

```

--Airport DDL
CREATE SCHEMA IF NOT EXISTS airport;

CREATE TABLE IF NOT EXISTS airport.Booking_agents (
    Agent_id VARCHAR NOT NULL,
    Agent_name CHAR,
    Agent_details TEXT,
    PRIMARY KEY(Agent_id)
);

CREATE TABLE IF NOT EXISTS airport.Passengers (
    Passenger_id VARCHAR NOT NULL,
    First_name CHAR,
    Second_name CHAR,
    Last_name CHAR,
    Phone_number INTEGER,
    Email_address VARCHAR,
    Address_lines VARCHAR,
    City CHAR,
    State_province_county CHAR,
    Country CHAR,
    Other_passenger_details TEXT,
    PRIMARY KEY (Passenger_id)
);

CREATE TABLE IF NOT EXISTS airport.Itinerary_reservations (
    Reservation_id VARCHAR NOT NULL,
    Agent_id VARCHAR NOT NULL,
    Passenger_id VARCHAR NOT NULL,
    Reservation_status_code VARCHAR NOT NULL,
    Ticket_type_code VARCHAR NOT NULL,
    Travel_class_code VARCHAR NOT NULL,
    Date_reservation_made DATE,
    Number_in_party INTEGER,
    PRIMARY KEY (Reservation_id),
    FOREIGN KEY (Agent_id) REFERENCES airport.Booking_agents(Agent_id),
    FOREIGN KEY (Passenger_id) REFERENCES airport.Passengers(Passenger_id)
);

CREATE TABLE IF NOT EXISTS airport.Reservation_payments (
    Reservation_id VARCHAR NOT NULL,
    Payment_id VARCHAR NOT NULL,
    PRIMARY KEY (Reservation_id, Payment_id)
);

CREATE TABLE IF NOT EXISTS airport.Payments (
    Payment_id VARCHAR NOT NULL,
    Payment_status_code VARCHAR NOT NULL,
    Payment_amount INTEGER,
    PRIMARY KEY (Payment_id)
);

CREATE TABLE IF NOT EXISTS airport.Itinerary_legs (
    Reservation_id VARCHAR NOT NULL,
    Leg_id VARCHAR NOT NULL,
    PRIMARY KEY (Reservation_id, Leg_id)
);

```

```

CREATE TABLE IF NOT EXISTS airport.airports (
    airport_code VARCHAR NOT NULL,
    airport_name CHAR,
    airport_location VARCHAR,
    other_details TEXT,
    PRIMARY KEY (airport_code)
);

CREATE TABLE IF NOT EXISTS airport.Flight_schedules (
    Flight_number INT NOT NULL,
    Airline_code VARCHAR NOT NULL,
    Usual_aircraft_type_code VARCHAR NOT NULL,
    Origin_airport_code VARCHAR NOT NULL,
    Destination_airport_code VARCHAR NOT NULL,
    Departure_date_time TIMESTAMP,
    Arrival_date_time TIMESTAMP,
    PRIMARY KEY (Flight_number)
    FOREIGN KEY (Usual_aircraft_type_code) REFERENCES
airport.airports(Usual_aircraft_type_code),
    FOREIGN KEY (Origin_airport_code) REFERENCES airport.Legs(Origin_airport),
    FOREIGN KEY (Destination_airport_code) REFERENCES
airport.Legs(Destination_airport)
);

CREATE TABLE IF NOT EXISTS airport.Legs (
    Leg_id VARCHAR NOT NULL,
    Flight_number INT NOT NULL,
    Origin_airport VARCHAR,
    Destination_airport VARCHAR,
    Actual_departure_time TIME,
    Actual_arrival_time TIME,
    PRIMARY KEY (Leg_id),
    FOREIGN KEY (Flight_number) REFERENCES
airport.Flight_schedules(Flight_number)
);

CREATE TABLE IF NOT EXISTS airport.flight_costs (
    flight_number INT NOT NULL,
    aircraft_type_code VARCHAR NOT NULL,
    valid_from_date DATE NOT NULL,
    valid_to_date DATE NOT NULL,
    flight_cost INT,
    PRIMARY KEY (flight_number, aircraft_type_code, valid_from_date)
);

CREATE TABLE IF NOT EXISTS airport.ref_calendar (
    day_date DATE NOT NULL,
    day_number INT,
    business_day_yn INT,
    PRIMARY KEY (day_date)
);

--Query 1
SELECT *
FROM airport.Itinerary_reservations i
WHERE i.passenger_id = 'abc123'

--Query 2
SELECT *

```

```
FROM airport.Itinerary_reservations
where reservation_id IN
  (select leg_id
   from airport.itinerary_legs
   where leg_id = 1234)
```

```
--Query 3
SELECT fs.flight_number, fs.origin_airport_code, a.airport_code
FROM flight_schedules fs, airports a
WHERE a.airport_code = fs.origin_airport_code;
```

```
--Query 4
CREATE VIEW flight_schedule AS
SELECT *
FROM airport.flight_schedules
WHERE flight_number = 1234;
```

```
--Query 5
SELECT fs.flight_number cast(fs.departure_date_time AS departure_time)
[departure_time], cast(fs.arrival_date_time AS arrival_time) [arrival_time]
FROM airport.flight_schedules fs, airport.legs as l
WHERE fs.departure_time < l.actual_departure_time AND fs.arrival_time <
l.actual_arrival_time
GROUP BY fs.flight_number;
--cast removes the date and keeps the time
```

```
--Query 6
SELECT fs.flight_number, SUM(p.payment_amount) as total_sales
FROM airport.payments p
JOIN airport.reservation_payments rp
ON rp.payment_id = p.payment_id
JOIN airport.itinerary_reservations ir
ON rp.reservation_id = ir.reservation_id
JOIN airport.itinerary_legs il
ON il.reservation_id = ir.reservation_id
JOIN airport.legs l
ON l.leg_id = il.leg_id
JOIN airport.flight_costs fc
ON fc.flight_number = l.flight_number
WHERE fc.flight_number = 1234
GROUP BY fc.flight_number;
```