**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](•%09https:/www.skillovilla.com/playground/sql?exerciseId=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*

*bp.ticket\_no,*

*bp.boarding\_no,*

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

*t.passenger\_id,*

*t.passenger\_name*

*FROM boarding\_passes AS bp*

*LEFT JOIN tickets AS t;*

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

**Answer:** *SELECT seat\_no*

*FROM (*

*SELECT seat\_no, count(seat\_no)*

*from boarding\_passes*

*group by 1*

*order by count(seat\_no)*

*limit 1) AS seat\_no;*

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 table1 AS (*

*SELECT*

*TO\_CHAR(b.book\_date, 'mon-yy') AS month\_name,*

*t.passenger\_id,*

*t.passenger\_name,*

*b.total\_amount*

*FROM bookings AS b*

*JOIN tickets AS t*

*ON b.book\_ref = t.book\_ref*

*GROUP BY 1,2,3,4),*

*table2 AS (*

*SELECT \*,*

*ROW\_NUMBER() OVER(PARTITION BY month\_name ORDER BY total\_amount DESC) AS ranking*

*FROM table1*

*)*

*SELECT month\_name,*

*passenger\_id,*

*passenger\_name,*

*total\_amount*

*FROM table2*

*WHERE ranking =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 table1 AS (*

*SELECT*

*TO\_CHAR(b.book\_date, 'mon-yy') AS month\_name,*

*t.passenger\_id,*

*t.passenger\_name,*

*b.total\_amount*

*FROM bookings AS b*

*JOIN tickets AS t*

*ON b.book\_ref = t.book\_ref*

*GROUP BY 1,2,3,4),*

*table2 AS (*

*SELECT \*,*

*ROW\_NUMBER() OVER(PARTITION BY month\_name ORDER BY total\_amount ASC) AS ranking*

*FROM table1*

*)*

*SELECT month\_name,*

*passenger\_id,*

*passenger\_name,*

*total\_amount*

*FROM table2*

*WHERE ranking =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:** *SELECT t.passenger\_id,*

*t.passenger\_name,*

*t.ticket\_no AS ticket\_number ,*

*COUNT(tf.flight\_id) AS flight\_count*

*FROM tickets AS t*

*LEFT JOIN ticket\_flights AS tf*

*ON t.ticket\_no = tf.ticket\_no*

*GROUP BY 1, 2, 3*

*HAVING COUNT(tf.flight\_id) > 1*

*ORDER BY flight\_count DESC;*

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

Expected Output: just one number is required.

**Answer: 251**

*SELECT COUNT(t.ticket\_no)*

*FROM tickets AS t*

*LEFT JOIN boarding\_passes AS 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:** *SELECT*

*flight\_no AS flight\_number,*

*departure\_airport,*

*arrival\_airport,*

*aircraft\_code,*

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

*FROM flights*

*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,*

*CAST(scheduled\_departure AS TIME) AS timings*

*FROM flights*

*WHERE CAST(scheduled\_departure AS TIME) BETWEEN '06:00:00' AND '11:00:00';*

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 table1 AS (*

*SELECT*

*flight\_id,*

*flight\_no AS flight\_number,*

*scheduled\_departure,*

*scheduled\_arrival,*

*departure\_airport,*

*CAST(scheduled\_departure AS TIME) AS time,*

*ROW\_NUMBER() OVER(PARTITION BY departure\_airport ORDER BY scheduled\_departure)AS row\_num*

*FROM flights*

*WHERE CAST(scheduled\_departure AS TIME) BETWEEN '02:00:00' AND '06:00:00'*

*)*

*SELECT*

*flight\_id,*

*flight\_number,*

*scheduled\_departure,*

*scheduled\_arrival,*

*departure\_airport*

*FROM table1*

*WHERE row\_num = 1;*

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 aircraft\_code,*

*fare\_conditions,*

*COUNT(\*) AS seat\_count*

*FROM seats*

*GROUP BY aircraft\_code, fare\_conditions*

*ORDER BY 1;*

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\_aircrafts*

*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 airports a*

*LEFT JOIN flights f*

*ON f.departure\_airport = a.airport\_code*

*GROUP BY 1*

*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 a.airport\_name*

*FROM airports AS a*

*JOIN flights AS f*

*ON f.departure\_airport = a.airport\_code*

*GROUP BY 1*

*ORDER BY COUNT(scheduled\_departure) ASC*

*LIMIT 1;*

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

Expected Output : Flight Count

**Answer: 127**

*SELECT COUNT(flight\_id) 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, a.aircraft\_code, a.range*

*FROM flights AS f*

*LEFT JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.range BETWEEN 3000 AND 6000*

*GROUP BY 1,2*

*ORDER BY 3;*

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

Expected Output : Flight\_count

**Answer:189**

*SELECT COUNT(\*)*

*from flights*

*where departure\_airport in ('URS','KUF');*

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

Expected Output : Flight count

**Answer: 301**

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

*FROM flights*

*WHERE departure\_airport IN ('NOZ','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, COUNT(\*) AS flight\_count*

*FROM flights*

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

*GROUP BY 1;*

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, f.aircraft\_code, a.range, f.departure\_airport*

*FROM flights AS f*

*LEFT JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.range BETWEEN 3000 AND 6000*

*AND departure\_airport ='DME'*

*GROUP BY 1,2,3,4*

*ORDER BY a.range;*

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, a.model*

*FROM flights AS f*

*LEFT JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.model LIKE '%Airbus%'*

*AND f.status IN ('Cancelled','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, a.model*

*FROM flights AS f*

*LEFT JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.model LIKE '%Boeing%'*

*AND f.status IN ('Cancelled','Delayed');*

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

Expected Output : Airport\_name

**Answer:** *SELECT a.airport\_name*

*FROM airports AS a*

*JOIN flights AS f*

*ON f.arrival\_airport = a.airport\_code*

*WHERE f.status = 'Cancelled'*

*GROUP BY 1*

*ORDER BY COUNT(\*) DESC*

*LIMIT 2;*

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

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

**Answer:** *SELECT f.flight\_id, a.model*

*FROM flights AS f*

*JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.model LIKE '%Airbus%'*

*ORDER BY 1;*

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

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

**Answer:** *WITH table1 AS (*

*SELECT*

*flight\_id,*

*flight\_no AS flight\_number,*

*scheduled\_departure,*

*departure\_airport,*

*RANK() OVER(PARTITION BY departure\_airport ORDER BY scheduled\_departure DESC)AS ranking*

*FROM flights)*

*SELECT flight\_id,*

*flight\_number,*

*scheduled\_departure,*

*departure\_airport*

*FROM table1*

*WHERE ranking= 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, SUM(tf.amount) AS total\_refund*

*FROM tickets AS t*

*JOIN ticket\_flights AS tf*

*ON t.ticket\_no = tf.ticket\_no*

*JOIN flights AS f*

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

*WHERE f.status = 'Cancelled'*

*GROUP BY 1;*

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

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

**Answer:***WITH table1 AS (*

*SELECT*

*flight\_id,*

*flight\_no AS flight\_number,*

*scheduled\_departure,*

*departure\_airport,*

*RANK() OVER(PARTITION BY departure\_airport ORDER BY scheduled\_departure) AS ranking*

*FROM flights*

*WHERE status ='Cancelled')*

*SELECT flight\_id,*

*flight\_number,*

*scheduled\_departure,*

*departure\_airport*

*FROM table1*

*WHERE ranking = 1;*

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

*Expected Output : Flight\_id*

**Answer:** *SELECT f.flight\_id*

*FROM flights AS f*

*JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code*

*WHERE a.model LIKE '%Airbus%'*

*AND f.status ='Cancelled';*

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

*Expected Output : Flight\_no, range*

**Answer:** *SELECT flight\_no, range*

*FROM (*

*SELECT*

*f.flight\_no,*

*a.range,*

*RANK() OVER(ORDER BY a.range DESC) AS ranking*

*FROM flights AS f*

*JOIN aircrafts AS a*

*ON f.aircraft\_code = a.aircraft\_code) AS table1*

*WHERE ranking = 1;*