EJERCICIO 6 - AERÓDROMO

1. SENTENCIA DE CREACIÓN DE LA BASE DE DATOS AIRPORTS.

$ CREATE DATABASE AIRPORTS ;

```
postgres=# CREATE DATABASE airports ;
CREATE DATABASE
postgres=# \l
```

```
                                                List of databases
   Name    |  Owner   | Encoding | Locale Provider |  Collate   |   Ctype    | Locale | ICU Rule
s |   Access privileges
-----------+----------+----------+-----------------+------------+------------+--------+---------
--+-----------------------
 airports  | postgres | UTF8     | libc            | en_US.utf8 | en_US.utf8 |        |
```

2. SENTENCIAS DE CREACIÓN DE LAS TABLAS.

TABLA 1 - Aerolíneas (airlines)

$ CREATE TABLE Airlines(
airline_id VARCHAR (5) PRIMARY KEY,
airline_name VARCHAR (50) NOT NULL
);

```
postgres=# CREATE TABLE Airlines (
    airline_id VARCHAR(5) PRIMARY KEY,
    airline_name VARCHAR(50) NOT NULL
) ;
CREATE TABLE
postgres=# \dt
            List of tables
 Schema |   Name   | Type  |  Owner
--------+----------+-------+----------
 public | airlines | table | postgres
(1 row)
```

TABLA 2 – Aviones (Airplanes)

```
$ CREATE TABLE Airplanes (
airplane_id VARCHAR(5) PRIMARY KEY,
airplane_model VARCHAR(50) NOT NULL,
airplane_capacity INTEGER NOT NULL CHECK   (arplain_capacity >
0),
airplane_year INTEGER
);
```

```
postgres=# CREATE TABLE Airplanes (
    airplane_id VARCHAR(5) PRIMARY KEY,
    airplane_model VARCHAR(50) NOT NULL,
    airplane_capacity INTEGER NOT NULL CHECK (airplane_capacity > 0),
    airplane_year INTEGER
) ;
CREATE TABLE
postgres=# \dt
            List of tables
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | airlines  | table | postgres
 public | airplanes | table | postgres
(2 rows)

postgres=#
```

TABLA 3 – Aeropuertos (Airports)

```
$ CREATE TABLE Airports (
airport_id VARCHAR (5) PRIMARY KEY,
airport_name VARCHAR (100) NOT NULL,
country VARCHAR(50) NOT NULL
) ;
```

```
postgres=# CREATE TABLE Airports (
    airport_id VARCHAR(5) PRIMARY KEY,
    airport_name VARCHAR(100) NOT NULL,
    country VARCHAR(50) NOT NULL
) ;
CREATE TABLE
postgres=# \dt
            List of tables
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | airlines  | table | postgres
 public | airplanes | table | postgres
 public | airports  | table | postgres
(3 rows)

postgres=#
```

TABLA 4 – Pasajeros (Passengers)

```
$ CREATE TABLE Passengers (
passenger_id VARCHAR(5) PRIMARY KEY,
passenger_name VARCHAR(100) NOT NULL
) ;
```

```
postgres=# CREATE TABLE Passengers (
    passenger_id VARCHAR(5) PRIMARY KEY,
    passenger_name VARCHAR(100) NOT NULL
) ;
CREATE TABLE
postgres=# \dt
             List of tables
 Schema |    Name    | Type  |  Owner
--------+------------+-------+----------
 public | airlines   | table | postgres
 public | airplanes  | table | postgres
 public | airports   | table | postgres
 public | passengers | table | postgres
(4 rows)

postgres=#
```

TABLA 5 – Vuelos (Flights)

```
$ CREATE TABLE Flights (
      Flight_id VARCHAR(5) PRIMARY KEY,
      Departure_datetime    TIMESTAMP NOT NULL,
      Arrival_datetime TIMESTAMP NOT NULL,
      Origin_airport_id VARCHAR(5) NOT NULL,
      Destination_airport_id VARCHAR(5) NOT NULL,
      Airplane_id VARCHAR(5) NOT NULL,
      Airline_id VARCHAR(5) NOT NULL,
      FOREIGN KEY (origin_airport_id) REFERENCES
Airports(airport_id) ON UPDATE CASCADE ON DELETE RESTRICT,
      FOREIGN KEY (destination_airport_id) REFERENCES
Airports(airport_id) ON UPDATE CASCADE ON DELETE RESTRICT,
      FOREIGN KEY (airplane_id) REFERENCES
Airplanes(airplane_id) ON UPDATE CASCADE ON DELETE RESTRICT,
      FOREIGN KEY (airline_id) REFERENCES Airlines(airline_id)
ON UPDATE CASCADE ON DELETE RESTRICT,
CHECK (arrival_datetime > departure_datetime));
```

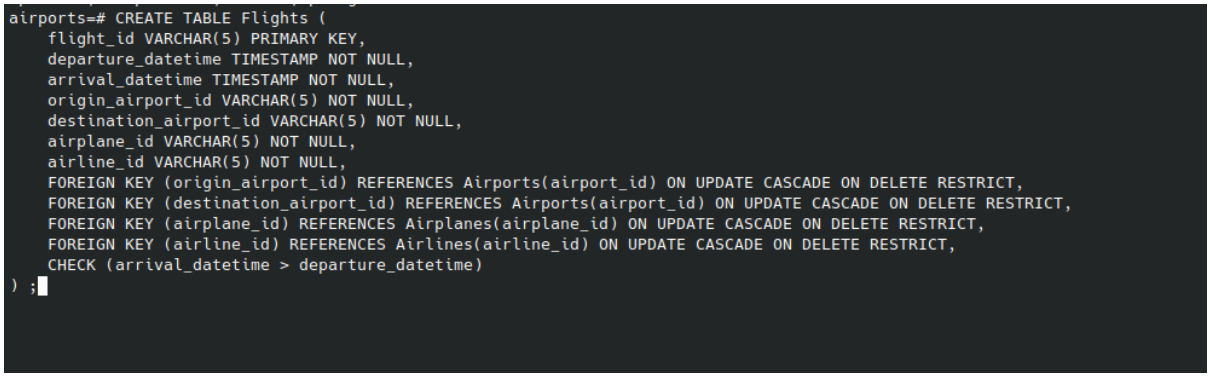

Tabla 6 – Reservas (Bookings)

```
CREATE TABLE Bookings ( flight_id VARCHAR(5) NOT NULL,
passenger_id VARCHAR(5) NOT NULL, seat VARCHAR(10),
PRIMARY KEY (flight_id, passenger_id),
FOREIGN KEY (flight_id) REFERENCES Flights(flight_id) ON UPDATE
CASCADE ON DELETE CASCADE,
FOREIGN KEY (passenger_id) REFERENCES Passengers(passenger_id)
ON UPDATE CASCADE ON DELETE RESTRICT
 );
```

```
airports=# CREATE TABLE Bookings (
    flight_id VARCHAR(5) NOT NULL,
    passenger_id VARCHAR(5) NOT NULL,
    seat VARCHAR(10),
    PRIMARY KEY (flight_id, passenger_id),
    FOREIGN KEY (flight_id) REFERENCES Flights(flight_id) ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (passenger_id) REFERENCES Passengers(passenger_id) ON UPDATE CASCADE ON DELETE RESTRICT
) ;
CREATE TABLE
airports=# \dt
            List of tables
 Schema |    Name    | Type  |  Owner
--------+------------+-------+----------
 public | airlines   | table | postgres
 public | airplanes  | table | postgres
 public | airports   | table | postgres
 public | bookings   | table | postgres
 public | flights    | table | postgres
 public | passengers | table | postgres
(6 rows)
```

3. SENTENCIAS DE INSERCIÓN DE LOS DATOS HASTA AHORA ALMACENADOS EN EXCEL TENIENDO EN CUENTA QUE LA INFORMACIÓN HA DE ESTAR SIEMPRE BIEN RELACIONADA.

INSERCION AEROLINEAS:

INSERT INTO Airlines (airline_id, airline_name) VALUES ('A001', 'Iberia'), ('A002', 'Air Europa');

```
airports=# INSERT INTO Airlines (airline_id, airline_name) VALUES
('A001', 'Iberia'),
('A002', 'Air Europa');
INSERT 0 2
airports=# select * from airlines ;
 airline_id | airline_name
------------+--------------
 A001       | Iberia
 A002       | Air Europa
(2 rows)

airports=#
```

INSERCION AVIONES

INSERT INTO Airplanes (airplane_id, airplane_model, airplane_capacity, airplane_year) VALUES ('P001', 'Airbus A320', 180, 2018), ('P002', 'Boeing 737', 150, 2015), ('P003', 'Airbus A330', 280, 2020);

```
airports=# INSERT INTO Airplanes (airplane_id, airplane_model, airplane_capacity, airplane_year) VALUES
('P001', 'Airbus A320', 180, 2018),
('P002', 'Boeing 737', 150, 2015),
('P003', 'Airbus A330', 280, 2020);
INSERT 0 3
```

```
airports=# select * from airplanes ;
 airplane_id | airplane_model | airplane_capacity | airplane_year
-------------+----------------+-------------------+---------------
 P001        | Airbus A320    |               180 |          2018
 P002        | Boeing 737     |               150 |          2015
 P003        | Airbus A330    |               280 |          2020
(3 rows)
```

INSERCION AEREOPUESTOS

INSERT INTO Airports (airport_id, airport_name, country) VALUES ('AP01', 'Adolfo Suárez Madrid-Barajas', 'España'), ('AP02', 'Aeropuerto de Barcelona-El Prat', 'España'), ('AP03', 'Charles de Gaulle', 'Francia'), ('AP04', 'Aeropuerto Internacional de Múnich', 'Alemania');

```
airports=# INSERT INTO Airports (airport_id, airport_name, country) VALUES
('AP01', 'Adolfo Suárez Madrid-Barajas', 'España'),
('AP02', 'Aeropuerto de Barcelona-El Prat', 'España'),
('AP03', 'Charles de Gaulle', 'Francia'),
('AP04', 'Aeropuerto Internacional de Múnich', 'Alemania');
INSERT 0 4
airports=# select * from air
airlines   airplanes  airports
airports=# select * from airports ;
 airport_id |            airport_name            | country
------------+------------------------------------+----------
 AP01       | Adolfo Suárez Madrid-Barajas       | España
 AP02       | Aeropuerto de Barcelona-El Prat    | España
 AP03       | Charles de Gaulle                  | Francia
 AP04       | Aeropuerto Internacional de Múnich | Alemania
(4 rows)

airports=#
```

INSERCION PASAJEROS

INSERT INTO Passengers (passenger_id, passenger_name) VALUES ('PAS01', 'Laura García'), ('PAS02', 'Mario Pérez'), ('PAS03', 'Sofía López'), ('PAS04', 'Javi Rivas');

```
airports=# INSERT INTO Passengers (passenger_id, passenger_name) VALUES
('PAS01', 'Laura García'),
('PAS02', 'Mario Pérez'),
('PAS03', 'Sofía López'),
('PAS04', 'Javi Rivas');
INSERT 0 4
airports=# select * from passengers ;
 passenger_id | passenger_name
--------------+----------------
 PAS01        | Laura García
 PAS02        | Mario Pérez
 PAS03        | Sofía López
 PAS04        | Javi Rivas
(4 rows)

airports=#
```

INSERT VUELOS

INSERT INTO Flights (flight_id, departure_datetime, arrival_datetime, origin_airport_id, destination_airport_id, airplane_id, airline_id) VALUES ('IB01', '2024-11-10 08:00:00', '2024-11-10 10:30:00', 'AP01', 'AP03', 'P001', 'A001'), -- Madrid a París (Iberia, A320) ('AE02', '2024-11-10 12:00:00', '2024-11-10 14:00:00', 'AP02', 'AP01', 'P002', 'A002'), -- Barcelona a Madrid (Air Europa, B737) ('IB03', '2024-11-11 16:00:00', '2024-11-11 18:45:00', 'AP01', 'AP04', 'P003', 'A001'), -- Madrid a Múnich (Iberia, A330) ('AE04', '2024-11-12 07:30:00', '2024-11-12 09:30:00', 'AP03', 'AP02', 'P001', 'A002'); -- París a Barcelona (Air Europa, A320)

```
airports=# INSERT INTO Flights (flight_id, departure_datetime, arrival_datetime, origin_airport_id, destination_airport_id, airplane_id, airline_id) VALUES
('IB01', '2024-11-10 08:00:00', '2024-11-10 10:30:00', 'AP01', 'AP03', 'P001', 'A001'), -- Madrid a París (Iberia, A320)
('AE02', '2024-11-10 12:00:00', '2024-11-10 14:00:00', 'AP02', 'AP01', 'P002', 'A002'), -- Barcelona a Madrid (Air Europa, B737)
('IB03', '2024-11-11 16:00:00', '2024-11-11 18:45:00', 'AP01', 'AP04', 'P003', 'A001'), -- Madrid a Múnich (Iberia, A330)
('AE04', '2024-11-12 07:30:00', '2024-11-12 09:30:00', 'AP03', 'AP02', 'P001', 'A002'); -- París a Barcelona (Air Europa, A320);
INSERT 0 4
```

```
 flight_id | departure_datetime  |  arrival_datetime   | origin_airport_id | destination_airport_id | airplane_id | airline_id
-----------+---------------------+---------------------+-------------------+------------------------+-------------+------------
 IB01      | 2024-11-10 08:00:00 | 2024-11-10 10:30:00 | AP01              | AP03                   | P001        | A001
 AE02      | 2024-11-10 12:00:00 | 2024-11-10 14:00:00 | AP02              | AP01                   | P002        | A002
 IB03      | 2024-11-11 16:00:00 | 2024-11-11 18:45:00 | AP01              | AP04                   | P003        | A001
 AE04      | 2024-11-12 07:30:00 | 2024-11-12 09:30:00 | AP03              | AP02                   | P001        | A002
(4 rows)
```

INSERCION RESERVAS

INSERT INTO Bookings (flight_id, passenger_id, seat) VALUES ('IB01', 'PAS01', '14A'), ('IB01', 'PAS02', '14B'), ('AE02', 'PAS03', '05C'),('IB03', 'PAS01', '22D'), ('IB03', 'PAS04', '22E');

```
airports=# INSERT INTO Bookings (flight_id, passenger_id, seat) VALUES ('IB01', 'PAS01', '14A'), ('IB01', 'PAS02', '14B'), ('AE02', 'PAS03', '05C'),('IB03', 'PAS01', '2
2D'), ('IB03', 'PAS04', '22E');
INSERT 0 5
airports=# select * from bookings ;
 flight_id | passenger_id | seat
-----------+--------------+------
 IB01      | PAS01        | 14A
 IB01      | PAS02        | 14B
 AE02      | PAS03        | 05C
 IB03      | PAS01        | 22D
 IB03      | PAS04        | 22E
(5 rows)

airports=#
```

4. CONSULTAS:

4.1. Vuelos y su número de pasajeros, ordenado de mayor a menor

SELECT f.flight_id, COUNT(b.passenger_id) AS total_pasajeros FROM Flights f JOIN Bookings b ON f.flight_id = b.flight_id GROUP BY f.flight_id ORDER BY total_pasajeros DESC;

```
airports=# SELECT
    f.flight_id,
    COUNT(b.passenger_id) AS total_pasajeros
FROM
    Flights f
JOIN
    Bookings b ON f.flight_id = b.flight_id
GROUP BY
    f.flight_id
ORDER BY
    total_pasajeros DESC;
 flight_id | total_pasajeros
-----------+-----------------
 IB03      |               2
 IB01      |               2
 AE02      |               1
(3 rows)

airports=#
```

4.2. Identificador de los vuelos, con los identificadores de los aeropuertos

SELECT flight_id, origin_airport_id, destination_airport_id FROM Flights ORDER BY flight_id;

```
airports=# SELECT flight_id, origin_airport_id, destination_airport_id FROM Flights ORDER BY flight_id;
 flight_id | origin_airport_id | destination_airport_id
-----------+-------------------+------------------------
 AE02      | AP02              | AP01
 AE04      | AP03              | AP02
 IB01      | AP01              | AP03
 IB03      | AP01              | AP04
(4 rows)

airports=#
```

4.3. Combinaciones distintas de aeropuertos de origen y destino

SELECT DISTINCT A_Origen.airport_name AS aeropuerto_origen, A_Destino.airport_name AS aeropuerto_destino FROM Flights f JOIN Airports A_Origen ON f.origin_airport_id = A_Origen.airport_id JOIN Airports A_Destino ON f.destination_airport_id = A_Destino.airport_id ORDER BY aeropuerto_origen, aeropuerto_destino;

```
airports=# SELECT DISTINCT
    A_Origen.airport_name AS aeropuerto_origen,
    A_Destino.airport_name AS aeropuerto_destino
FROM
    Flights f
JOIN
    Airports A_Origen ON f.origin_airport_id = A_Origen.airport_id
JOIN
    Airports A_Destino ON f.destination_airport_id = A_Destino.airport_id
ORDER BY
    aeropuerto_origen, aeropuerto_destino;
        aeropuerto_origen        |         aeropuerto_destino
---------------------------------+------------------------------------
 Adolfo Suárez Madrid-Barajas    | Aeropuerto Internacional de Múnich
 Adolfo Suárez Madrid-Barajas    | Charles de Gaulle
 Aeropuerto de Barcelona-El Prat | Adolfo Suárez Madrid-Barajas
 Charles de Gaulle               | Aeropuerto de Barcelona-El Prat
(4 rows)

airports=#
```

4.4. Número de vuelos por avión

SELECT a.airplane_model, a.airplane_id, COUNT(f.flight_id) AS total_vuelos FROM Airplanes a JOIN Flights f ON a.airplane_id = f.airplane_id GROUP BY a.airplane_model, a.airplane_id ORDER BY total_vuelos DESC;

```
airports=# SELECT
    a.airplane_model,
    a.airplane_id,
    COUNT(f.flight_id) AS total_vuelos
FROM
    Airplanes a
JOIN
    Flights f ON a.airplane_id = f.airplane_id
GROUP BY
    a.airplane_model, a.airplane_id
ORDER BY
    total_vuelos DESC;
 airplane_model | airplane_id | total_vuelos
----------------+-------------+--------------
 Airbus A320    | P001        |            2
 Boeing 737     | P002        |            1
 Airbus A330    | P003        |            1
(3 rows)

airports=#
```

4.5. Suma total de pasajeros por aeropuerto, un pasajero puede pasar más de una vez por el mismo aeropuerto, se deben tener en cuenta aeropuertos de origen y destino

WITH PassengerTraffic AS (SELECT f.origin_airport_id AS airport_id FROM Flights f JOIN Bookings b ON f.flight_id = b.flight_id

```
UNION ALL

SELECT
    f.destination_airport_id AS airport_id
FROM
    Flights f
JOIN
    Bookings b ON f.flight_id = b.flight_id
```

) SELECT a.airport_name, COUNT(t.airport_id) AS total_pasajeros_transito FROM Airports a JOIN PassengerTraffic t ON a.airport_id = t.airport_id GROUP BY a.airport_name ORDER BY total_pasajeros_transito DESC;

```
airports=# WITH PassengerTraffic AS (
    -- Pasajeros en Aeropuerto de ORIGEN
    SELECT
        f.origin_airport_id AS airport_id
    FROM
        Flights f
    JOIN
        Bookings b ON f.flight_id = b.flight_id

    UNION ALL

    -- Pasajeros en Aeropuerto de DESTINO
    SELECT
        f.destination_airport_id AS airport_id
    FROM
        Flights f
    JOIN
        Bookings b ON f.flight_id = b.flight_id
)
SELECT
    a.airport_name,
    COUNT(t.airport_id) AS total_pasajeros_transito
FROM
    Airports a
JOIN
    PassengerTraffic t ON a.airport_id = t.airport_id
GROUP BY
    a.airport_name
ORDER BY
    total_pasajeros_transito DESC;
          airport_name           | total_pasajeros_transito
---------------------------------+--------------------------
 Adolfo Suárez Madrid-Barajas    |                        5
 Charles de Gaulle               |                        2
 Aeropuerto Internacional de Múnich |                     2
 Aeropuerto de Barcelona-El Prat |                        1
(4 rows)

airports=#
```