**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**

**order by 1**

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 t.ticket\_no,**

**boarding\_no,**

**seat\_no ,**

**passenger\_id ,**

**passenger\_name**

**from**

**tickets as t**

**join**

**boarding\_passes as bp**

**on t.ticket\_no=bp.ticket\_no**

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

**Answer:**

**select**

**seat\_no,**

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

**from**

**boarding\_passes**

**group by**

**seat\_no**

**order by**

**seat\_count**

**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:**

**select**

**to\_char(book\_date,'mon-yy') month\_name,**

**passenger\_id,**

**passenger\_name,**

**sum(total\_amount) total\_amount**

**from**

**tickets as t**

**join**

**bookings as b**

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

**group by 1,2,3**

**order by 4 desc**

**limit 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:**

**select**

**to\_char(book\_date,'mon-yy')**

**month\_name,**

**passenger\_id,**

**passenger\_name,**

**sum(total\_amount) total\_amount**

**from**

**tickets as t**

**join**

**bookings as b**

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

**group by 1,2,3**

**order by 4 limit 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**

**passenger\_id,**

**passenger\_name,**

**tf.ticket\_no,**

**count(f.flight\_id) as flight\_count**

**from**

**ticket\_flights as tf**

**join tickets as t**

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

**join flights as f**

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

**where f.status = 'Arrived'**

**group by 1,2,3**

**having count(f.flight\_id)>1**

**order by 1**

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

Expected Output: just one number is required.

**Answer: there is zero tickets without boarding passes**

**Query: select**

**boarding\_no,**

**count(ticket\_no) as ticket\_count**

**from boarding\_passes**

**group by 1**

**order by 1**

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,**

**departure\_airport,**

**arrival\_airport,**

**aircraft\_code,**

**(scheduled\_departure -scheduled\_arrival) as durations**

**From flights**

**order by 1 desc**

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\_number,**

**scheduled\_departure,**

**scheduled\_arrival,**

**case**

**when extract(hour from scheduled\_departure) >= 6**

**and extract(hour from scheduled\_departure) < 11 then 'Morning'**

**else 'Not Morning'**

**end 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: select**

**flight\_id,**

**flight\_no,**

**scheduled\_departure,**

**scheduled\_arrival,**

**departure\_airport,**

**case**

**when extract(hour from scheduled\_departure) >= 2**

**and extract(hour from scheduled\_arrival) < 6 then 'early Morning'**

**else 'Morning'**

**end as timings**

**from**

**flights**

**where**

**extract(hour from scheduled\_departure) >= 2**

**and extract(hour from scheduled\_arrival) < 6**

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(seat\_no) as seat\_count**

**from seats**

**group by 1,2**

**order by 1 asc**

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 departure\_airport as airport\_name**

**from flights**

**group by departure\_airport**

**order by count(scheduled\_departure) desc**

**limit 1**

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

Expected Output : Airport\_name

**Answer:** **select departure\_airport as airport\_name**

**from flights**

**group by departure\_airport**

**order by count(scheduled\_departure) dasc**

**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**

**flight\_no,**

**ac.aircraft\_code,**

**ac.range**

**from**

**aircrafts as ac**

**join**

**flights as f**

**on ac.aircraft\_code=f.aircraft\_code**

**where ac.range between 3000 and 6000**

**order by 1**

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' or arrival\_airport = 'URS')**

**and (departure\_airport = 'KUF' or arrival\_airport = 'KUF')**

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 arrival\_airport = 'NOZ')**

**or (departure\_airport = 'KRR' or arrival\_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,**

**count(\*) as count\_of\_flights\_flying\_from\_these\_airports**

**from**

**flights**

**where**

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

**group by 1**

**order 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**

**flight\_no,**

**ac.aircraft\_code,**

**ac.range,**

**departure\_airport**

**from**

**aircrafts as ac**

**join**

**flights as f**

**on ac.aircraft\_code=f.aircraft\_code**

**where (ac.range between 3000 and 6000)**

**and departure\_airport='DME'**

**order by 1**

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**

**flight\_id,**

**model as aircraft\_model**

**from**

**aircrafts as ac**

**join**

**flights as f**

**on ac.aircraft\_code=f.aircraft\_code**

**where model 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**

**flight\_id,**

**model as aircraft\_model**

**from**

**aircrafts as ac**

**join**

**flights as f**

**on ac.aircraft\_code=f.aircraft\_code**

**where model like'%Boeing%'**

**and(f.status = 'Cancelled' or f.status = 'Delayed');**

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

Expected Output : Airport\_name

**Answer:**

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

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

**Answer: select**

**distinct flight\_id,**

**model**

**from flights as f**

**join**

**aircrafts as ac**

**on f.aircraft\_code=ac.aircraft\_code**

**where model 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 ranked\_flights AS (**

**SELECT**

**flight\_id,**

**flight\_no,**

**scheduled\_departure,**

**departure\_airport,**

**TO\_CHAR(scheduled\_departure::timestamp, 'YYYY-MM-DD') AS extracted\_date,**

**ROW\_NUMBER() OVER (PARTITION BY TO\_CHAR(scheduled\_departure::timestamp, 'YYYY-MM-DD'), departure\_airport ORDER BY scheduled\_departure DESC) AS flight\_rank**

**FROM**

**flights**

**)**

**SELECT**

**flight\_id,**

**flight\_no,**

**scheduled\_departure,**

**departure\_airport**

**FROM**

**ranked\_flights**

**WHERE**

**flight\_rank = 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**

**ticket\_flights AS tf**

**JOIN**

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

**JOIN**

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

**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 ranked\_cancelled\_flights as (**

**select**

**flight\_id,**

**flight\_no,**

**scheduled\_departure,**

**departure\_airport,**

**to\_char(scheduled\_departure::timestamp, 'yyyy-mm-dd') as extracted\_date,**

**row\_number() over (partition by to\_char(scheduled\_departure::timestamp, 'yyyy-mm-dd'), departure\_airport order by scheduled\_departure) as flight\_rank**

**from**

**flights**

**where**

**status = 'cancelled'**

**)**

**select**

**flight\_id,**

**flight\_no,**

**scheduled\_departure,**

**departure\_airport**

**from**

**ranked\_cancelled\_flights**

**where**

**flight\_rank = 1;**

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

*Expected Output : Flight\_id*

**Answer: select**

**t.passenger\_name,**

**sum(tf.amount) as total\_refund**

**from**

**ticket\_flights as tf**

**join**

**flights as f on tf.flight\_id = f.flight\_id**

**join**

**tickets as t on t.ticket\_no = tf.ticket\_no**

**where**

**f.status = 'cancelled'**

**group by**

**t.passenger\_name**

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

*Expected Output : Flight\_no, range*

**Answer: with d1 as**

**(**

**select**

**f.flight\_id,**

**ac.range,**

**dense\_rank() over(order by ac.range desc) as rnk**

**from**

**aircrafts as ac**

**join**

**flights as f**

**on ac.aircraft\_code=f.aircraft\_code**

**)**

**select**

**flight\_id,**

**range**

**from d1**

**where**

**rnk=1**

**order by flight\_id desc**