

# DDL

-- 1. Task

```
CREATE TABLE airline_info (  
    airline_id INT GENERATED ALWAYS AS IDENTITY 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 DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW(),  
    info VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE airport (  
    airport_id INT GENERATED ALWAYS AS IDENTITY 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 DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()  
);  
  
CREATE TABLE passengers (  
    passenger_id INT GENERATED ALWAYS AS IDENTITY 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 DEFAULT NOW(),  
    updated_at TIMESTAMP NOT NULL DEFAULT NOW()
```

);

CREATE TABLE flights (

flight\_id INT GENERATED ALWAYS AS IDENTITY 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,

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 DEFAULT NOW(),

updated\_at TIMESTAMP NOT NULL DEFAULT NOW()

);

CREATE TABLE booking (

booking\_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

flight\_id INT NOT NULL,

passenger\_id INT NOT NULL,

booking\_platform VARCHAR(50) NOT NULL,

created\_at TIMESTAMP NOT NULL DEFAULT NOW(),

updated\_at TIMESTAMP NOT NULL DEFAULT NOW(),

status VARCHAR(50) NOT NULL,

price DECIMAL(7,2) NOT NULL

);

CREATE TABLE booking\_flight (

booking\_flight\_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

booking\_id INT NOT NULL,

flight\_id INT NOT NULL,

created\_at TIMESTAMP NOT NULL DEFAULT NOW(),

updated\_at TIMESTAMP NOT NULL DEFAULT NOW()

);

CREATE TABLE boarding\_pass (

```

boarding_pass_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
booking_id INT NOT NULL,
seat VARCHAR(50) NOT NULL,
boarding_time TIMESTAMP NOT NULL,
created_at TIMESTAMP NOT NULL DEFAULT NOW(),
updated_at TIMESTAMP NOT NULL DEFAULT NOW()
);

CREATE TABLE baggage (
    baggage_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
    weight_in_kg DECIMAL(4,2) NOT NULL,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW(),
    booking_id INT NOT NULL
);

CREATE TABLE baggage_check (
    baggage_check_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
    check_result VARCHAR(50) NOT NULL,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW(),
    booking_id INT NOT NULL,
    passenger_id INT NOT NULL
);

CREATE TABLE security_check (
    security_check_id INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,
    check_result VARCHAR(20) NOT NULL,
    created_at TIMESTAMP NOT NULL DEFAULT NOW(),
    updated_at TIMESTAMP NOT NULL DEFAULT NOW(),
    passenger_id INT NOT NULL
);

-- 2. done

-- 4. done

-- 5

ALTER TABLE airline_info RENAME TO airline;

```

-- 6

ALTER TABLE booking RENAME COLUMN price TO ticket\_price;

-- 7

ALTER TABLE flights ALTER COLUMN departing\_gate TYPE TEXT;

-- 8

ALTER TABLE airline DROP COLUMN info;

-- 9

ALTER TABLE booking

ADD CONSTRAINT fk\_booking\_passenger FOREIGN KEY (passenger\_id) REFERENCES  
passengers(passenger\_id) ON DELETE CASCADE,

ADD CONSTRAINT fk\_booking\_flight FOREIGN KEY (flight\_id) REFERENCES flights(flight\_id) ON DELETE  
CASCADE;

-- booking\_flight references booking and flights

ALTER TABLE booking\_flight

ADD CONSTRAINT fk\_booking\_flight\_booking FOREIGN KEY (booking\_id) REFERENCES  
booking(booking\_id) ON DELETE CASCADE,

ADD CONSTRAINT fk\_booking\_flight\_flights FOREIGN KEY (flight\_id) REFERENCES flights(flight\_id) ON  
DELETE CASCADE;

-- boarding\_pass references booking

ALTER TABLE boarding\_pass

ADD CONSTRAINT fk\_boarding\_pass\_booking FOREIGN KEY (booking\_id) REFERENCES  
booking(booking\_id) ON DELETE CASCADE;

-- baggage references booking

ALTER TABLE baggage

ADD CONSTRAINT fk\_baggage\_booking FOREIGN KEY (booking\_id) REFERENCES booking(booking\_id) ON  
DELETE CASCADE;

-- baggage\_check references booking and passengers

ALTER TABLE baggage\_check

ADD CONSTRAINT fk\_baggage\_check\_booking FOREIGN KEY (booking\_id) REFERENCES  
booking(booking\_id) ON DELETE CASCADE,

ADD CONSTRAINT fk\_baggage\_check\_passenger FOREIGN KEY (passenger\_id) REFERENCES  
passengers(passenger\_id) ON DELETE CASCADE;

-- security\_check references passengers

ALTER TABLE security\_check

```

    ADD CONSTRAINT fk_security_check_passenger FOREIGN KEY (passenger_id) REFERENCES
    passengers(passenger_id) ON DELETE CASCADE;

-- flights references airport (departing/arriving) and airline

ALTER TABLE flights

    ADD CONSTRAINT fk_flights_departing_airport FOREIGN KEY (departing_airport_id) REFERENCES
    airport(airport_id) ON DELETE RESTRICT,

    ADD CONSTRAINT fk_flights_arriving_airport FOREIGN KEY (arriving_airport_id) REFERENCES
    airport(airport_id) ON DELETE RESTRICT,

    ADD CONSTRAINT fk_flights_airline FOREIGN KEY (airline_id) REFERENCES airline(airline_id) ON DELETE
    RESTRICT;

```

## DML

```

-- 1. Insert random airports using WITH and generate_series

WITH countries AS (
    SELECT UNNEST(ARRAY[
        'Kazakhstan','Turkey','France','Brazil','Poland','USA','Germany','Spain','Italy','Canada',
        'UK','UAE','Qatar','Japan','Korea','India','China','Australia','Mexico','Netherlands'
    ]) AS country
),
states AS (
    SELECT UNNEST(ARRAY[
        'Almaty Region','Akmola','Ankara','Istanbul','Île-de-France','São Paulo','Mazowieckie','Lesser Poland',
        'California','Texas','Bavaria','Catalonia','Lombardy','Ontario','England','Dubai','Doha','Tokyo',
        'Seoul','Delhi','Beijing','NSW','Jalisco','Noord-Holland'
    ]) AS state
),
cities AS (
    SELECT UNNEST(ARRAY[
        'Almaty','Astana','Ankara','Istanbul','Paris','São Paulo','Warsaw','Kraków',
        'Los Angeles','San Francisco','Munich','Barcelona','Milan','Toronto','London',
        'Dubai','Doha','Tokyo','Seoul','New Delhi','Beijing','Sydney','Guadalajara','Amsterdam'
    ]) AS city

```

)

INSERT INTO airport (airport\_name, country, state, city)

SELECT

'Airport ' || gs::TEXT AS airport\_name,

(SELECT country FROM countries ORDER BY random() LIMIT 1) AS country,

(SELECT state FROM states ORDER BY random() LIMIT 1) AS state,

(SELECT city FROM cities ORDER BY random() LIMIT 1) AS city

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

-- 2. Insert airlines

INSERT INTO airline (airline\_code, airline\_name, airline\_country)

VALUES ('KZ','KazAir','Kazakhstan');

-- 3. Update airline country

UPDATE airline

SET airline\_country = 'Turkey',

updated\_at = NOW()

WHERE airline\_name = 'KazAir';

-- 4. Add multiple airlines

INSERT INTO airline (airline\_code, airline\_name, airline\_country)

VALUES

('AE','AirEasy','France'),

('FH','FlyHigh','Brazil'),

('FF','FlyFly','Poland');

-- 5. Delete flights with arrival times in 2024

DELETE FROM flights

WHERE EXTRACT(YEAR FROM act\_arrival\_time) = 2024

OR EXTRACT(YEAR FROM sch\_arrival\_time) = 2024;

-- 6. Update ticket prices (+15%)

UPDATE booking

```
SET ticket_price = ROUND(ticket_price * 1.15, 2),  
    updated_at = NOW();
```

```
-- 7. Delete bookings with ticket_price < 10000
```

```
DELETE FROM booking
```

```
WHERE ticket_price < 10000;
```