Assignment 1
DTI 5126: Fundamentals for Applied Data Science
2023

GROUP 3

1. Aya Metwally

2. Amira Abu Issa

Part A

a)

```sql
-- a)
USE Flight_System;
SELECT
    COUNT(DISTINCT TAIL_NUMBER) AS num_aircrafts,
    COUNT(*) AS total_flights,
    MIN(DEPARTURE_DELAY) AS min_departure_delay,
    MAX(DEPARTURE_DELAY) AS max_departure_delay,
    AVG(DEPARTURE_DELAY) AS avg_departure_delay
FROM FLIGHTS;
```

Results | Messages

	num_aircrafts	total_flights	min_departure_delay	max_departure_delay	avg_departure_delay
1	4402	478000	-48	1988	9

b)

```sql
-- b)
CREATE VIEW FlightSummaryView AS
SELECT
  CONVERT(date, CONVERT(varchar, YEAR) + '-' + CONVERT(varchar, MONTH) + '-' + CONVERT(varchar, DAY)) AS date,
  ORIGIN_AIRPORT AS iata_code,
  AIRPORT AS origin_airport,
  CITY + ', ' + STATE + ', ' + COUNTRY AS Address,
  COUNT(*) AS total_flights
FROM FLIGHTS f
JOIN AIRPORTS a ON f.ORIGIN_AIRPORT = a.IATA_CODE
WHERE YEAR = 2015 AND MONTH = 1 AND DAY BETWEEN 1 AND 7
GROUP BY CONVERT(date, CONVERT(varchar, YEAR) + '-' + CONVERT(varchar, MONTH) + '-' + CONVERT(varchar, DAY)),
ORIGIN_AIRPORT, AIRPORT, CITY, STATE, COUNTRY;
SELECT * FROM FlightSummaryView ORDER BY iata_code desc;
```

	date	iata_code	origin_airport	Address	total_flights
1	2015-01-03	YUM	Yuma International Airport	Yuma, AZ, USA	5
2	2015-01-05	YUM	Yuma International Airport	Yuma, AZ, USA	6
3	2015-01-01	YUM	Yuma International Airport	Yuma, AZ, USA	6
4	2015-01-04	YUM	Yuma International Airport	Yuma, AZ, USA	6
5	2015-01-07	YUM	Yuma International Airport	Yuma, AZ, USA	6
6	2015-01-06	YUM	Yuma International Airport	Yuma, AZ, USA	5
7	2015-01-02	YUM	Yuma International Airport	Yuma, AZ, USA	6
8	2015-01-07	YAK	Yakutat Airport	Yakutat, AK, USA	2
9	2015-01-06	YAK	Yakutat Airport	Yakutat, AK, USA	2
10	2015-01-02	YAK	Yakutat Airport	Yakutat, AK, USA	2
11	2015-01-04	YAK	Yakutat Airport	Yakutat, AK, USA	2
12	2015-01-03	YAK	Yakutat Airport	Yakutat, AK, USA	2
13	2015-01-05	YAK	Yakutat Airport	Yakutat, AK, USA	2
14	2015-01-01	XNA	Northwest Arkansas Regional Airport	Fayetteville/Springdale/Rogers, AR, USA	18
15	2015-01-03	XNA	Northwest Arkansas Regional Airport	Fayetteville/Springdale/Rogers, AR, USA	20
16	2015-01-05	XNA	Northwest Arkansas Regional Airport	Fayetteville/Springdale/Rogers, AR, USA	24
17	2015-01-06	XNA	Northwest Arkansas Regional Airport	Fayetteville/Springdale/Rogers, AR, USA	29

Query executed successfully. | (local) (14.0 RTM) | DESKTOP-STQVT9U\Dell (61) | Flight_System | 00:00:00 | 2,162 rows

c)

```sql
--c)
WITH rank_routes AS (
SELECT
  origin_airport,
  destination_airport,
  COUNT(*) AS nb_flights
FROM FLIGHTS
GROUP BY origin_airport,destination_airport),
ranked_routes AS (
SELECT
  origin_airport,
  destination_airport,
  ROW_NUMBER() OVER(PARTITION BY origin_airport ORDER BY nb_flights DESC) AS rank FROM rank_routes)
SELECT
  origin_airport,
  destination_airport,
  rank
FROM ranked_routes
WHERE rank <= 3
ORDER BY origin_airport,rank;
```

	origin_airport	destination_airport	rank
1	ABE	DTW	1
2	ABE	ATL	2
3	ABE	ORD	3
4	ABI	DFW	1
5	ABQ	PHX	1
6	ABQ	LAX	2
7	ABQ	DAL	3
8	ABR	MSP	1
9	ABY	ATL	1
10	ACT	DFW	1

Query executed successfully.   (local) (14.0 RTM) | DESKTOP-STQVT9U\Dell (61) | Flight_System | 00:00:00 | 747 rows

d)

```sql
--d)
SELECT AIRPORTS.IATA_CODE AS "airport iata_code", FLIGHTS.ORIGIN_AIRPORT AS "airport name",
airlines.IATA_CODE AS "airline iata_code" , airlines.AIRLINE AS "airline name",
FLIGHTS.FLIGHT_NUMBER, FLIGHTS.TAIL_NUMBER,
FLIGHTS.DESTINATION_AIRPORT,
FLIGHTS.DEPARTURE_TIME, FLIGHTS.ARRIVAL_TIME
FROM  FLIGHTS
left JOIN AIRPORTS ON FLIGHTS.ORIGIN_AIRPORT = AIRPORTS.IATA_CODE
left JOIN airlines ON FLIGHTS.AIRLINE =airlines.IATA_CODE
WHERE
DAY_OF_WEEK IN (6,7)
AND FLIGHTS.ARRIVAL_TIME   >= 0400
AND FLIGHTS.ARRIVAL_TIME <= 0500
ORDER BY FLIGHTS.ARRIVAL_TIME;


--d)
SELECT AIRPORTS.IATA_CODE AS "airport iata_code", FLIGHTS.ORIGIN_AIRPORT AS "airport name",
airlines.IATA_CODE AS "airline iata_code" , airlines.AIRLINE AS "airline name",
FLIGHTS.FLIGHT_NUMBER, FLIGHTS.TAIL_NUMBER,
FLIGHTS.DESTINATION_AIRPORT,
FLIGHTS.DEPARTURE_TIME, FLIGHTS.ARRIVAL_TIME
FROM  FLIGHTS
INNER JOIN AIRPORTS ON FLIGHTS.ORIGIN_AIRPORT = AIRPORTS.IATA_CODE
INNER JOIN airlines ON FLIGHTS.AIRLINE =airlines.IATA_CODE
WHERE
DAY_OF_WEEK IN (6,7)
AND FLIGHTS.ARRIVAL_TIME   >= 0400
AND FLIGHTS.ARRIVAL_TIME <= 0500
```

	airport iata_code	airport name	airline iata_code	airline name	FLIGHT_NUMBER	TAIL_NUMBER	DESTINATION_AIRPORT	DEPARTURE_TIME	ARRIVAL_TIME
1	EWR	EWR	B6	JetBlue Airways	1389	N796JB	SJU	2313	0400
2	FLL	FLL	B6	JetBlue Airways	706	N529JB	EWR	0136	0401
3	DEN	DEN	F9	Frontier Airlines Inc.	432	N905FR	OMA	0137	0402
4	PHX	PHX	US	US Airways Inc.	889	N669AW	MSP	2352	0402
5	OGG	OGG	UA	United Air Lines Inc.	1157	N73251	LAX	2123	0402
6	MCO	MCO	B6	JetBlue Airways	767	N317JB	PSE	0016	0403
7	LAS	LAS	AS	Alaska Airlines Inc.	165	N581AS	ANC	2355	0403
8	OGG	OGG	UA	United Air Lines Inc.	1157	N18220	LAX	2129	0406
9	ANC	ANC	AS	Alaska Airlines Inc.	98	N435AS	SEA	2350	0406
10	FAT	FAT	AA	American Airlines Inc.	1242	N4XMAA	DFW	2330	0408

Query executed successfully.   (local) (14.0 RTM) | DESKTOP-STQVT9U\Dell (61) | Flight_System | 00:00:00 | 226 rows

e)

```sql
--e) using from subset subquery
SELECT JFK_Flights,
       CAST(JFK_Flights AS FLOAT) / TotalFlights.Total * 100 AS JFK_Percentage
FROM (
    SELECT COUNT(*) AS JFK_Flights
    FROM FLIGHTS
    WHERE ORIGIN_AIRPORT = 'JFK'
) AS JFKCount, (
    SELECT COUNT(*) AS Total
    FROM FLIGHTS
) AS TotalFlights
```

Results | Messages

	JFK_Flights	JFK_Percentage
1	8600	1.799916317991632

f)

```sql
--f)
-- Retrieve the flight information for all flights
SELECT *
FROM FLIGHTS
WHERE DESTINATION_AIRPORT IN ('JFK', 'LGA', 'EWR')
    AND ELAPSED_TIME > 500
    AND CANCELLED = 0;
-- Update their cancelled status from 0 to 1.
UPDATE FLIGHTS
SET CANCELLED = 1
WHERE DESTINATION_AIRPORT IN ('JFK', 'LGA', 'EWR')
    AND ELAPSED_TIME > 500
    AND CANCELLED = 0;
```

Results | Messages

	_record_number	YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	FLIGHT_NUMBER	TAIL_NUMBER	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDULED_DEPARTURE	DEPARTURE_TIME	DEPARTURE_DELAY	TAXI_OUT	W
1	13358	2015	1	1	4	UA	14	N67052	HNL	EWR	2125	2149	24	20	2
2	61693	2015	1	4	7	UA	14	N76064	HNL	EWR	2125	2131	6	15	2
3	168445	2015	1	11	7	UA	14	N69059	HNL	EWR	2125	2118	-7	17	2
4	206594	2015	1	13	2	HA	50	N380HA	HNL	JFK	1625	1627	2	13	1
5	242094	2015	1	15	4	UA	14	N69059	HNL	EWR	2125	2128	3	22	2
6	270460	2015	1	17	6	UA	14	N76054	HNL	EWR	2125	2120	-5	27	2
7	452207	2015	1	20	4	UA	14	N67052	HNL	EWR	2125	2118	7	17	2

Query executed successfully.   (local) (14.0 RTM) | DESKTOP-STQVT9U\Dell (55) | Flight_System | 00:00:00 | 71 rows

g)

```sql
--g)
WITH Departure_Delays AS (
    SELECT a.IATA_CODE, a.AIRLINE,
            CASE
                WHEN f.DEPARTURE_DELAY > 120 THEN 'Big Delay'
                WHEN f.DEPARTURE_DELAY BETWEEN 60 AND 30 THEN 'Medium Delay'
                ELSE 'Small Delay'
            END AS Delay_Category,
            COUNT(*) AS Total_Delays
    FROM FLIGHTS f
    INNER JOIN AIRLINES a ON f.AIRLINE = a.IATA_CODE
    WHERE f.DEPARTURE_DELAY > 0
    GROUP BY a.IATA_CODE, a.AIRLINE,
            CASE
                WHEN f.DEPARTURE_DELAY > 120 THEN 'Big Delay'
                WHEN f.DEPARTURE_DELAY BETWEEN 60 AND 30 THEN 'Medium Delay'
                ELSE 'Small Delay'
            END
)
SELECT * INTO Departure_Delays
FROM Departure_Delays
ORDER BY Total_Delays DESC;
select * from Departure_Delays order by Total_Delays DESC;
```
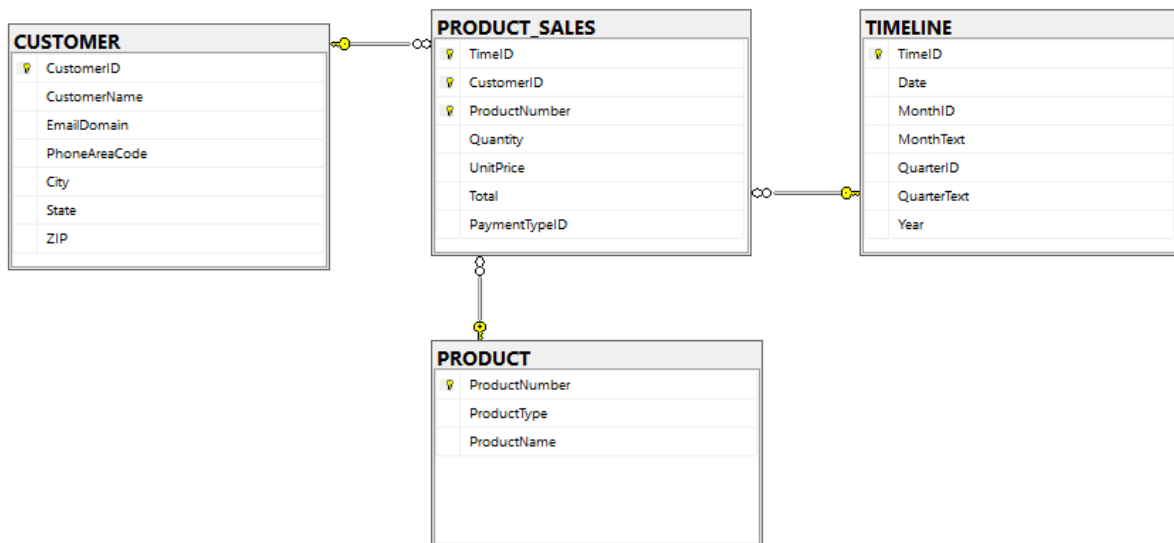
▦ Results   ▥ Messages

	IATA_CODE	AIRLINE	Delay_Category	Total_Delays
1	WN	Southwest Airlines Co.	Small Delay	43091
2	UA	United Air Lines Inc.	Small Delay	19678
3	DL	Delta Air Lines Inc.	Small Delay	18998
4	AA	American Airlines Inc.	Small Delay	15908
5	OO	Skywest Airlines Inc.	Small Delay	15309
6	EV	Atlantic Southeast Airlines	Small Delay	15013
7	MQ	American Eagle Airlines Inc.	Small Delay	11489
8	US	US Airways Inc.	Small Delay	9812
9	B6	JetBlue Airways	Small Delay	7520
10	NK	Spirit Air Lines	Small Delay	3510
11	AS	Alaska Airlines Inc.	Small Delay	3487
12	F9	Frontier Airlines Inc.	Small Delay	2705
13	HA	Hawaiian Airlines Inc.	Small Delay	1859
14	OO	Skywest Airlines Inc.	Big Delay	1547
15	VX	Virgin America	Small Delay	1518
16	EV	Atlantic Southeast Airlines	Big Delay	1237
17	WN	Southwest Airlines Co.	Big Delay	1225
18	UA	United Air Lines Inc.	Big Delay	924
19	MQ	American Eagle Airlines Inc.	Big Delay	906
20	AA	American Airlines Inc.	Big Delay	879
21	DL	Delta Air Lines Inc.	Big Delay	771
22	B6	JetBlue Airways	Big Delay	466
23	US	US Airways Inc.	Big Delay	443
24	F9	Frontier Airlines Inc.	Big Delay	337
25	NK	Spirit Air Lines	Big Delay	265
26	AS	Alaska Airlines Inc.	Big Delay	177
27	VX	Virgin America	Big Delay	72
28	HA	Hawaiian Airlines Inc.	Big Delay	27

✅ Query executed successfully.

Part 2

1- Star schema

2- a)

```
--2)--a
SELECT c.CustomerID, c.CustomerName, SUM(ps.Quantity) AS Total_Quantity, SUM(ps.Total) AS Total_Amount
FROM CUSTOMER c
INNER JOIN PRODUCT_SALES ps ON c.CustomerID = ps.CustomerID
INNER JOIN TIMELINE t ON ps.TimeID = t.TimeID
WHERE t.Date BETWEEN DATEADD(day, -90, '2018-05-31') AND '2018-05-31'
GROUP BY c.CustomerID, c.CustomerName;
```

⊞ Results ▤ Messages

	CustomerID	CustomerName	Total_Quantity	Total_Amount
1	1	Jacobs, Nancy	3	47.89
2	3	Able, Ralph	1	24.95
3	4	Baker, Susan	3	64.85
4	5	Eagleton, Sam	3	47.89
5	6	Foxtrot, Kathy	5	94.79
6	7	George, Sally	4	94.80
7	8	Hullett, Shawn	3	54.89
8	9	Pearson, Bobbi	5	70.83
9	11	Tyler, Jenny	6	109.78

✅ Query executed successfully.

b)

```sql
--b
WITH AvgOrder AS (
    SELECT AVG(Total) AS OverallAvg
    FROM PRODUCT_SALES
), CustomerAvg AS (
    SELECT c.CustomerID, c.CustomerName, AVG(ps.Total) AS Avg_Total
    FROM CUSTOMER c
    INNER JOIN PRODUCT_SALES ps ON c.CustomerID = ps.CustomerID
    GROUP BY c.CustomerID, c.CustomerName
)
SELECT ca.CustomerID, ca.CustomerName, ca.Avg_Total
FROM CustomerAvg ca
INNER JOIN AvgOrder ao ON ca.Avg_Total > ao.OverallAvg;
```

Results | Messages

	CustomerID	CustomerName	Avg_Total
1	7	George, Sally	23.950000
2	11	Tyler, Jenny	31.132000

c)

```sql
--c
WITH ProductSalesWithDate AS (
    SELECT ps.CustomerID, p.ProductNumber, p.ProductName, t.Date,
           LAG(t.Date) OVER (PARTITION BY ps.CustomerID, p.ProductNumber ORDER BY t.Date) AS End_Date
    FROM PRODUCT_SALES ps
    INNER JOIN PRODUCT p ON ps.ProductNumber = p.ProductNumber
    INNER JOIN TIMELINE t ON ps.TimeID = t.TimeID
), DaysBetweenProductSales AS (
    SELECT CustomerID, ProductNumber, ProductName, Date, End_Date,
           DATEDIFF(day, End_Date, Date) AS Days_between_Product_Sales
    FROM ProductSalesWithDate
)
SELECT c.CustomerID, c.CustomerName, dbps.ProductNumber, dbps.ProductName, dbps.Date, dbps.End_Date, dbps.Days_between_Product_Sales
FROM CUSTOMER c
INNER JOIN DaysBetweenProductSales dbps ON c.CustomerID = dbps.CustomerID
ORDER BY c.CustomerID;
```

Results | Messages

	CustomerID	CustomerName	ProductNumber	ProductName	Date	End_Date	Days_between_Product_Sales
1	1	Jacobs, Nancy	BK001	Kitchen Remodeling Basics For Everyone	2018-04-08	NULL	NULL
2	1	Jacobs, Nancy	VB001	Kitchen Remodeling Basics	2018-04-08	NULL	NULL
3	1	Jacobs, Nancy	VK001	Kitchen Remodeling Basics	2018-04-08	NULL	NULL
4	3	Able, Ralph	BK001	Kitchen Remodeling Basics For Everyone	2018-04-23	NULL	NULL
5	3	Able, Ralph	BK002	Advanced Kitchen Remodeling For Ever...	2018-06-05	NULL	NULL
6	3	Able, Ralph	VB001	Kitchen Remodeling Basics	2017-10-15	NULL	NULL
7	3	Able, Ralph	VB001	Kitchen Remodeling Basics	2018-06-05	2017-10...	233
8	3	Able, Ralph	VB002	Advanced Kitchen Remodeling I	2018-06-05	NULL	NULL

Query executed successfully.　(local) (14.0 RTM)　DESKTOP-STQVT9U\Dell (61)　Sales　00:00:00　48 rows

d)

```sql
--d
SELECT Year, QuarterText, SUM(Total) AS Total_Sales
FROM PRODUCT_SALES ps
INNER JOIN TIMELINE t ON ps.TimeID = t.TimeID
GROUP BY ROLLUP(Year, QuarterText);
```

Results | Messages

	Year	QuarterText	Total_Sales
1	2017	Qtr3	95.78
2	2017	NULL	95.78
3	2018	Qtr1	302.33
4	2018	Qtr2	542.78
5	2018	NULL	845.11
6	NULL	NULL	940.89

Query executed successfully.　(local) (14.0 RTM)　DESKTOP-STQVT9U\Dell (61)　Sales　00:00:00　6 rows

3-

```
fact = read.csv("customer_churn.csv")

revenue_cube <-
  tapply(fact$Total.Revenue,
         fact[,c("Contract", "offer", "Internet.Type","Customer.Status")],
         FUN=function(x){return(sum(x))})


sub_data =apply(revenue_cube, c("Contract", "offer","Internet.Type"),
                FUN=function(x) {return(sum(x, na.rm=TRUE))})

sub_data["Two Year" , , ]
```

```
            Internet.Type
 Offer           Cable          DSL Fiber Optic
   None       554678.24 1235674.43  2075682.41
   offer A 349750.71   720759.09  1474862.52
   offer B 164188.21   362671.78   473681.65
   offer C  10360.26    34148.92    80196.94
   offer D   5443.22     5592.51    22464.25
   offer E   1684.86     3019.54     6134.10
```

```
total = sum(revenue_cube[,"Offer B",,],na.rm = TRUE)
churned = revenue_cube["Month-to-Month","Offer B","Cable","Churned"]
churned/total *100
```

```
[1] 0.4511609
>
```