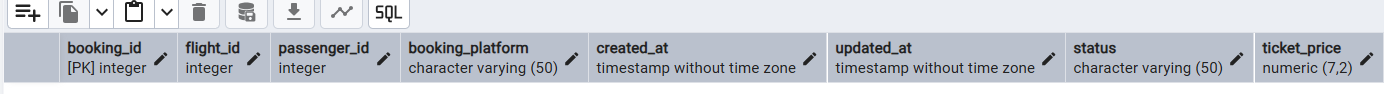
);



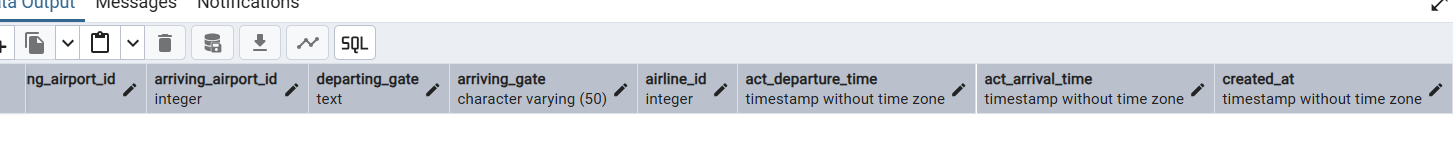
ALTER TABLE airline\_info RENAME TO airline;

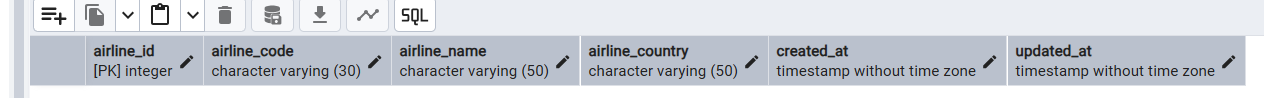
ALTER TABLE booking RENAME COLUMN price TO ticket\_price;



ALTER TABLE flights ALTER COLUMN departing\_gate TYPE TEXT;



ALTER TABLE airline DROP COLUMN info;



9 task created with tables

**DML**

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

BEGIN;

-- 1) airport (200 rows)

INSERT INTO airport (airport\_id, airport\_name, country, state, city, created\_at, updated\_at)

SELECT g,

'Airport\_' || g,

'Country\_' || ((g % 30) + 1),

'State\_' || ((g % 50) + 1),

'City\_' || ((g % 100) + 1),

now() - ( (g % 30) || ' days')::interval,

now() - ( ((g+1) % 10) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 2) airline (200 rows) (у тебя таблица переименована из airline\_info -> airline)

INSERT INTO airline (airline\_id, airline\_code, airline\_name, airline\_country, created\_at, updated\_at)

SELECT g,

'AL' || lpad((g % 999)::text,3,'0'),

'Airline\_' || g,

'Country\_' || ((g % 25) + 1),

now() - ( ((g\*2) % 40) || ' days')::interval,

now() - ( ((g\*3) % 5) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 3) passengers (200 rows)

INSERT INTO passengers (

passenger\_id, first\_name, last\_name, date\_of\_birth, gender,

country\_of\_citizenship, country\_of\_residence, passport\_number,

created\_at, updated\_at

)

SELECT g,

'FirstName\_' || (g % 100),

'LastName\_' || g,

(current\_date - ((18 + (g % 63)) || ' years')::interval)::date, -- возраст 18..80

CASE WHEN (g % 2)=0 THEN 'Female' ELSE 'Male' END,

'Country\_' || ((g % 25) + 1),

'Country\_' || (( (g+5) % 25) + 1),

upper(substr(md5(g::text || clock\_timestamp()::text),1,10)),

now() - ( ((g\*5) % 60) || ' days')::interval,

now() - ( ((g\*2) % 7) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 4) flights (200 rows)

-- departing\_airport\_id and arriving\_airport\_id chosen randomly 1..200 (may coincide sometimes)

INSERT INTO flights (

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

)

SELECT g,

now() + ((g % 7) || ' days')::interval + ((g % 24) || ' hours')::interval,

now() + ((g % 7) || ' days')::interval + (((g % 24)+2) || ' hours')::interval,

((random()\*199)::int + 1), -- random airport id 1..200

((random()\*199)::int + 1), -- random airport id 1..200

'Gate\_' || ((g % 50) + 1),

'Gate\_' || (((g+3) % 50) + 1),

((random()\*199)::int + 1), -- random airline 1..200

now() + (((g+1) % 7) || ' days')::interval + (((g+3) % 24) || ' hours')::interval,

now() + (((g+1) % 7) || ' days')::interval + (((g+5) % 24) || ' hours')::interval,

now() - ( ((g\*3) % 10) || ' days')::interval,

now() - ( ((g\*2) % 3) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 5) booking (200 rows)

-- booking.flight\_id set to a random flight (1..200) to keep plausible; passenger\_id random 1..200

INSERT INTO booking (

booking\_id, flight\_id, passenger\_id, booking\_platform,

created\_at, updated\_at, status, ticket\_price

)

SELECT g,

((random()\*199)::int + 1), -- random flight id

((random()\*199)::int + 1), -- random passenger id

CASE (g % 4)

WHEN 0 THEN 'Website'

WHEN 1 THEN 'MobileApp'

WHEN 2 THEN 'Agent'

ELSE 'Kiosk'

END,

now() - ((g % 40) || ' days')::interval,

now() - ((g % 10) || ' days')::interval,

CASE (g % 5)

WHEN 0 THEN 'CONFIRMED'

WHEN 1 THEN 'CANCELLED'

WHEN 2 THEN 'CHECKED\_IN'

WHEN 3 THEN 'PENDING'

ELSE 'COMPLETED'

END,

( (100 + (random()\*900))::numeric(7,2) ) -- price between 100 and 1000

FROM generate\_series(1,200) AS g;

-- 6) booking\_flight (200 rows)

-- booking\_id 1..200, flight\_id random 1..200

INSERT INTO booking\_flight (

booking\_flight\_id, booking\_id, flight\_id, created\_at, updated\_at

)

SELECT g,

g, -- map booking\_flight\_id -> booking\_id (ensures booking exists)

((random()\*199)::int + 1),

now() - (((g\*2) % 20) || ' days')::interval,

now() - (((g\*3) % 7) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 7) boarding\_pass (200 rows)

-- booking\_id random 1..200

INSERT INTO boarding\_pass (

boarding\_pass\_id, booking\_id, seat, boarding\_time, created\_at, updated\_at

)

SELECT g,

((random()\*199)::int + 1),

( (CHR(65 + (g % 6)) || (1 + (g % 30))::text) ), -- seats like A1..F30

now() + ((g % 3) || ' days')::interval + ((g % 12) || ' hours')::interval,

now() - (((g+2) % 12) || ' days')::interval,

now() - (((g+1) % 5) || ' days')::interval

FROM generate\_series(1,200) AS g;

-- 8) baggage (200 rows)

INSERT INTO baggage (

baggage\_id, weight\_in\_kg, created\_at, updated\_at, booking\_id

)

SELECT g,

( (5 + random()\*35)::numeric(4,2) ), -- weight 5.00 .. 40.00

now() - (((g\*4) % 30) || ' days')::interval,

now() - (((g\*2) % 6) || ' days')::interval,

((random()\*199)::int + 1)

FROM generate\_series(1,200) AS g;

-- 9) baggage\_check (200 rows)

INSERT INTO baggage\_check (

baggage\_check\_id, check\_result, created\_at, updated\_at, booking\_id, passenger\_id

)

SELECT g,

CASE (g % 3)

WHEN 0 THEN 'OK'

WHEN 1 THEN 'OVERWEIGHT'

ELSE 'NEEDS\_INSPECTION'

END,

now() - ((g % 15) || ' days')::interval,

now() - (((g+3) % 7) || ' hours')::interval,

((random()\*199)::int + 1),

((random()\*199)::int + 1)

FROM generate\_series(1,200) AS g;

-- 10) security\_check (200 rows)

INSERT INTO security\_check (

security\_check\_id, check\_result, created\_at, updated\_at, passenger\_id

)

SELECT g,

CASE (g % 4)

WHEN 0 THEN 'CLEARED'

WHEN 1 THEN 'FLAGGED'

WHEN 2 THEN 'ADDITIONAL\_SCREEN'

ELSE 'DENIED'

END,

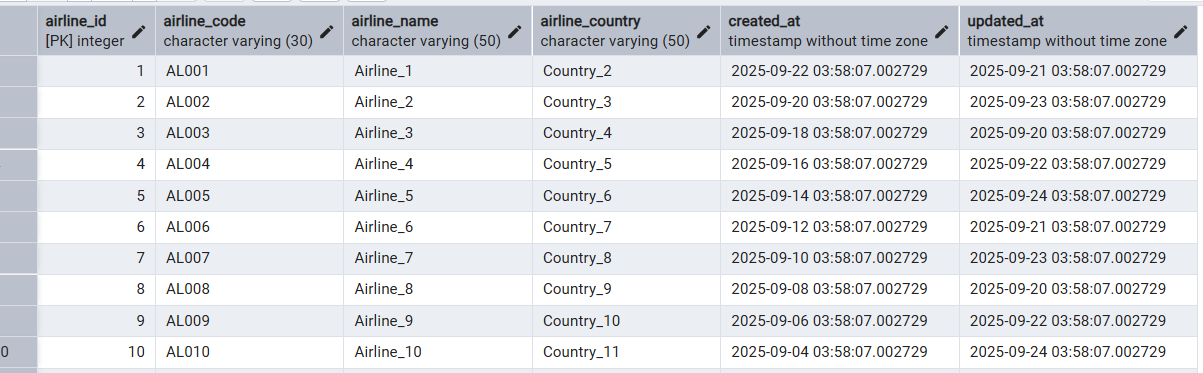
now() - (((g\*7) % 20) || ' days')::interval,

now() - (((g\*2) % 3) || ' hours')::interval,

((random()\*199)::int + 1)

FROM generate\_series(1,200) AS g;

COMMIT;



2 task

INSERT INTO airline (airline\_id, airline\_code, airline\_name, airline\_country, created\_at, updated\_at)

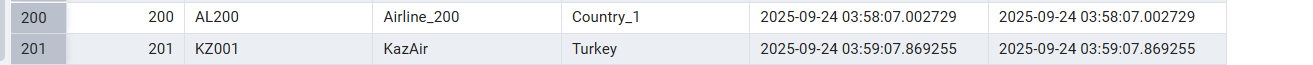
VALUES (201, 'KZ001', 'KazAir', 'Kazakhstan', NOW(), NOW());



3 task

update airline set airline\_country='Turkey' where airline\_name='KazAir';

SELECT \* FROM airline;



4 task

INSERT INTO airline (airline\_id, airline\_code, airline\_name, airline\_country, created\_at, updated\_at)

VALUES (202, 'CH012', 'CHINAIR', 'China', NOW(), NOW()),

(203, 'FR018', 'PARISair','France',NOW(),NOW()),

(204, 'USA111','LA','NY',NOW(),NOW());



INSERT INTO airline (airline\_id, airline\_code, airline\_name, airline\_country, created\_at, updated\_at)

VALUES

(205, 'FR001', 'AirEasy', 'France', NOW(), NOW()),

(206, 'BR001', 'FlyHigh', 'Brazil', NOW(), NOW()),

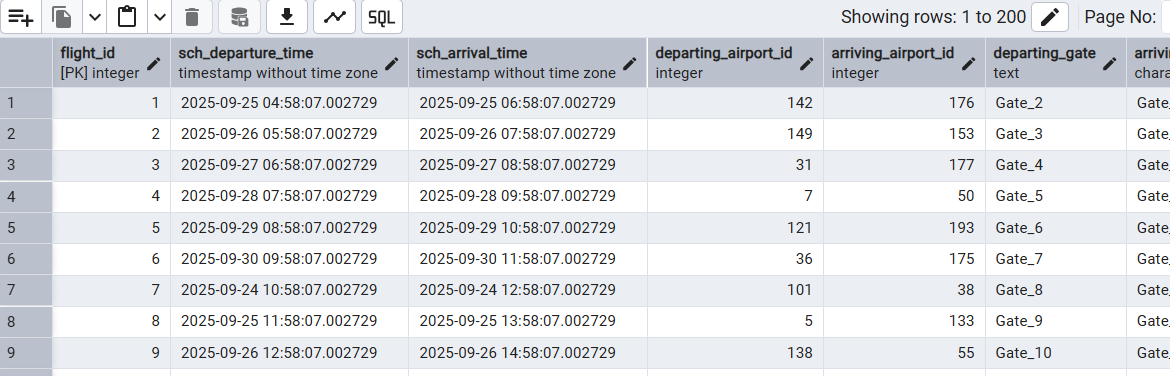
(207, 'PL001', 'FlyFly', 'Poland', NOW(), NOW());

select \* from airline;



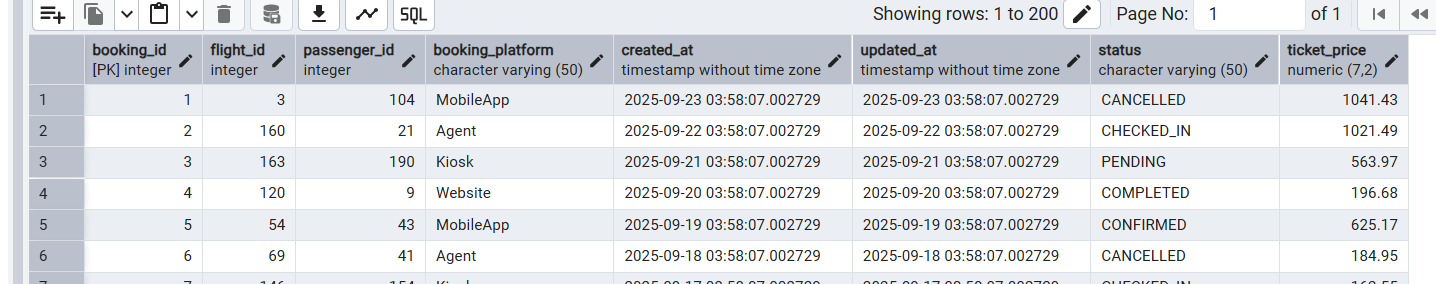
5 task

delete from flights where sch\_arrival\_time between '2024-01-01' and '2024-12-31' ;



6 task

update booking set ticket\_price = ticket\_price \* 1.15;



7 task

DELETE FROM baggage WHERE booking\_id IN (SELECT booking\_id FROM booking WHERE ticket\_price < 10000);

DELETE FROM boarding\_pass WHERE booking\_id IN (SELECT booking\_id FROM booking WHERE ticket\_price < 10000);

DELETE FROM baggage\_check WHERE booking\_id IN (SELECT booking\_id FROM booking WHERE ticket\_price < 10000);

DELETE FROM booking\_flight WHERE booking\_id IN (SELECT booking\_id FROM booking WHERE ticket\_price < 10000);

DELETE FROM booking WHERE ticket\_price < 10000;

select \* from booking ;

