CS779
DATA WAREHOUSE ON WEATHER&TRAVEL

Yuandi TANG U65674688

RESEARCH QUESTION

RESEARCH SCENARIO

Safety: Weather conditions can have a significant impact on aviation safety. Studying weather data can help airlines and aviation authorities predict and prepare for potential weather-related hazards such as turbulence, thunderstorms, or icing conditions.

Efficiency: Analyzing airplane data can help airlines and aviation authorities identify patterns and trends that can lead to improvements in fuel efficiency, flight times, and overall operational efficiency.

RESEARCH SCENARIO

Environmental impact: Researching weather and airplane data can also help to mitigate the environmental impact of aviation by reducing fuel consumption and emissions.

Improving technology: Studying weather and airplane data can help to identify areas for technological advancements and improvements in aircraft design and performance.

TOPICS

1. REQUIRMENTS
2. ETL
3. DATA QUALITY
4. ERD
5. SECURITY & COMPLIANCE
6. BACKUP & RECOVERY
7. TUNING & OPTIMIZATION
8. VISUAL & ANALYSIS
9. COMM THEORIES

1.REQUIRMENTS

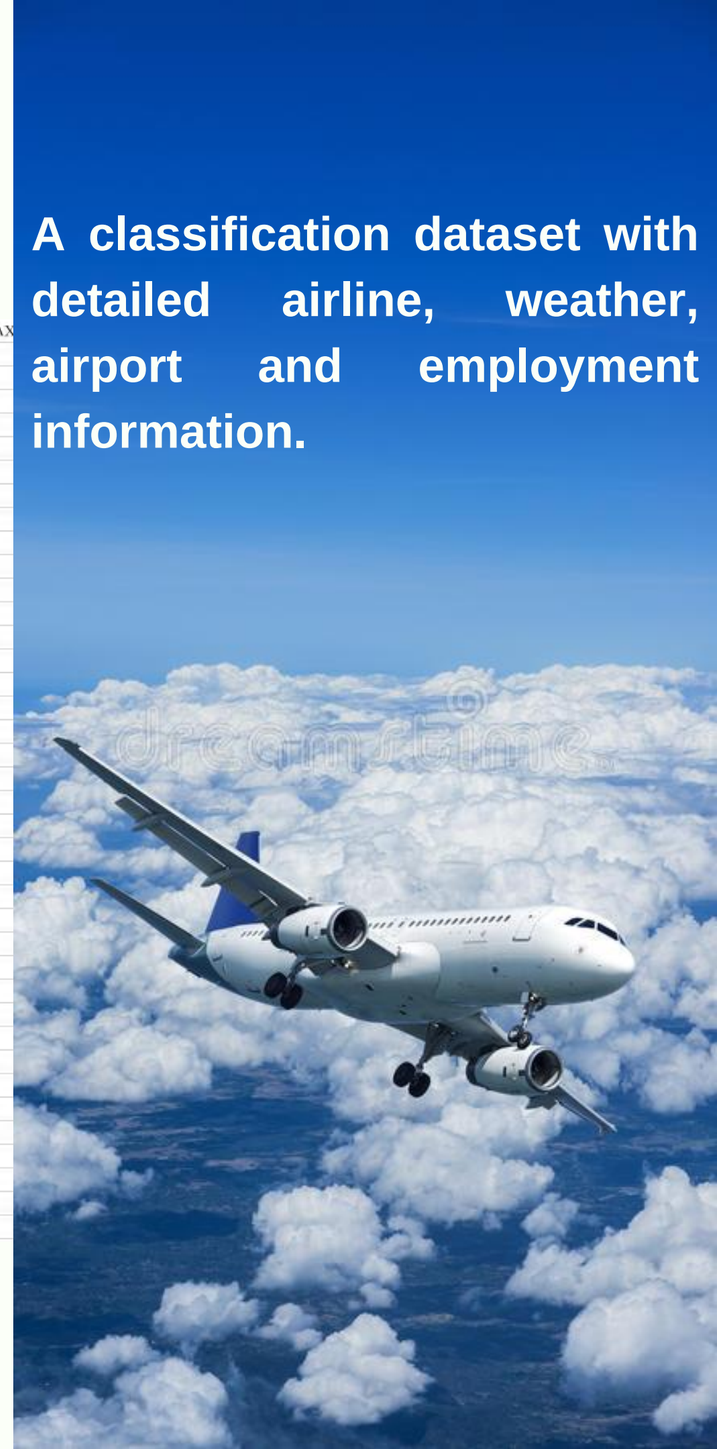
RESEARCH QUESTION

RAW DATASET

2019 Flight Delays w/Weather and Airport

Y_OF_W	DEP_DEL	DEP_TIME	DISTANCE	SEGMENT	CONCURR	NUMBER_	CARRIER_	AIRPORT_	AIRLINE_	AIRLINE_	AVG_MON	AVG_MON	FLT_ATTE	GROUND_	PLANE_AG	DEPARTIN	LATITUDE	LONGITU	PREVIOUS	PRCP	SNOW	SNWD	TMAX
7	0	0800-0859	2	1	25	143	Southwest A	13056	107363	5873	1903352	13382999	6.18E-05	9.89E-05	8	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0700-0759	7	1	29	191	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	7	1	27	199	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	18	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	9	1	27	180	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	2	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0001-0559	7	1	10	182	Spirit Air L	13056	15023	1257	1903352	2688839	9.17E-06	0.00012465	1	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0001-0559	3	1	10	180	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	5	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0700-0759	6	1	29	186	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	2	McCarran I	36.08	-115.152	NONE	0	0	0	
7	1	0001-0559	7	1	10	186	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0001-0559	7	1	10	180	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	8	1	27	186	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	1	McCarran I	36.08	-115.152	NONE	0	0	0	
7	1	2300-2359	6	1	17	180	Frontier Ai	13056	9496	581	1903352	1857122	0.00011573	7.13E-06	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	1	1	27	149	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	1200-1259	1	1	26	119	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	12	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	2	1	27	146	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	7	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	4	1	27	181	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	2	McCarran I	36.08	-115.152	NONE	0	0	0	
7	1	0700-0759	4	1	29	181	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	4	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0800-0859	4	1	25	181	Alaska Airl	13056	20315	717	1903352	2884187	3.23E-05	0.0001746	4	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0001-0559	11	1	10	294	Hawaiian A	13056	6791	80	1903352	905990	0.00012049	0.00019785	6	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0900-0959	11	1	28	294	Hawaiian A	13056	6791	80	1903352	905990	0.00012049	0.00019785	5	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0700-0759	1	1	29	181	American A	13056	75506	1174	1903352	11744595	9.82E-05	0.00017729	5	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	8	1	27	187	American A	13056	75506	1174	1903352	11744595	9.82E-05	0.00017729	18	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0800-0859	2	1	25	173	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	6	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0800-0859	3	1	25	142	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	22	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0900-0959	7	1	28	154	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	1	1000-1059	3	1	29	142	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	19	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	7	1	27	173	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	4	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0700-0759	1	1	29	154	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	21	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	2	1	27	154	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	20	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	9	1	27	173	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	11	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0001-0559	5	1	10	154	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	3	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	3	1	27	173	United Air	13056	46218	1108	1903352	8501631	0.0002538	0.00022899	4	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	1000-1059	2	1	29	180	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	6	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	2	1	27	110	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	19	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0700-0759	4	1	29	110	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	19	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0900-0959	6	1	28	180	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	1	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	1	1	27	160	Delta Air L	13056	73508	1174	1903352	12460183	0.00014417	0.00014866	18	McCarran I	36.08	-115.152	NONE	0	0	0	
7	1	2200-2259	9	1	9	162	JetBlue Airv	13056	23463	299	1903352	3190369	0.00016004	0.00012687	15	McCarran I	36.08	-115.152	NONE	0	0	0	
7	0	0600-0659	9	1	27	129	JetBlue Airv	13056	23463	299	1903352	3190369	0.00016004	0.00012687	11	McCarran I	36.08	-115.152	NONE	0	0	0	

A classification dataset with detailed airline, weather, airport and employment information.



Travel Dataset - Datathon 2019

A synthetic dataset of corporate travels

travelCode	userCode	name	place	days	price	total	date
0	0	Hotel A	Florianopo	4	313.02	1252.08	09/26/2019
2	0	Hotel K	Salvador (B	2	263.41	526.82	10/10/2019
7	0	Hotel K	Salvador (B	3	263.41	790.23	11/14/2019
11	0	Hotel K	Salvador (B	4	263.41	1053.64	12/12/2019
13	0	Hotel A	Florianopo	1	313.02	313.02	12/26/2019
15	0	Hotel BD	Natal (RN)	2	242.88	485.76	01/09/2020
22	0	Hotel Z	Aracaju (SE	2	208.04	416.08	02/27/2020
29	0	Hotel					
32	0	Hotel					
33	0	Hotel					
34	0	Hotel					
38	0	Hotel					
39	0	Hotel					

hotel data

travelCode	userCode	locfrom	locto	flightType	price	time	distance	agency	date
0	0	Recife (PE)	Florianopo	firstClass	1434.38	1.76	676.53	FlyingDrop	09/26/2019
0	0	Florianopo	Recife (PE)	firstClass	1292.29	1.76	676.53	FlyingDrop	09/30/2019
1	0	Brasilia (DI	Florianopo	firstClass	1487.52	1.66	637.56	CloudFy	10/03/2019
1	0	Florianopo	Brasilia (DI	firstClass	1127.36	1.66	637.56	CloudFy	10/04/2019
2	0	Aracaju (SE	Salvador (B	firstClass	1684.05	2.16	830.86	CloudFy	10/10/2019
2	0	Salvador (B	Aracaju (SE	firstClass	1531.92	2.16	830.86	CloudFy	10/12/2019
3	0	Aracaju (SE	Campo						
3	0	Campo Gra	Aracaju						
4	0	Recife (PE)	Florian						
4	0	Florianopo	Recife (						
5	0	Brasilia (DI	Aracaju						
5	0	Aracaju (SE	Brasilia						
6	0	Recife (PE)	Florian						

Flight data

code	company	name	gender	age
0	4You	Roy Braun	male	21
1	4You	Joseph Hols	male	37
2	4You	Wilma Mci	female	48
3	4You	Paula Dani	female	23
4	4You	Patricia Ca	female	44
5	4You	Trina Thom	none	47
6	4You	Jesse Decell	male	46
7	4You	Gregoria G	female	21
8	4You	Jack Sabo	none	41
9	4You	Debbie Hel	none	35
10	4You	Melvin Lov	male	36
11	4You	Virginia Ro	female	61
12	4You	David Thor	male	53

user data

DATA AUDIT/USER STORY

*The **user table** is mocked clients identification, it is **not relevant** to the research scenario and question. So it will not be upload and process in the following.*

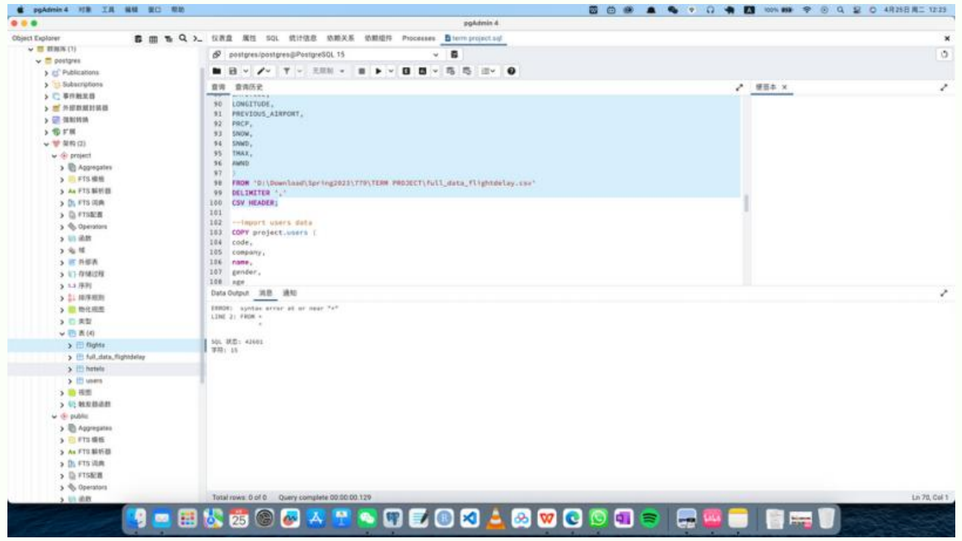
code	company	name	gender	age	
0	4You	Roy Braun	male	21	user data
1	4You	Joseph Hols	male	37	
2	4Yo	Wilma Mci	female	48	
3	4You	Paula Dani	female	23	
4	4You	Pat a Ca	female	44	
5	4You	Trina Tho	none	47	
6	4You	Jesse Decell	mal	46	
7	4You	Gregoria G	female	21	
8	4You	Jack Sabo	none	41	
9	4You	Debbie Hel	none	3	
10	4You	Melvin Lov	male	36	
11	4You	Virginia Ro	female	61	
12	4You	David Thor	male	53	

2.ETL

LOAD THE DATA

LOADING

```
COPY  project.full_data_flightdelay (
MONTH,
DAY_OF_WEEK,
DEP_DEL15,
DEP_TIME_BLK,
DISTANCE_GROUP,
SEGMENT_NUMBER,
CONCURRENT_FLIGHTS,
NUMBER_OF_SEATS,
CARRIER_NAME,
AIRPORT_FLIGHTS_MONTH,
AIRLINE_FLIGHTS_MONTH,
AIRLINE_AIRPORT_FLIGHTS_MONTH,
AVG_MONTHLY_PASS_AIRPORT,
AVG_MONTHLY_PASS_AIRLINE,
FLT_ATTENDANTS_PER_PASS,
GROUND_SERV_PER_PASS,
PLANE_AGE,
DEPARTING_AIRPORT,
LATITUDE,
LONGITUDE,
PREVIOUS_AIRPORT,
PRCP,
SNOW,
SNWD,
TMAX,
AWND
)
FROM 'D:\Download\Spring2023\779\TERM PROJECT\full_data_flightdelay.csv'
DELIMITER ','
CSV HEADER;
```



CLEAN/TRANSFORM/FILTER

The data is clean, without anomalies, and well structured.

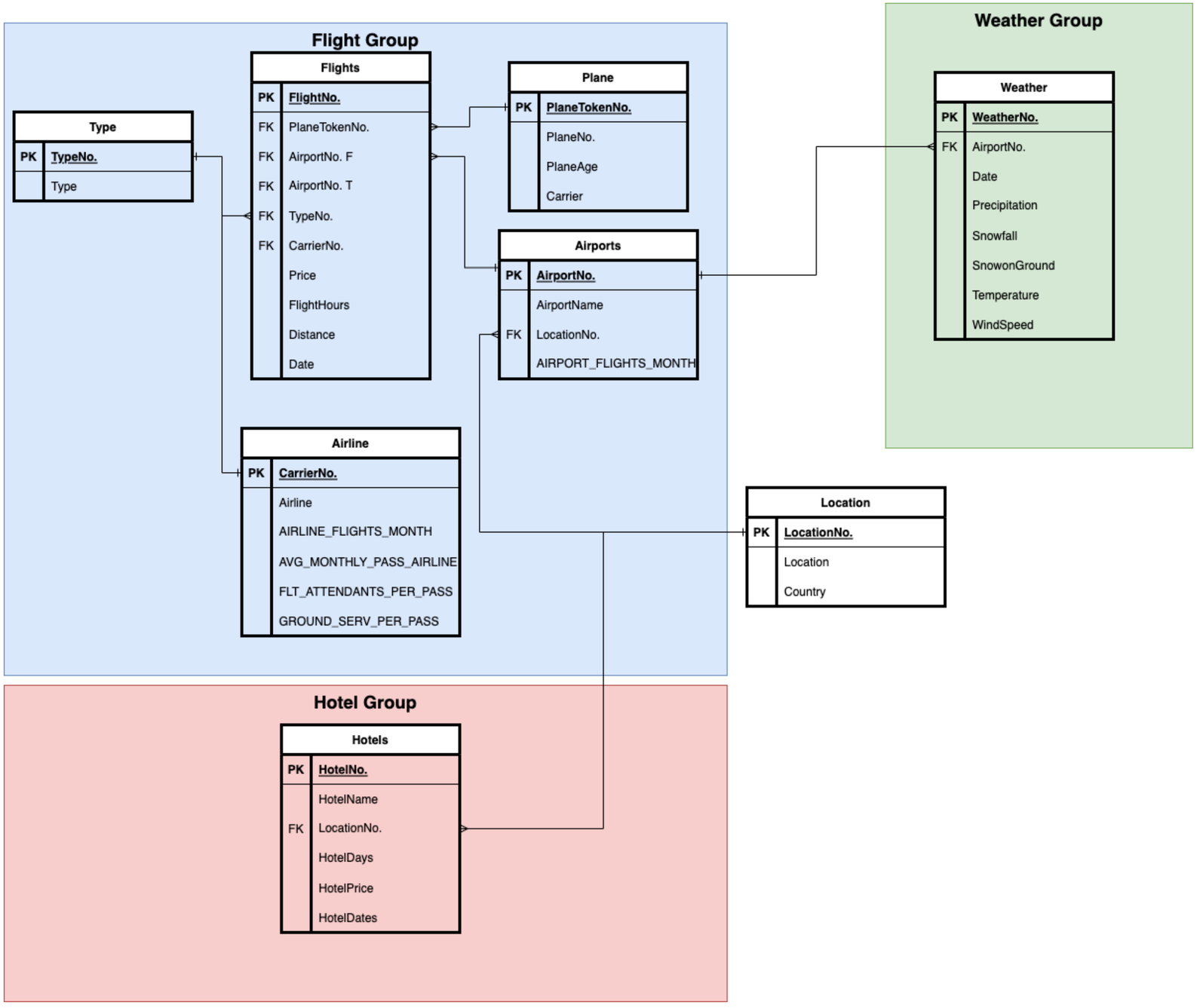
The data format is audited as follows:

- ***all date DATE***
- ***all time TIME***
- **flight number/weather report NUMERIC**
- **flight name/hotel name/name VARCHAR(25)**
- **PK SERIAL**

ALTER TABLE Project.flights
ALTER COLUMN date TYPE date USING date::date,

ALTER TABLE Project.hotels
ALTER COLUMN date TYPE date USING date::date;

3.ERD


Flight Group
Flights
PK FlightNo.
FK PlaneTokenNo.
FK AirportNo. F
FK AirportNo. T
FK TypeNo.
FK CarrierNo.
Price
FlightHours
Distance
Date
Type
PK TypeNo.
Type
Plane
PK PlaneTokenNo.
PlaneNo.
PlaneAge
Carrier
Airports
PK AirportNo.
AirportName
FK LocationNo.
AIRPORT_FLIGHTS_MONTH
Airline
PK CarrierNo.
Airline
AIRLINE_FLIGHTS_MONTH
AVG_MONTHLY_PASS_AIRLINE
FLT_ATTENDANTS_PER_PASS
GROUND_SERV_PER_PASS
Weather Group
Weather
PK WeatherNo.
FK AirportNo.
Date
Precipitation
Snowfall
SnowonGround
Temperature
WindSpeed
Location
PK LocationNo.
Location
Country
Hotel Group
Hotels
PK HotelNo.
HotelName
FK LocationNo.
HotelDays
HotelPrice
HotelDates

CREATING TABLES

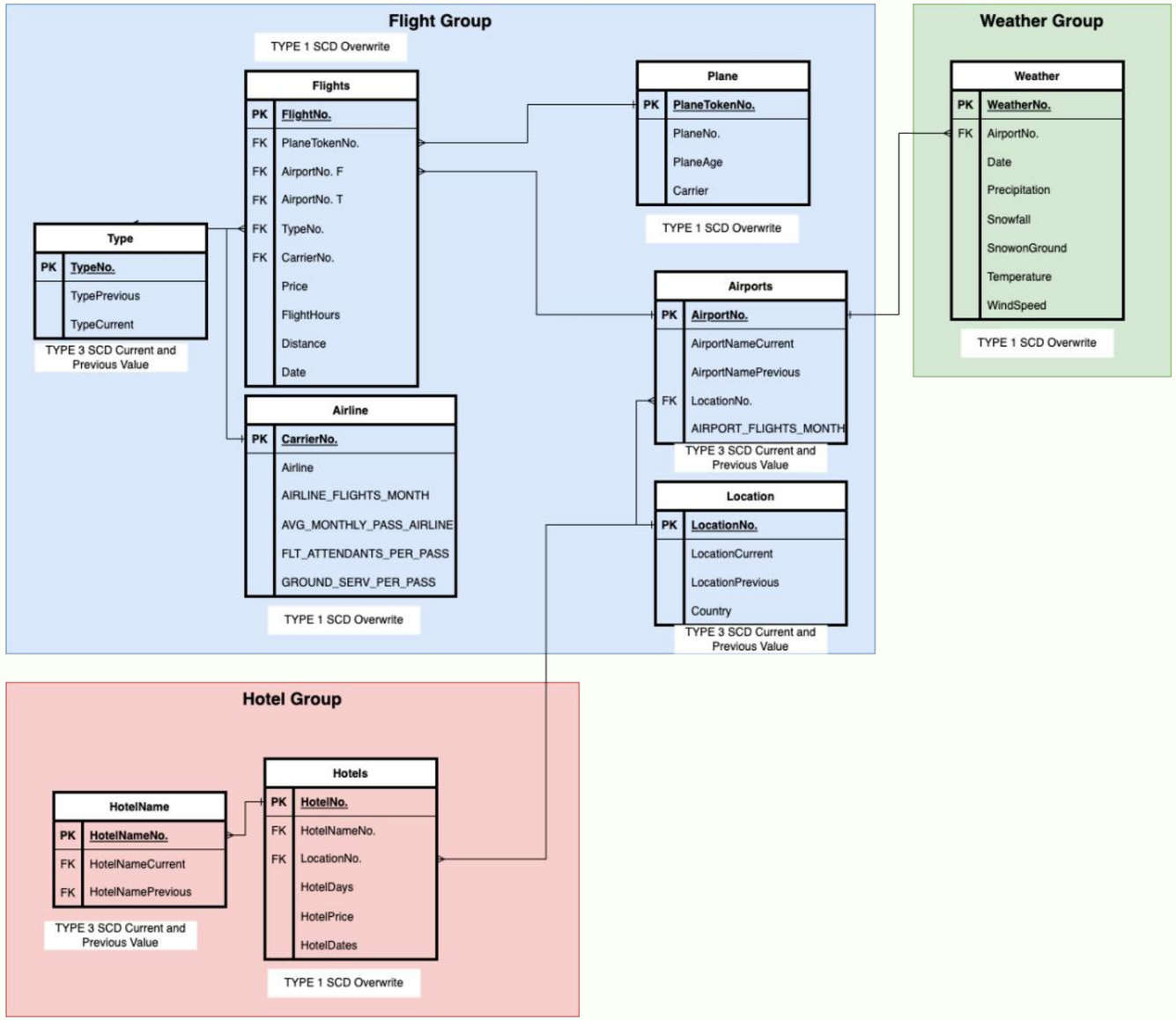
```
CREATE TABLE project.type (
    TypeNo SERIAL PRIMARY KEY,
    TypePrevious varchar(25),
 TypeCurrent varchar(25)
);

INSERT INTO project.type (TypeCurrent)
SELECT DISTINCT project.f.flightType
FROM project.f
```

BCNF

Flights	
PK	**FlightNo.**
FK	PlaneTokenNo.
FK	AirportNo. F
FK	AirportNo. T
FK	TypeNo.
FK	CarrierNo.
	Price
	FlightHours
	Distance
	Date

Flights	
PK	**FlightNo.**
FK	**PlaneTokenNo. AA11-1808--AA11-1809**
FK	**AirportNo. F 001-002**
FK	**AirportNo. T 003-004**
FK	**TypeNo. Boeing-Airbus**
FK	**CarrierNo. American Airlines-Delta Airlines**
	Price 150-180
	FlightHours 1.5-1.67
	Distance
	Date

SCDs


Flight Group
TYPE 1 SCD Overwrite
Flights
PK FlightNo.
FK PlaneTokenNo.
FK AirportNo. F
FK AirportNo. T
FK TypeNo.
FK CarrierNo.
Price
FlightHours
Distance
Date
Plane
PK PlaneTokenNo.
PlaneNo.
PlaneAge
Carrier
TYPE 1 SCD Overwrite
Type
PK TypeNo.
TypePrevious
TypeCurrent
TYPE 3 SCD Current and Previous Value
Airline
PK CarrierNo.
Airline
AIRLINE_FLIGHTS_MONTH
AVG_MONTHLY_PASS_AIRLINE
FLT_ATTENDANTS_PER_PASS
GROUND_SERV_PER_PASS
TYPE 1 SCD Overwrite
Airports
PK AirportNo.
AirportNameCurrent
AirportNamePrevious
FK LocationNo.
AIRPORT_FLIGHTS_MONTH
TYPE 3 SCD Current and Previous Value
Location
PK LocationNo.
LocationCurrent
LocationPrevious
Country
TYPE 3 SCD Current and Previous Value
Weather Group
Weather
PK WeatherNo.
FK AirportNo.
Date
Precipitation
Snowfall
SnowonGround
Temperature
WindSpeed
TYPE 1 SCD Overwrite
Hotel Group
HotelName
PK HotelNameNo.
FK HotelNameCurrent
FK HotelNamePrevious
TYPE 3 SCD Current and Previous Value
Hotels
PK HotelNo.
FK HotelNameNo.
FK LocationNo.
HotelDays
HotelPrice
HotelDates
TYPE 1 SCD Overwrite


dreamstime

4.SECURITY COMPLIANCE

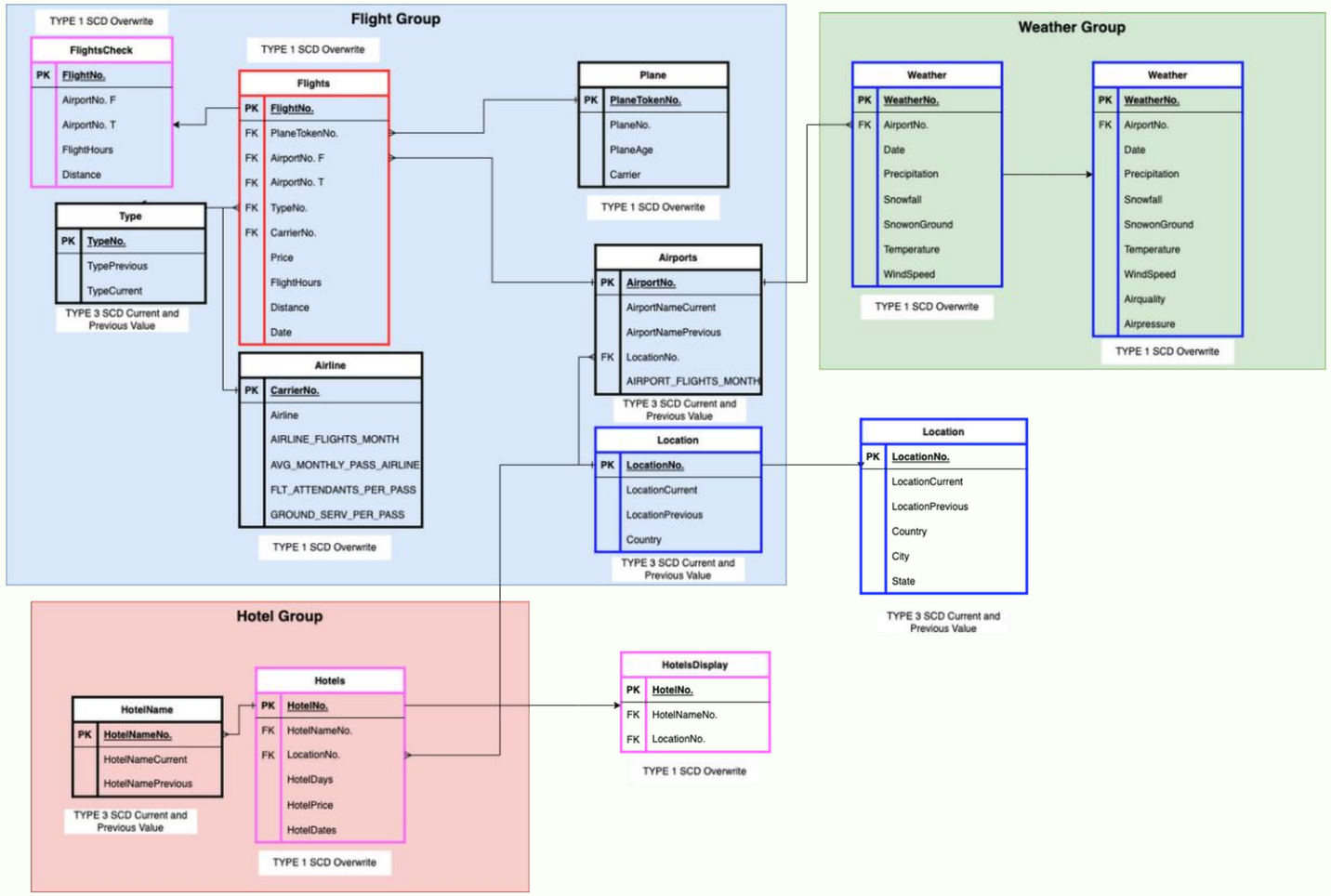
GDPR & HIPAA

- ***The travel data does not have confidential data on medical records or personal privacy information.***

- ***The dataset is GDPR & HIPAA-compliant.***

5.BACKUP AND RECOVERY

PARTITION



Red: Horizontal Partitioning (Shows local data for specific regions)

Blue: Vertical Partitioning(Modify attributes for regional local uses)

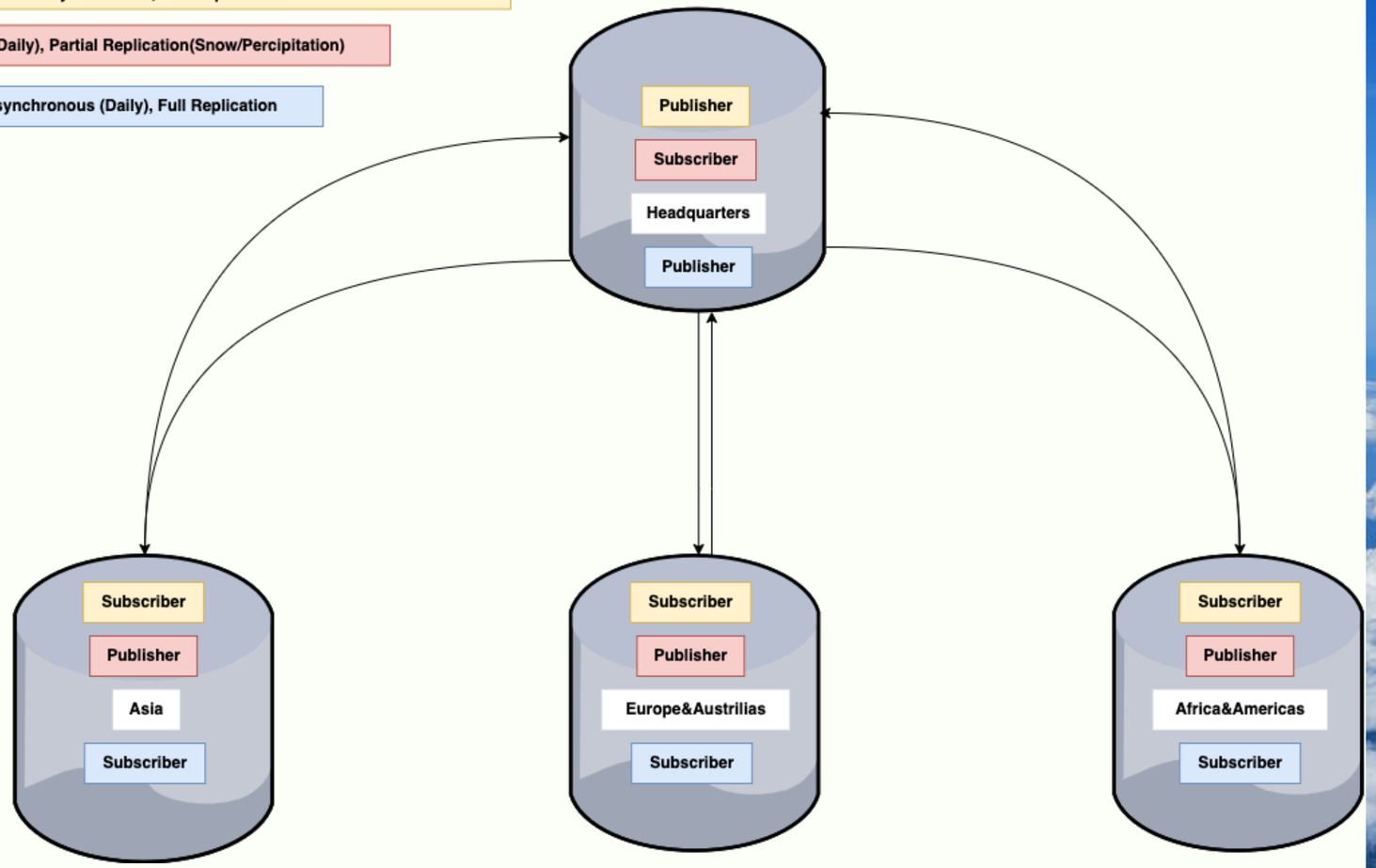
Pink: Vertical Partitioning(Show different tables for different department uses)

SYNCHRONIZATION

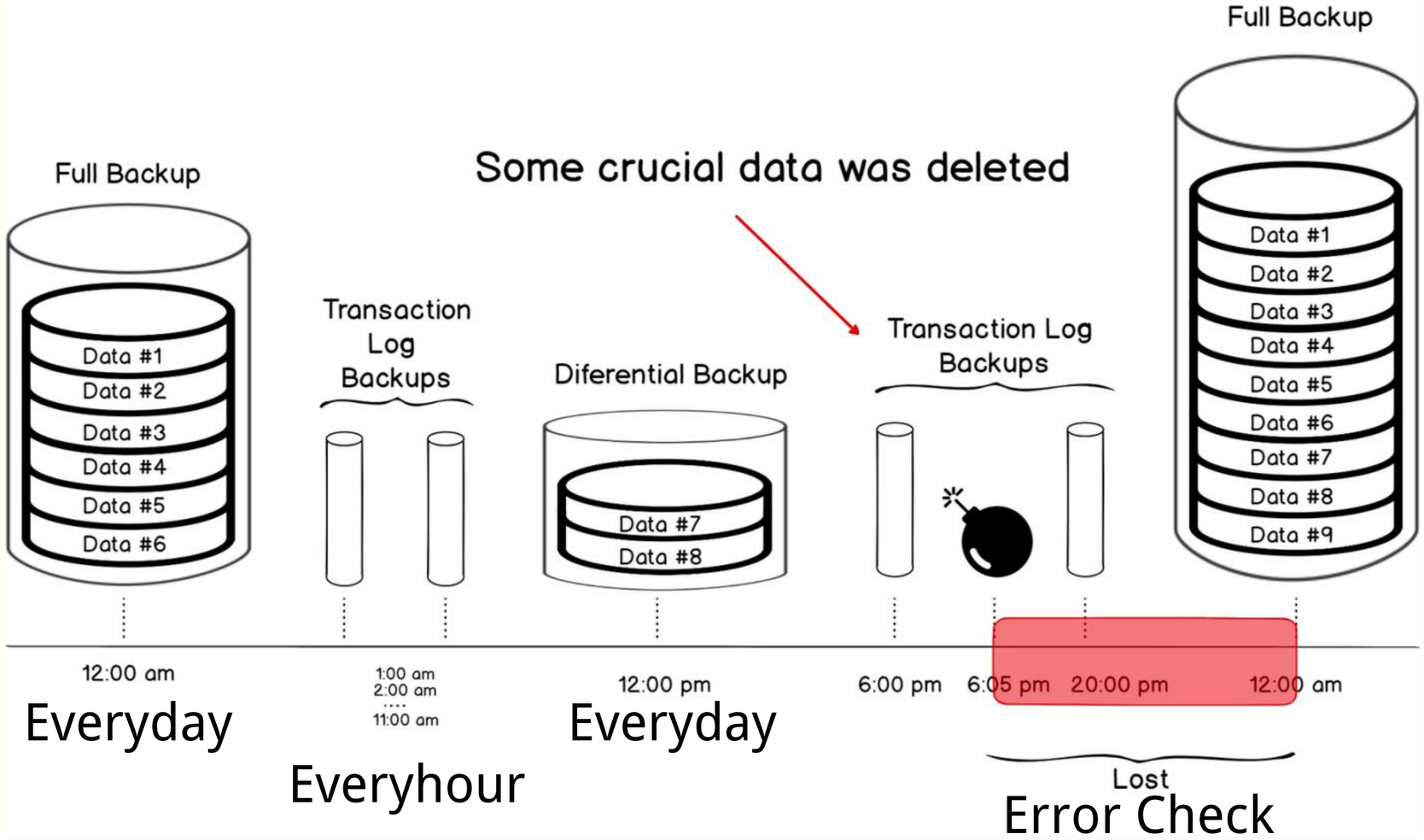
Hotel/Hotel Display/Flight Check Table: Synchronous, Partial Replication For HQ(Display)
Flight Table: Synchronous, Full Replication

Weather Table: Synchronous (Daily), Partial Replication(Snow/Percipitation)

Type/Airline/Location Table: Asynchronous (Daily), Full Replication



RECOVERY PLANS



https://www.sqlshack.com/sql-server-transaction-log-and-recovery-models/

DDT CHOICES

	Amazon Redshift	Microsoft Azure Synapse	Google BigQuery	Snowflake Cloud Data Platform
Initial Release	2012	2016	2010	2014
Separates Storage and Compute	No	Yes	Yes	Yes
Multi-Cloud	No	No	No	Yes ✓
Query Language	Amazon Redshift SQL	TSQL	Standard SQL 2011 & BigQuery SQL	Snowflake SQL
Elasticity	Yes - Manual	Yes – Manual and Automatic	Yes – Automatic	Yes – Automatic ✓
MPP	Yes	Yes	Yes	Yes
Columnar	Yes	Yes	Yes	Yes
Foreign Keys	Yes	Yes	No	Yes
Transaction	ACID	ACID	ACID	ACID
Concurrency	Yes	Yes	Yes	Yes
Durability	Yes	Yes	Yes	Yes
Automation	No	No	No	No
Website	Link	Link	Link	Link
Free Trial	Yes	Yes	Yes	Yes

https://www.sqlshack.com/sql-server-transaction-log-and-recovery-models/

6.TUNING

HIGH AND LOW CARDINALITIES

- ***HIGH CARDINALITIES(a table has a large number of unique values)***
- Flight
- Hotels
- Weather

- ***LOW CARDINALITIES(a table has)***
- *Airlines*
- *Planes*

INDEXING-TYPE

- **Data Types: Bitmap and B-tree indexes can handle both numeric and character data types, while hash indexes are typically limited to numeric data types**

- **Cardinality: Bitmap indexes are effective for low cardinality columns, while B-tree and hash indexes are suitable for high cardinality columns.**

- **Size and Storage: Bitmap indexes require less storage space compared to B-tree and hash indexes. However, B-tree indexes are more flexible and can handle large data sets. Hash indexes are typically faster than B-tree and bitmap indexes for exact match queries, but they are less flexible and less efficient for range queries.**

- **Query Performance: Bitmap indexes are useful for boolean operations such as AND, OR, and NOT, while B-tree and hash indexes are better suited for range queries and exact match lookups, respectively.**

- **Maintenance: B-tree and hash indexes are easier to maintain and update compared to bitmap indexes, which can become complex to maintain as data sets grow in size.**

https://docs.oracle.com/database/121/CNCPT/indexiot.htm

INDEXING-IMPLEMENTATION

- **CREATE INDEX idx_weather2_hash ON project.weather USING btree(Snowfall);**
- **CREATE INDEX idx_weather3_hash ON project.weather USING btree(SnowonGround);**
- **CREATE INDEX idx_weather4_hash ON project.weather USING btree(Temperature);**
- **CREATE INDEX idx_weather5_hash ON project.weather USING btree(WindSpeed)**

https://docs.oracle.com/database/121/CNCPT/indexiot.htm

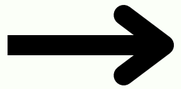
INDEXING-IMPLEMENTATION

查询 查询历史

```
300 --Tuning
301 --explain a concat query before indexing
302 EXPLAIN SELECT * FROM project.weather
303 WHERE Precipitation >2 AND SNOWFALL<5
304 ORDER BY precipitation DESC;
305 --create a indexing
306 CREATE INDEX idx_weather1_hash ON project.weather USING btree
307 CREATE INDEX idx_weather2_hash ON project.weather USING btree
308 CREATE INDEX idx_weather3_hash ON project.weather USING btree
309 CREATE INDEX idx_weather4_hash ON project.weather USING btree
310 CREATE INDEX idx_weather5_hash ON project.weather USING btree
311 --explain on it after indexing
312 EXPLAIN SELECT * FROM project.weather
313 WHERE Precipitation >2 AND SNOWFALL<5
314 ORDER BY precipitation DESC;
315
```

Data Output 消息 通知

	QUERY PLAN text
1	Gather Merge (cost=349555.50..409261.03 rows=511726 width=292)
2	Workers Planned: 2
3	-> Sort (cost=348555.47..349195.13 rows=255863 width=292)
4	Sort Key: precipitation DESC
5	-> Parallel Seq Scan on weathe (cost=0.00..255607.56 rows=255863 width=2...
6	Filter: ((precipitation > '2'::numeric) AND (snowfall < '5'::numeric))



查询 查询历史

```
309 CREATE INDEX idx_weather4_hash ON project.weather USING btree(Temperature);
310 CREATE INDEX idx_weather5_hash ON project.weather USING btree(WindSpeed);
311 --explain on it after indexing
312 EXPLAIN SELECT * FROM project.weather
313 WHERE Precipitation >2 AND SNOWFALL<5
314 ORDER BY precipitation DESC;
315
316 --trigger prevent deletion
317 CREATE OR REPLACE FUNCTION prevent_airline_deletion()
318 RETURNS trigger AS
319 $$
320 BEGIN
321   RAISE EXCEPTION 'Deletion of records in the airline table is not allowed'
322   RETURN NULL;
323 END;
```

Data Output 消息 通知

	QUERY PLAN text
1	Index Scan Backward using idx_weather1_hash on weather (cost=0.56..240323.67 rows=92054 width=1...
2	Index Cond: (precipitation > '2'::numeric)
3	Filter: (snowfall < '5'::numeric)

https://docs.oracle.com/database/121/CNCPT/indexiot.htm

INDEXING-IMPROVEMENTS

- Index Combination: In some cases, creating a combined index on multiple columns can be more effective than creating separate indexes on each individual column. This can be particularly useful if queries often filter on multiple columns. For example, you might create an index on instead of separate indexes on and . **(Temperature, Precipitation)TemperaturePrecipitation**
- Index Type Selection: As mentioned before, the hash index may be more efficient for columns with a small number of distinct values. For example, if has only a few distinct values, it may be more efficient to create a hash index on this column.**Airquality**

https://docs.oracle.com/database/121/CNCPT/indexiot.htm

INDEXING-IMPROVEMENTS

- Index Maintenance: Regularly maintaining the indexes can help to ensure their efficiency. This can include reindexing the table to eliminate fragmentation, or updating the statistics to help the optimizer choose the best execution plan.
- Query Optimization: Optimizing the queries themselves can often have a greater impact on performance than simply creating indexes. Reviewing query execution plans and identifying areas for improvement, such as reducing unnecessary joins or subqueries, can help to improve overall performance.

https://docs.oracle.com/database/121/CNCPT/indexiot.htm

TRIGGERS

Prevent deletion on airline table

```
CREATE OR REPLACE FUNCTION prevent_airline_deletion()
RETURNS trigger AS
$$
BEGIN
  RAISE EXCEPTION 'Deletion of records in the airline table is not allowed';
  RETURN NULL;
END;
$$
LANGUAGE plpgsql;

CREATE TRIGGER prevent_airline_deletion_trigger
BEFORE DELETE ON project.airline
FOR EACH ROW
EXECUTE FUNCTION prevent_airline_deletion();
```

TRIGGERS-IMPLEMENTATION

查询　查询历史

```
321  END;
322  $$
323  LANGUAGE plpgsql;
324
325  CREATE TRIGGER prevent_airline_deletion_trigger
326  BEFORE DELETE ON project.airline
327  FOR EACH ROW
328  EXECUTE FUNCTION prevent_airline_deletion();
329
330  delete from project.airline
331  where carrierno =1
332
333  --stored procedure
334  CREATE OR REPLACE PROCEDURE update_airport_name(
335      airportno integer,
```

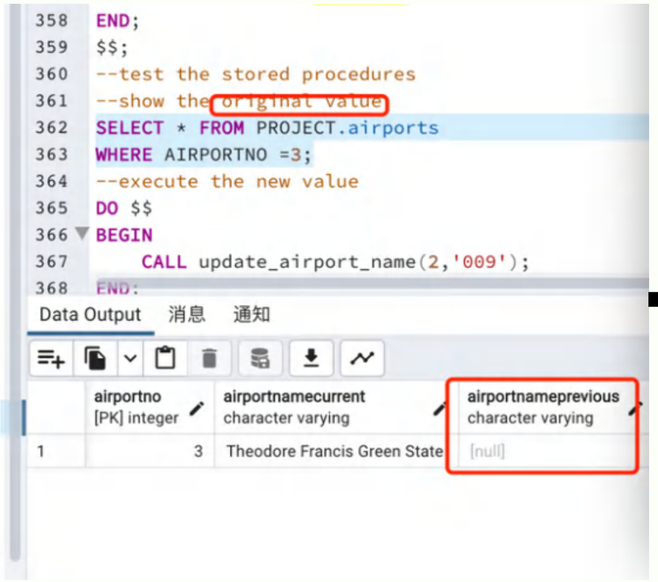
Data Output　消息　通知

```
ERROR:  Deletion of records in the airline table is not allowed
CONTEXT:  PL/pgSQL function prevent_airline_deletion() line 3 at RAISE

SQL 状态: P0001
```

STORED PROCEDURES

```
CREATE OR REPLACE PROCEDURE update_hotel_name(
    hotelno integer,
    new_name varchar(25)
)
LANGUAGE plpgsql
AS $$
DECLARE
    old_name varchar(25);
BEGIN
    -- Get the current value of the "HotelNameCurrent" attribute
    SELECT HotelNameCurrent INTO old_name FROM project.HotelName WHERE HotelNameNo = hotelno;

    -- If the new value is the same as the old value, do nothing
    IF old_name = new_name THEN
        RETURN;
    END IF;

    -- Otherwise, update the "HotelNamePrevious" attribute with the old value
    UPDATE project.HotelName SET HotelNamePrevious = old_name WHERE HotelNameNo = hotelno;

    -- Update the "HotelNameCurrent" attribute with the new value
    UPDATE project.HotelName SET HotelNameCurrent = new_name WHERE HotelNameNo = hotelno;
END;
$$;
```

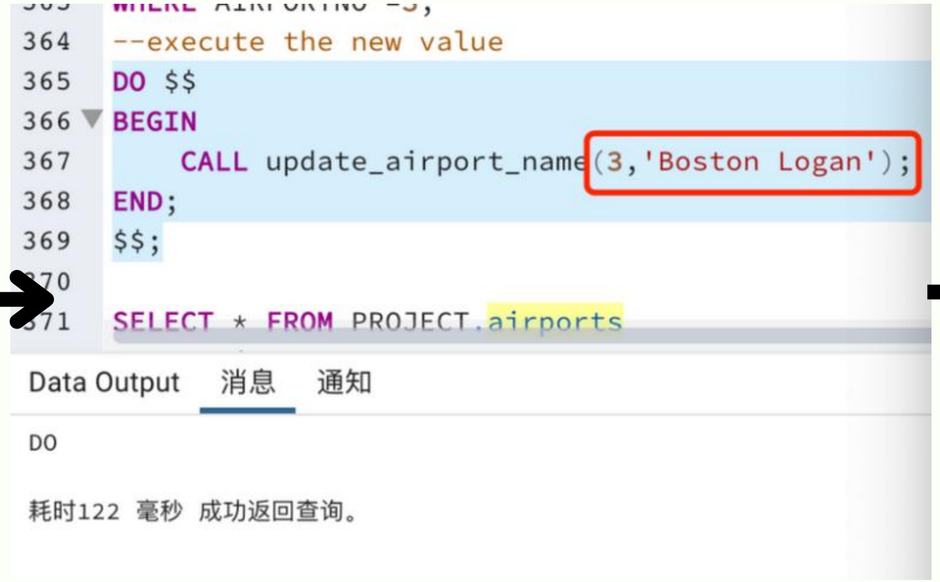
STORED PROCEDURES-RENEW



```
358 END;
359 $$;
360 --test the stored procedures
361 --show the original value
362 SELECT * FROM PROJECT.airports
363 WHERE AIRPORTNO =3;
364 --execute the new value
365 DO $$
366 BEGIN
367     CALL update_airport_name(2,'009');
368 END;
```

Data Output 消息 通知

	airportno [PK] integer	airportnamecurrent character varying	airportnameprevious character varying
1	3	Theodore Francis Green State	[null]

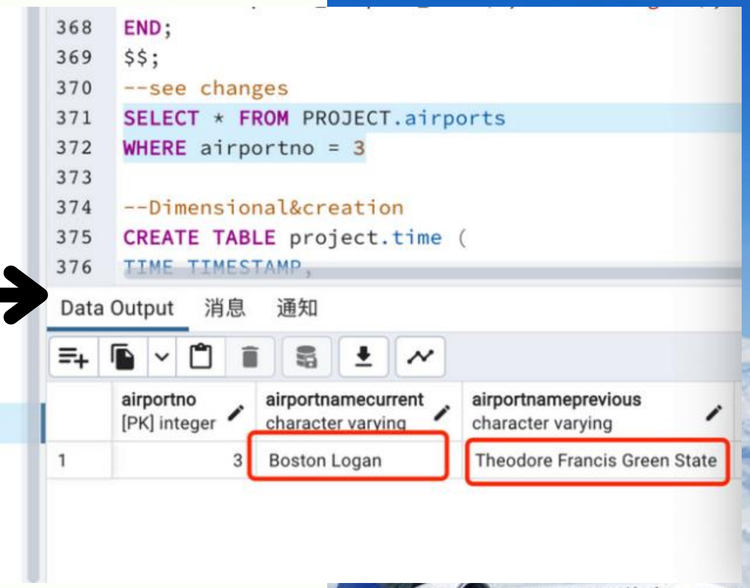


```
364 --execute the new value
365 DO $$
366 BEGIN
367     CALL update_airport_name(3,'Boston Logan');
368 END;
369 $$;
370
371 SELECT * FROM PROJECT.airports
```

Data Output 消息 通知

DO

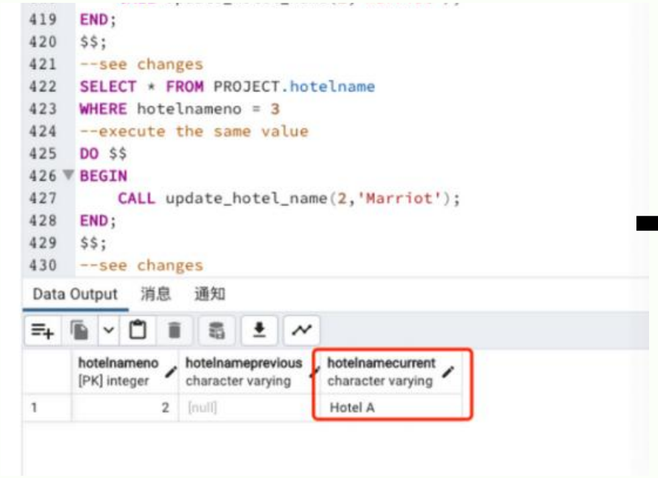
耗时122 毫秒 成功返回查询。



```
368 END;
369 $$;
370 --see changes
371 SELECT * FROM PROJECT.airports
372 WHERE airportno = 3
373
374 --Dimensional&creation
375 CREATE TABLE project.time (
376 TIME TIMESTAMP,
```

Data Output 消息 通知

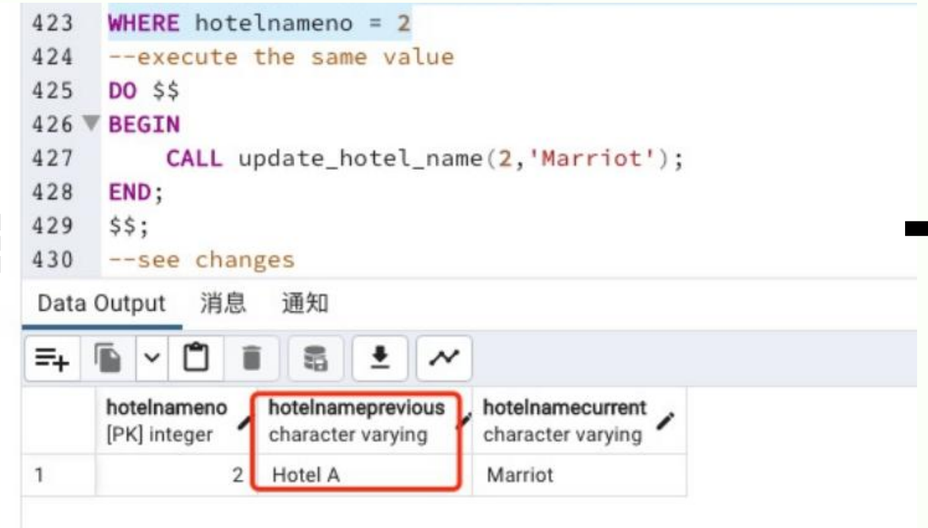
	airportno [PK] integer	airportnamecurrent character varying	airportnameprevious character varying
1	3	Boston Logan	Theodore Francis Green State



```
419 END;
420 $$;
421 --see changes
422 SELECT * FROM PROJECT.hotelname
423 WHERE hotelnameno = 3
424 --execute the same value
425 DO $$
426 BEGIN
427     CALL update_hotel_name(2,'Marriot');
428 END;
429 $$;
430 --see changes
```

Data Output 消息 通知

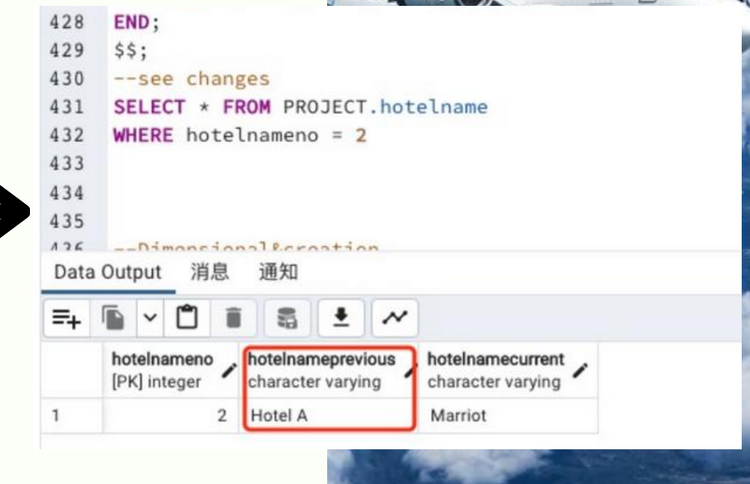
	hotelnameno [PK] integer	hotelnameprevious character varying	hotelnamecurrent character varying
1	2	[null]	Hotel A



```
423 WHERE hotelnameno = 2
424 --execute the same value
425 DO $$
426 BEGIN
427     CALL update_hotel_name(2,'Marriot');
428 END;
429 $$;
430 --see changes
```

Data Output 消息 通知

	hotelnameno [PK] integer	hotelnameprevious character varying	hotelnamecurrent character varying
1	2	Hotel A	Marriot



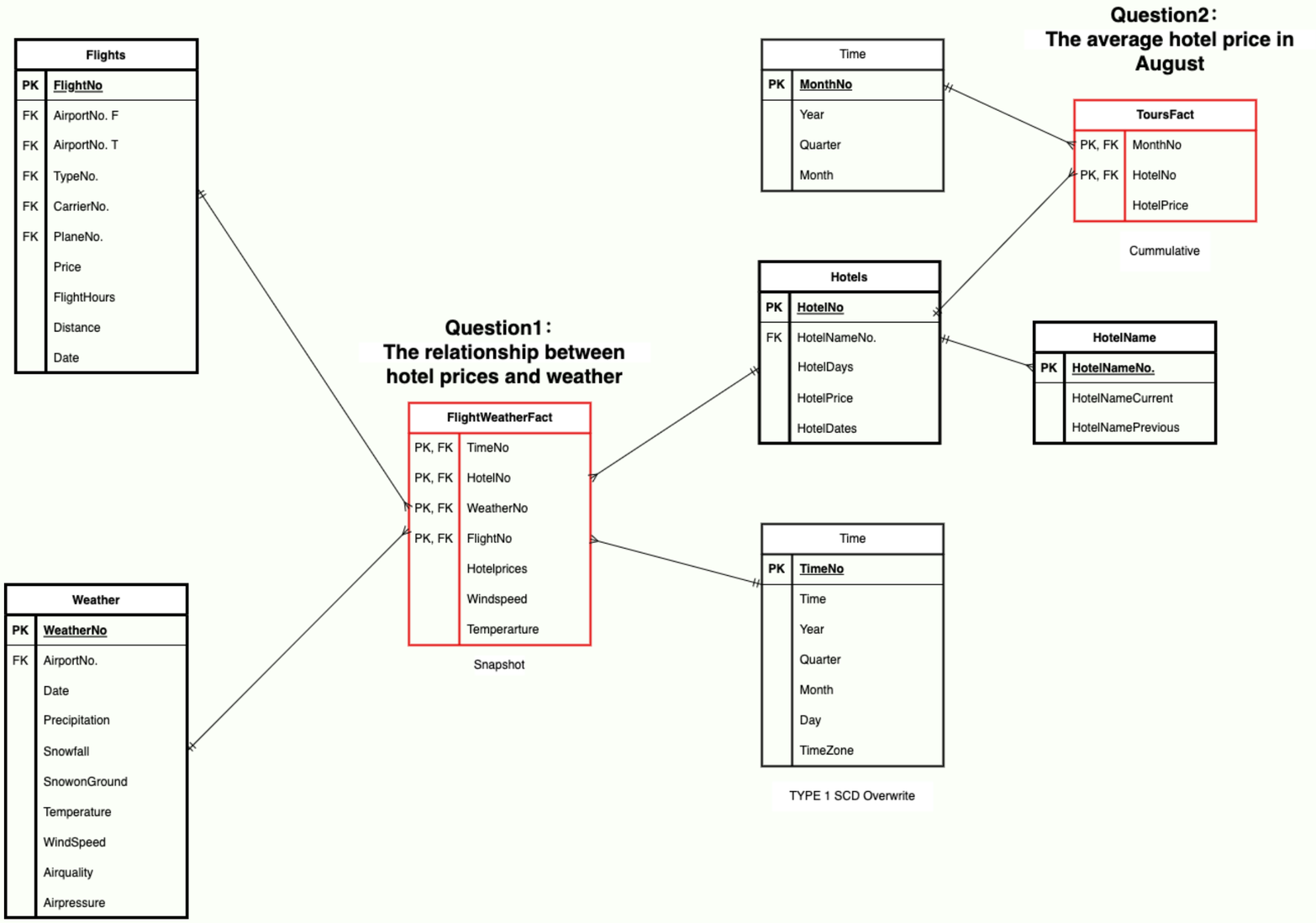
```
428 END;
429 $$;
430 --see changes
431 SELECT * FROM PROJECT.hotelname
432 WHERE hotelnameno = 2
433
434
435
```

Data Output 消息 通知

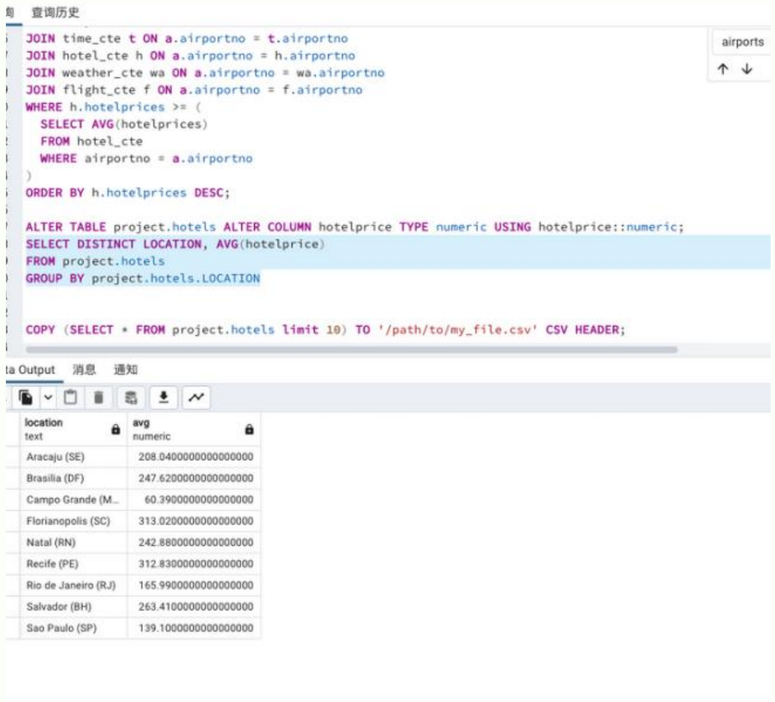
	hotelnameno [PK] integer	hotelnameprevious character varying	hotelnamecurrent character varying
1	2	Hotel A	Marriot

7.VISUAL AND ANALYSIS

DIMENSION&FACT



Question: What is the average hotel price in each area

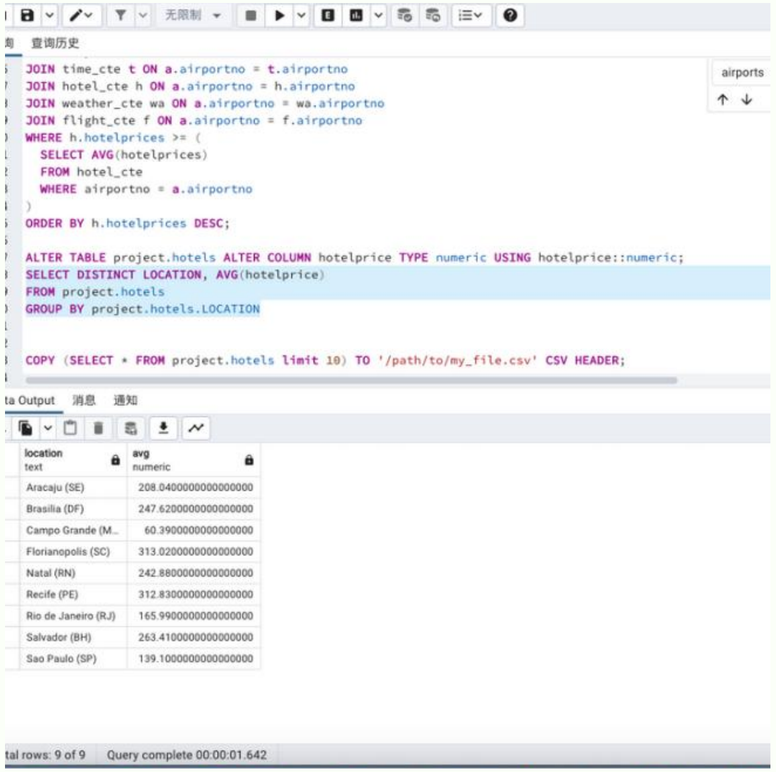


```
JOIN time_cte t ON a.airportno = t.airportno
JOIN hotel_cte h ON a.airportno = h.airportno
JOIN weather_cte wa ON a.airportno = wa.airportno
JOIN flight_cte f ON a.airportno = f.airportno
WHERE h.hotelprices >= (
  SELECT AVG(hotelprices)
  FROM hotel_cte
  WHERE airportno = a.airportno
)
ORDER BY h.hotelprices DESC;

ALTER TABLE project.hotels ALTER COLUMN hotelprice TYPE numeric USING hotelprice::numeric;
SELECT DISTINCT LOCATION, AVG(hotelprice)
FROM project.hotels
GROUP BY project.hotels.LOCATION


COPY (SELECT * FROM project.hotels limit 10) TO '/path/to/my_file.csv' CSV HEADER;
```

location text	avg numeric
Aracaju (SE)	208.0400000000000000
Brasilia (DF)	247.6200000000000000
Campo Grande (M...	60.3900000000000000
Florianopolis (SC)	313.0200000000000000
Natal (RN)	242.8800000000000000
Recife (PE)	312.8300000000000000
Rio de Janeiro (RJ)	165.9900000000000000
Salvador (BH)	263.4100000000000000
Sao Paulo (SP)	139.1000000000000000

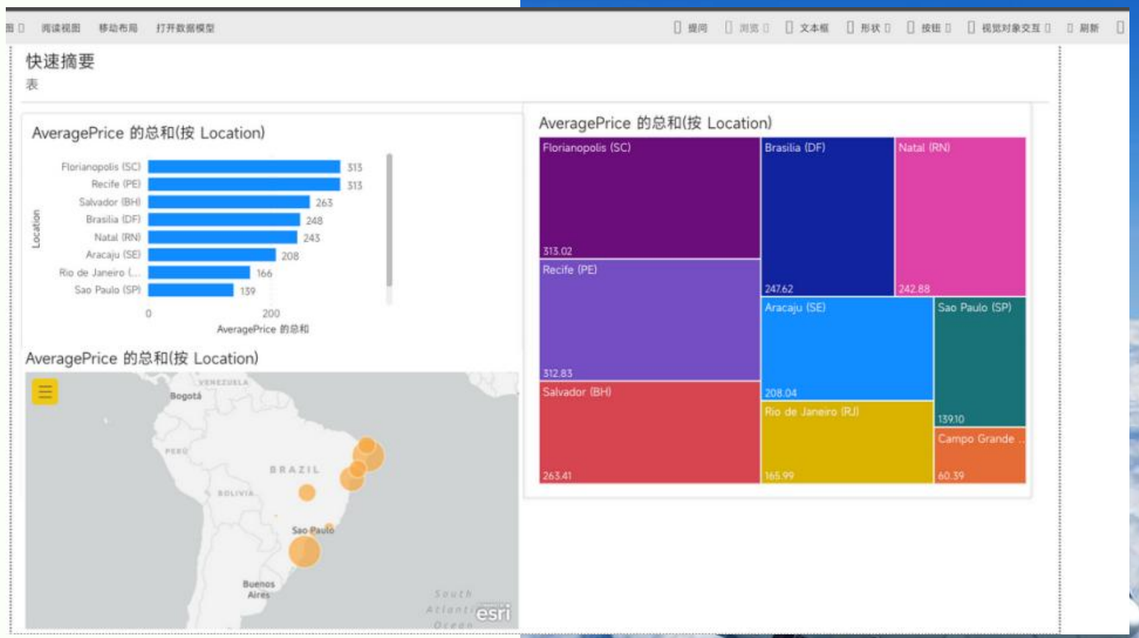
COPY (SELECT DISTINCT LOCATION, AVG(hotelprice)
FROM project.hotels
GROUP BY project.hotels.LOCATION) TO 'TO
'/users/yuanditang/downloads/avg.csv';

POWERBI CHARTS



```
JOIN time_cte t ON a.airportno = t.airportno
JOIN hotel_cte h ON a.airportno = h.airportno
JOIN weather_cte wa ON a.airportno = wa.airportno
JOIN flight_cte f ON a.airportno = f.airportno
WHERE h.hotelprices >= (
  SELECT AVG(hotelprices)
  FROM hotel_cte
  WHERE airportno = a.airportno
)
ORDER BY h.hotelprices DESC;

ALTER TABLE project.hotels ALTER COLUMN hotelprice TYPE numeric USING hotelprice::numeric;
SELECT DISTINCT LOCATION, AVG(hotelprice)
FROM project.hotels
GROUP BY project.hotels.LOCATION


COPY (SELECT * FROM project.hotels limit 10) TO '/path/to/my_file.csv' CSV HEADER;
```

location text	avg numeric
Aracaju (SE)	208.0400000000000000
Brasilia (DF)	247.6200000000000000
Campo Grande (M...	60.3900000000000000
Florianopolis (SC)	313.0200000000000000
Natal (RN)	242.8800000000000000
Recife (PE)	312.8300000000000000
Rio de Janeiro (RJ)	165.9900000000000000
Salvador (BH)	263.4100000000000000
Sao Paulo (SP)	139.1000000000000000



COPY (SELECT DISTINCT LOCATION, AVG(hotelprice)
FROM project.hotels
GROUP BY project.hotels.LOCATION) TO 'TO
'/users/yuanditang/downloads/avg.csv';

8. COMM THEORIES

COMM THEORY IN USE

Elaboration Likelihood Model (ELM): This theory proposes that there are two routes to persuasion: a central route and a peripheral route. The central route involves careful consideration and evaluation of the message, while the peripheral route relies on cues such as source credibility or emotional appeals. In airline marketing, the ELM could be used to design messages that appeal to both routes, such as highlighting the safety record of the airline (central route) while also using attractive visuals and emotional music in the marketing campaign (peripheral route).

Oliver, M. B., Raney, A. A., & Bryant, J. (2019). Media Effects: Advances in Theory and Research (4th Edition). New York, NY: Routledge. ISBN-13: 978-1138590229

COMM THEORY IN USE

Social Exchange Theory: This theory suggests that human relationships are based on an exchange of rewards and costs. In airline marketing, the social exchange theory could be applied by emphasizing the rewards of flying with the airline (such as comfort, convenience, and status) and minimizing the costs (such as price, inconvenience, and risk).

Oliver, M. B., Raney, A. A., & Bryant, J. (2019). Media Effects: Advances in Theory and Research (4th Edition). New York, NY: Routledge. ISBN-13: 978-1138590229

COMM THEORY IN USE

Uses and Gratifications Theory: This theory proposes that people use media to satisfy specific needs, such as information, entertainment, or social interaction. In airline marketing, the uses and gratifications theory could be applied by designing messages that appeal to the different needs of the target audience, such as providing information about flight schedules and amenities, showcasing the destination and cultural experiences, or promoting social connections and community through travel.

Oliver, M. B., Raney, A. A., & Bryant, J. (2019). Media Effects: Advances in Theory and Research (4th Edition). New York, NY: Routledge. ISBN-13: 978-1138590229

COMM THEORY IN USE

Social Learning Theory: This theory suggests that people learn new behaviors by observing and imitating others. In airline marketing, the social learning theory could be applied by featuring endorsements and testimonials from satisfied customers, celebrities, or influencers who embody the values and lifestyle associated with the airline brand. This could also be extended to social media campaigns that encourage user-generated content and sharing of travel experiences.

Oliver, M. B., Raney, A. A., & Bryant, J. (2019). Media Effects: Advances in Theory and Research (4th Edition). New York, NY: Routledge. ISBN-13: 978-1138590229

THANK YOU

Yuandi TANG U65674688