**Please answer the following questions using Airline DB database.**

**Instruction to attempt questions:**

* Students need to write queries for the questions mentioned in the using Airline DB database
* Read the questions carefully before writing the query in **Airline Playground** (in the Playground chapter of SQL)
* Airline DB: [https://www.skillovilla.com/playground/sql?exerciseId=0181e251-6ea8-4595-ae2b-0c690119f8db](file:///C:\Users\Hp\Downloads\•%09https:\www.skillovilla.com\playground\sql%3fexerciseId=0181e251-6ea8-4595-ae2b-0c690119f8db)

**How to submit the capstone:**

* Copy the SQL query code and paste it in the answer section in this file.
* Once the assignment is done, submit the file over LMS.

**Invalid Submissions:**

* Pasting pictures of the code as answer is **NOT** acceptable.
* Uploading output data (CSVs) of the SQL queries is **NOT** acceptable.

**Write your answers(query) in the answer and submit it. To write the answer in the assignment, please follow the below example in yellow**

Example:

Questions*: Extract all the columns of the flights table*

Answer: *SELECT \* FROM flights*

**Attempt the following Questions-**

1. ***Represent the “book\_date” column in “yyyy-mmm-dd” format using Bookings table***

*Expected output: book\_ref, book\_date (in “yyyy-mmm-dd” format) , total amount*

**Answer:**

**SELECT book\_ref, to\_char(book\_date, 'yyyy-mon-dd') AS book\_date, total\_amount**

**FROM Bookings**

1. **Get the following columns in the exact same sequence.**

Expected columns in the output: ticket\_no, boarding\_no, seat\_number, passenger\_id, passenger\_name.

**Answer:**

**SELECT**

**Ti.ticket\_no,**

**BP.boarding\_no,**

**BP.seat\_no AS seat\_number,**

**Ti.passenger\_id,**

**Ti.passenger\_name**

**FROM**

**Boarding\_passes BP**

**INNER JOIN Tickets Ti ON BP.ticket\_no = Ti.ticket\_no**

1. **Write a query to find the seat number which is least allocated among all the seats?**

**Answer:**

**SELECT**

**s.seat\_no**

**FROM**

**Seats s**

**LEFT JOIN**

**Boarding\_passes bp ON s.seat\_no = bp.seat\_no**

**GROUP BY**

**s.seat\_no**

**ORDER BY**

**COUNT(bp.seat\_no) ASC**

**LIMIT 1**

1. ***In the database, identify the month wise highest paying passenger name and passenger id.***

Expected output: Month\_name(“mmm-yy” format), passenger\_id, passenger\_name and total amount

**Answer:**

**WITH MonthlyPayments AS (**

**SELECT**

**TO\_CHAR(B.book\_date, 'Mon-YY') AS month\_year,**

**T.passenger\_id,**

**T.passenger\_name,**

**SUM(B.total\_amount) AS total\_amount**

**FROM**

**Bookings B**

**INNER JOIN**

**Tickets T ON B.book\_ref = T.book\_ref**

**GROUP BY**

**TO\_CHAR(B.book\_date, 'Mon-YY'),**

**T.passenger\_id,**

**T.passenger\_name**

**)**

**SELECT**

**month\_year,**

**passenger\_id,**

**passenger\_name,**

**total\_amount**

**FROM (**

**SELECT**

**month\_year,**

**passenger\_id,**

**passenger\_name,**

**total\_amount,**

**ROW\_NUMBER() OVER (PARTITION BY month\_year ORDER BY total\_amount DESC ) AS rn**

**FROM**

**MonthlyPayments**

**) ranked**

**WHERE**

**rn = 1**

1. ***In the database, identify the month wise least paying passenger name and passenger id?***

Expected output: Month\_name(“mmm-yy” format), passenger\_id, passenger\_name and total amount

**Answer:**

**WITH MonthlyPayments AS (**

**SELECT**

**TO\_CHAR(B.book\_date, 'Mon-YY') AS month\_year,**

**T.passenger\_id,**

**T.passenger\_name,**

**SUM(B.total\_amount) AS total\_amount**

**FROM**

**Bookings B**

**INNER JOIN**

**Tickets T ON B.book\_ref = T.book\_ref**

**GROUP BY**

**TO\_CHAR(B.book\_date, 'Mon-YY'),**

**T.passenger\_id,**

**T.passenger\_name**

**)**

**SELECT**

**month\_year,**

**passenger\_id,**

**passenger\_name,**

**total\_amount**

**FROM (**

**SELECT**

**month\_year,**

**passenger\_id,**

**passenger\_name,**

**total\_amount,**

**ROW\_NUMBER() OVER (PARTITION BY month\_year ORDER BY total\_amount ASC) AS rn**

**FROM**

**MonthlyPayments**

**) ranked**

**WHERE**

**rn = 1**

1. **Identify the travel details of non stop journeys or return journeys (having more than 1 flight).**

Expected Output: Passenger\_id, passenger\_name, ticket\_number and flight count.

**Answer:**

**WITH FlightCounts AS (**

**SELECT**

**T.passenger\_id,**

**T.passenger\_name,**

**T.ticket\_no,**

**COUNT(\*) AS flight\_count**

**FROM**

**Ticket\_flights TF**

**JOIN**

**Tickets T ON TF.ticket\_no = T.ticket\_no**

**GROUP BY**

**T.passenger\_id,**

**T.passenger\_name,**

**T.ticket\_no**

**HAVING**

**COUNT(\*) > 1**

**)**

**SELECT**

**FC.passenger\_id,**

**FC.passenger\_name,**

**FC.ticket\_no,**

**FC.flight\_count**

**FROM**

**FlightCounts FC**

1. **How many tickets are there without boarding passes?**

Expected Output: just one number is required.

**Answer:**

**SELECT**

**COUNT(\*) AS tickets\_without\_boarding\_passes**

**FROM**

**Tickets T**

**LEFT JOIN**

**Boarding\_passes BP ON T.ticket\_no = BP.ticket\_no**

**WHERE**

**BP.ticket\_no IS NULL**

1. **Identify details of the longest flight (using flights table)?**

Expected Output: Flight number, departure airport, arrival airport, aircraft code and durations.

**Answer:**

**WITH FlightDurations AS (**

**SELECT**

**flight\_no,**

**departure\_airport,**

**arrival\_airport,**

**aircraft\_code,**

**(actual\_arrival - actual\_departure) AS duration**

**FROM**

**Flights**

**WHERE**

**actual\_arrival IS NOT NULL AND actual\_departure IS NOT NULL**

**)**

**SELECT**

**flight\_no,**

**departure\_airport,**

**arrival\_airport,**

**aircraft\_code,**

**duration**

**FROM**

**FlightDurations**

**ORDER BY**

**duration DESC**

**LIMIT 1**

1. **Identify details of all the morning flights (morning means between 6AM to 11 AM, using flights table)?**

Expected output: flight\_id, flight\_number, scheduled\_departure, scheduled\_arrival and timings.

**Answer:**

**SELECT**

**flight\_id,**

**flight\_no AS flight\_number,**

**scheduled\_departure,**

**scheduled\_arrival,**

**'Morning' AS timings**

**FROM**

**Flights**

**WHERE**

**EXTRACT(HOUR FROM scheduled\_departure) >= 6**

**AND EXTRACT(HOUR FROM scheduled\_departure) < 11;**

1. **Identify the earliest morning flight available from every airport.**

Expected output: flight\_id, flight\_number, scheduled\_departure, scheduled\_arrival, departure airport and timings.

**Answer:**

**WITH EarliestMorningFlights AS (**

**SELECT**

**MIN(scheduled\_departure) AS earliest\_departure,**

**departure\_airport**

**FROM**

**Flights**

**WHERE**

**EXTRACT(HOUR FROM scheduled\_departure) >= 2**

**AND EXTRACT(HOUR FROM scheduled\_departure) < 6**

**GROUP BY**

**departure\_airport**

**)**

**SELECT**

**F.flight\_id,**

**F.flight\_no AS flight\_number,**

**F.scheduled\_departure,**

**F.scheduled\_arrival,**

**F.departure\_airport,**

**'Early Morning' AS timings**

**FROM**

**Flights F**

**INNER JOIN**

**EarliestMorningFlights EMF ON F.scheduled\_departure = EMF.earliest\_departure**

**AND F.departure\_airport = EMF.departure\_airport;**

1. **Questions:** **Find list of airport codes in Europe/Moscow timezone**

Expected Output: Airport\_code.

**Answer:**

**SELECT**

**airport\_code**

**FROM**

**Airports**

**WHERE**

**timezone = 'Europe/Moscow'**

1. **Write a query to get the count of seats in various fare condition for every aircraft code?**

Expected Outputs: Aircraft\_code, fare\_conditions ,seat count

**Answer:**

**SELECT**

**S.aircraft\_code,**

**S.fare\_conditions,**

**COUNT(\*) AS seat\_count**

**FROM**

**Seats S**

**GROUP BY**

**S.aircraft\_code,**

**S.fare\_conditions**

1. **How many aircrafts codes have at least one Business class seats?**

Expected Output : Count of aircraft codes

**Answer:**

**SELECT**

**COUNT(DISTINCT aircraft\_code) AS count\_of\_aircraft\_codes**

**FROM**

**Seats**

**WHERE**

**fare\_conditions = 'Business'**

1. **Find out the name of the airport having maximum number of departure flight**

Expected Output : Airport\_name

**Answer:**

**SELECT**

**A.airport\_name**

**FROM**

**Flights F**

**INNER JOIN**

**Airports A ON F.departure\_airport = A.airport\_code**

**GROUP BY**

**A.airport\_name**

**ORDER BY**

**COUNT(\*) DESC**

**LIMIT 1**

1. **Find out the name of the airport having least number of scheduled departure flights**

Expected Output : Airport\_name

**Answer:**

**SELECT**

**AD.airport\_name**

**FROM**

**(SELECT**

**departure\_airport,**

**COUNT(\*) AS departure\_count**

**FROM**

**Flights**

**GROUP BY**

**departure\_airport) AS DepartureCounts**

**INNER JOIN**

**Airports\_data AD ON DepartureCounts.departure\_airport = AD.airport\_code**

**ORDER BY**

**DepartureCounts.departure\_count**

**LIMIT 1**

1. **How many flights from ‘DME’ airport don’t have actual departure?**

Expected Output : Flight Count

**Answer:**

**SELECT**

**COUNT(\*) AS Flight\_Count**

**FROM**

**Flights**

**WHERE**

**departure\_airport = 'DME'**

**AND actual\_departure IS NULL**

1. **Identify flight ids having range between 3000 to 6000**

Expected Output : Flight\_Number , aircraft\_code, ranges

**Answer:**

**SELECT**

**F.flight\_no AS Flight\_Number,**

**F.aircraft\_code,**

**A.range AS Ranges**

**FROM**

**Flights F**

**INNER JOIN**

**Aircrafts A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**A.range BETWEEN 3000 AND 6000**

**GROUP BY 1,2,3**

1. **Write a query to get the count of flights flying between URS and KUF?**

Expected Output : Flight\_count

**Answer:**

**SELECT**

**COUNT(\*) AS Flight\_count**

**FROM**

**Flights**

**WHERE**

**(departure\_airport = 'URS' AND arrival\_airport = 'KUF') OR**

**(departure\_airport = 'KUF' AND arrival\_airport = 'URS')**

1. **Write a query to get the count of flights flying from either from NOZ or KRR?**

Expected Output : Flight count

**Answer:**

**SELECT**

**COUNT(\*) AS Flight\_count**

**FROM**

**Flights**

**WHERE**

**departure\_airport = 'NOZ' OR departure\_airport = 'KRR'**

1. **Write a query to get the count of flights flying from KZN,DME,NBC,NJC,GDX,SGC,VKO,ROV**

Expected Output : Departure airport ,count of flights flying from these airports.

**Answer:**

**SELECT**

**departure\_airport AS Departure\_airport,**

**COUNT(\*) AS Count\_flight\_flying\_from\_these\_airports**

**FROM**

**Flights**

**WHERE**

**departure\_airport IN ('KZN', 'DME', 'NBC', 'NJC', 'GDX', 'SGC', 'VKO', 'ROV')**

**GROUP BY**

**departure\_airport**

1. **Write a query to extract flight details having range between 3000 and 6000 and flying from DME**

Expected Output :Flight\_no,aircraft\_code,range,departure\_airport

**Answer:**

**SELECT**

**F.flight\_no AS Flight\_no,**

**F.aircraft\_code,**

**A.range,**

**F.departure\_airport**

**FROM**

**Flights F**

**JOIN**

**Aircrafts A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**A.range BETWEEN 3000 AND 6000**

**AND F.departure\_airport = 'DME'**

1. **Find the list of flight ids which are using aircrafts from “Airbus” company and got cancelled or delayed**

Expected Output : Flight\_id,aircraft\_model

**Answer:**

**SELECT**

**F.flight\_id AS Flight\_id,**

**(A.model::json->>'en') AS Aircraft\_model**

**FROM**

**Flights F**

**JOIN**

**Aircrafts A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**(A.model::json->>'en') LIKE '%Airbus%'**

**AND (F.status = 'Cancelled' OR F.status = 'Delayed')**

1. **Find the list of flight ids which are using aircrafts from “Boeing” company and got cancelled or delayed**

Expected Output : Flight\_id,aircraft\_model

**Answer:**

**SELECT**

**F.flight\_id AS Flight\_id,**

**(A.model::json->>'en') AS Aircraft\_model**

**FROM**

**Flights F**

**JOIN**

**Aircrafts A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**(A.model::json->>'en') LIKE '%Boeing%'**

**AND (F.status = 'Cancelled' OR F.status = 'Delayed')**

1. **Which airport(name) has most cancelled flights (arriving)?**

Expected Output : Airport\_name

Answer:

**SELECT**

**A.airport\_name AS Airport\_name**

**FROM**

**(SELECT**

**arrival\_airport,**

**COUNT(\*) AS canceled\_count**

**FROM**

**Flights**

**WHERE**

**status = 'Cancelled'**

**GROUP BY**

**arrival\_airport**

**ORDER BY**

**canceled\_count DESC**

**LIMIT 1) AS CancelCounts**

**INNER JOIN**

**Airports A ON CancelCounts.arrival\_airport = A.airport\_code**

1. ***Identify flight ids which are using “Airbus aircrafts”***

*Expected Output : Flight\_id,aircraft\_model*

**Answer:**

**SELECT**

**F.flight\_id AS Flight\_id,**

**(A.model::json->>'en') AS Aircraft\_model**

**FROM**

**Flights F**

**INNER JOIN**

**Aircrafts A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**(A.model::json->>'en') LIKE '%Airbus%'**

1. ***Identify date-wise last flight id flying from every airport?***

*Expected Output: Flight\_id,flight\_number,schedule\_departure,departure\_airport*

**Answer:**

**WITH RankedFlights AS (**

**SELECT**

**flight\_id AS Flight\_id,**

**flight\_no AS Flight\_number,**

**scheduled\_departure,**

**departure\_airport,**

**RANK() OVER(PARTITION BY DATE(scheduled\_departure), departure\_airport ORDER BY scheduled\_departure DESC) AS r**

**FROM**

**Flights**

**)**

**SELECT**

**Flight\_id as LastFlight,**

**Flight\_number,**

**scheduled\_departure,**

**departure\_airport**

**FROM**

**RankedFlights**

**WHERE**

**r = 1**

1. ***Identify list of customers who will get the refund due to cancellation of the flights and how much amount they will get?***

*Expected Output : Passenger\_name,total\_refund.*

**Answer:**

**SELECT**

**T.passenger\_name AS Passenger\_name, SUM(Tf.amount) AS total\_refund**

**FROM**

**Tickets AS T**

**INNER JOIN Ticket\_flights AS Tf**

**ON T.ticket\_no = Tf.ticket\_no**

**INNER JOIN Flights f**

**ON Tf.flight\_id = f.flight\_id**

**WHERE f.status = 'Cancelled'**

**GROUP BY T.passenger\_name**

1. ***Identify date wise first cancelled flight id flying for every airport?***

*Expected Output : Flight\_id,flight\_number,schedule\_departure,departure\_airport*

**Answer:**

**WITH FirstCancelledFlights AS (**

**SELECT**

**flight\_id AS Flight\_id,**

**flight\_no AS Flight\_number,**

**scheduled\_departure,**

**departure\_airport,**

**DATE(scheduled\_departure) AS flight\_date,**

**ROW\_NUMBER() OVER (PARTITION BY DATE(scheduled\_departure), departure\_airport ORDER BY scheduled\_departure) AS rn**

**FROM**

**Flights**

**WHERE**

**status = 'Cancelled'**

**)**

**SELECT**

**Flight\_id,**

**Flight\_number,**

**scheduled\_departure,**

**departure\_airport**

**FROM**

**FirstCancelledFlights**

**WHERE**

**rn = 1**

1. ***Identify list of Airbus flight ids which got cancelled.***

*Expected Output : Flight\_id*

**Answer:**

**SELECT**

**F.flight\_id AS Flight\_id**

**FROM**

**Flights F**

**JOIN**

**Aircrafts\_data A ON F.aircraft\_code = A.aircraft\_code**

**WHERE**

**CAST(A.model AS JSON)->>'en' LIKE 'Airbus%'**

**AND F.status = 'Cancelled'**

1. ***Identify list of flight ids having highest range.***

*Expected Output : Flight\_no, range*

**Answer:**

**WITH RankedFlights AS (**

**SELECT**

**F.flight\_id AS Flight\_id,**

**A.range AS Range,**

**RANK() OVER (ORDER BY A.range DESC) AS Rank**

**FROM**

**Flights F**

**JOIN**

**Aircrafts\_data A ON F.aircraft\_code = A.aircraft\_code**

**)**

**SELECT**

**Flight\_id,**

**Range**

**FROM**

**RankedFlights**

**WHERE**

**Rank = 1**