**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-**

* ***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 booking\_date,total\_amount

from bookings;

* **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: Since there will be some tickets with no boarding passes, I am using Left Join, alternatively if we want only those tickets which has boarding passes,we can use Inner Join**

Select

t.ticket\_no,boarding\_no,seat\_no,passenger\_id,

passenger\_name

from

tickets as t

left join

boarding\_passes as bp

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

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

**Answer:** Select seat\_no

from

boarding\_passes

group by 1

order by count(seat\_no) asc

limit 3;

(Or)

Select seat\_no from

boarding\_passes

group by 1

order by count(seat\_no) asc

limit 1;

**Note: There are 3 seats with lowest allocation**

* ***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 monthly\_bookings as

(Select to\_char(book\_date,'MON-YY') as booking\_date,

passenger\_id,

passenger\_name,

sum(total\_amount) as total\_booking\_amount

from

bookings as b

join

tickets as t

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

join

ticket\_flights as tf

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

group by 1,2,3)

Select

booking\_date,

passenger\_id,

passenger\_name,

total\_booking\_amount

from

(Select booking\_date,

passenger\_id,

passenger\_name,

total\_booking\_amount,

row\_number()over(partition by booking\_date order by total\_booking\_amount desc) as ranking

from

monthly\_bookings) as a

where ranking = 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 monthly\_bookings as

(Select to\_char(book\_date,'MON-YY') as booking\_date,

passenger\_id,

passenger\_name,

sum(total\_amount) as total\_booking\_amount

from

bookings as b

join

tickets as t

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

join

ticket\_flights as tf

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

group by 1,2,3)

Select

booking\_date,

passenger\_id,

passenger\_name,

total\_booking\_amount

from

(Select booking\_date,

passenger\_id,

passenger\_name,

total\_booking\_amount,

row\_number()over(partition by booking\_date order by total\_booking\_amount asc) as ranking

from

monthly\_bookings) as a

where ranking = 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(flight\_id) as flight\_count

from

ticket\_flights as tf

join

tickets as t

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

group by 1,2,3

having count(flight\_id) > 1;

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

Expected Output: just one number is required.

**Answer:** Select

count(\*)

from

tickets as t

left join

boarding\_passes as bp

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

where boarding\_no is null;

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

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

**Answer:**  Select

distinct flight\_no,

departure\_airport,

arrival\_airport,

aircraft\_code,

(scheduled\_arrival - scheduled\_departure) as duration

from flights

order by 5 desc

limit 4;

(OR)

Select

distinct flight\_no,

departure\_airport,

arrival\_airport,

aircraft\_code,

(scheduled\_arrival - scheduled\_departure) as duration

from flights

order by 5 desc

limit 1;

**Note: There are 4 flights with same duration, hence using limit 4. If we want only a single output, we can use limit 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,

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';

* **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 flight\_details as (Select

flight\_id,

Flight\_no,

scheduled\_departure,

scheduled\_arrival,departure\_airport,cast(scheduled\_departure as time) as timings from flights)

Select flight\_id,

Flight\_no,

scheduled\_departure,

scheduled\_arrival,departure\_airport,timings from

(Select \*,row\_number()over(partition by departure\_airport order by timings asc) as ranking

from flight\_details where timings between '06:00:00' and '11:00:00' ) as a

where ranking = 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';

* **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;

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

Expected Output : Count of aircraft codes

**Answer:** Select count(distinct aircraft\_code)

from seats

where fare\_conditions = 'Business';

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

Expected Output : Airport\_name

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

from

flights as f

join

airports as a

on f.departure\_airport = a.airport\_code

group by 1

order by count(flight\_id) desc

limit 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

flights as f

join

airports as a

on f.departure\_airport = a.airport\_code

group by 1

order by count(flight\_id) asc

limit 1;

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

Expected Output : Flight Count

**Answer:** Select count(\*) as count\_flights from flights

where departure\_airport = 'DME' and actual\_departure is not null;

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

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

**Answer:** Select flight\_no,f.aircraft\_code,range

from flights as f

left join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

where range between 3000 and 6000;

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

Expected Output : Flight\_count

**Answer:** Select count(\*) as count\_flight

from flights

where

departure\_airport in ('URS','KUF') and arrival\_airport in ('URS','KUF');

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

Expected Output : Flight count

**Answer:** Select count(\*) as count\_flight

from flights

where

departure\_airport in ('NOZ','KRR') ;

* **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\_flight

from flights

where

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

group by 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 distinct flight\_no,f.aircraft\_code,range,departure\_airport

from flights as f

join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

where departure\_airport = 'DME' and range between 3000 and 6000;

* **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

from flights as f

join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

where

status in ('Cancelled','Delayed') and model like '%Airbus%';

* **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

from flights as f

join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

where status in ('Cancelled','Delayed') and model like '%Boeing%';

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

Expected Output : Airport\_name

**Answer:** Select airport\_name

from

flights as f

join airports as a

on f.arrival\_airport = a.airport\_code

where status = 'Cancelled'

group by 1

order by count(flight\_id) desc

limit 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 a

on f.aircraft\_code = a.aircraft\_code

where model like '%Airbus%';

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

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

**Answer:** Select flight\_id,flight\_no,scheduled\_departure,departure\_airport

from

(Select

flight\_id,flight\_no,scheduled\_departure,departure\_airport,

row\_number()over(partition by cast(scheduled\_departure as date),departure\_airport order by cast(scheduled\_departure as time) desc) as ranking

from flights) as a

where ranking = 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:** With cancelled\_bookings as (Select \*

from

ticket\_flights as tf

left join

tickets as t

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

left join

flights as f

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

Select passenger\_name,sum(amount)

from

cancelled\_bookings

where status = 'Cancelled'

group by 1;

**Note:It seems that there are some flight ids in flights table which does not have a corresponding flight\_id in the ticket flights table.**

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

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

**Answer:** Select flight\_id,flight\_no,scheduled\_departure,departure\_airport

from

(Select

flight\_id,flight\_no,scheduled\_departure,departure\_airport,

row\_number()over(partition by cast(scheduled\_departure as date),departure\_airport order by cast(scheduled\_departure as time) asc) as ranking

from flights

where status = 'Cancelled') as a

where ranking = 1;

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

*Expected Output : Flight\_id*

**Answer:** Select flight\_id from

flights as f

join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

where status = 'Cancelled' and model like '%Airbus%';

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

*Expected Output : Flight\_no, range*

**Answer:** Select flight\_no,max(range)

from

flights as f

join

aircrafts as a

on f.aircraft\_code = a.aircraft\_code

group by 1;