



---

# ANALYZE AND MODEL DATA WAREHOUSE FOR AIRLINE SYSTEM

---

Analyzing and Data Warehouse Modeling Project



**MADE BY:**

DINA HOSNY  
FATMA NABIL  
KAREEM BELTAGY  
MOHAMED LABIB

<b>INTRODUCTION:</b>	<b>2</b>
<b>DIMENSIONAL MODELING PROCESS:</b>	<b>2</b>
1- BUSINESS PROCESS:	2
2- GRANULARITY:	2
3- DIMENSIONS TABLES:	2
4- FACTS TABLES:	2
<b>PROJECT PROCESSES:</b>	<b>3</b>
SCHEMA TYPE:	3
PROCESS DIAGRAM:	3
<b>LOGICAL MODEL:</b>	<b>3</b>
<b>TABLES IDENTIFICATIONS:</b>	<b>5</b>
1- PASSENGER DIMENSION.	5
2- FLIGHT DIMENSION.	5
3- DATE DIMENSION.	5
4- CLASS DIMENSION.	5
5- RESERVATION CHANNEL DIMENSION.	5
6- FARE BASE DIMENSION.	5
7- HOTEL DIMENSION.	5
8- CUSTOMER SERVICES DIMENSION.	6
9- INTERACTION DIMENSION.	6
10- RESERVATION FACT.	6
11- TRANSIT HOTEL FACT.	6
12- CUSTOMER CARE FACT.	6
<b>ASSUMPTIONS:</b>	<b>7</b>
<b>PHYSICAL MODEL:</b>	<b>7</b>
<b>INSERT DUMMY DATA INTO THE PHYSICAL TABLES:</b>	<b>13</b>
<b>INSERT THE MEASURED DATA INTO FACT TABLES</b>	<b>19</b>
<b>SQL QUERIES TO ANSWER BUSINESS QUESTIONS:</b>	<b>23</b>

## Introduction:

The project's purpose is to analyze the flight activities of some airline companies and their frequent flyers and model its data warehouse schema.

## Dimensional Modeling Process:

### **1- Business process:**

Analyze the Airlines flights activities.

### **2- Granularity:**

Per Reservation.

**NOTE:** Customer Care contains transactions that can be committed before any reservation processes like making an interaction before the flight reservation. In this case and for this business process, the granularity will be Per Each Interaction.

### **3- Dimensions Tables:**

- 1- Passenger Dimension.
- 2- Flight Dimension.
- 3- Date Dimension.
- 4- Class Dimension.
- 5- Reservation Channel Dimension.
- 6- Fare Base Dimension.
- 7- Hotel Dimension.
- 8- Interaction Dimension.
- 9- Customer Services Dimension.

### **4- Facts Tables:**

- 1- Reservation Fact.
- 2- Transit Hotel Fact.
- 3- Customer Care Fact.

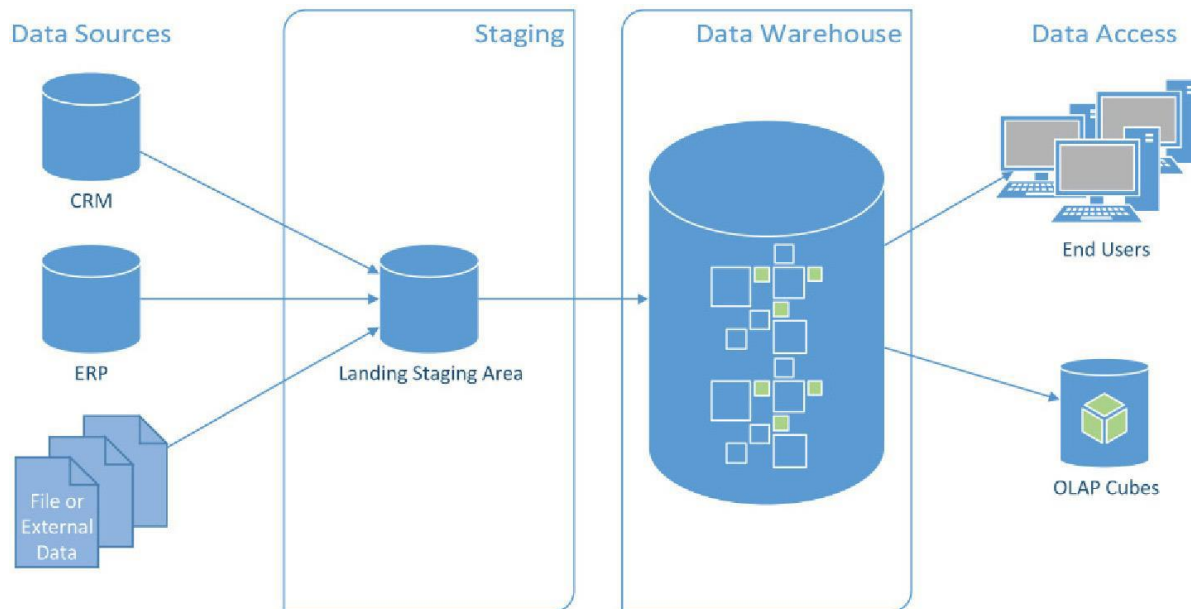
## Project Processes:

### Schema Type:

**Galaxy Schema:** The business process required more than one fact table to represent different processes.

### Process Diagram:

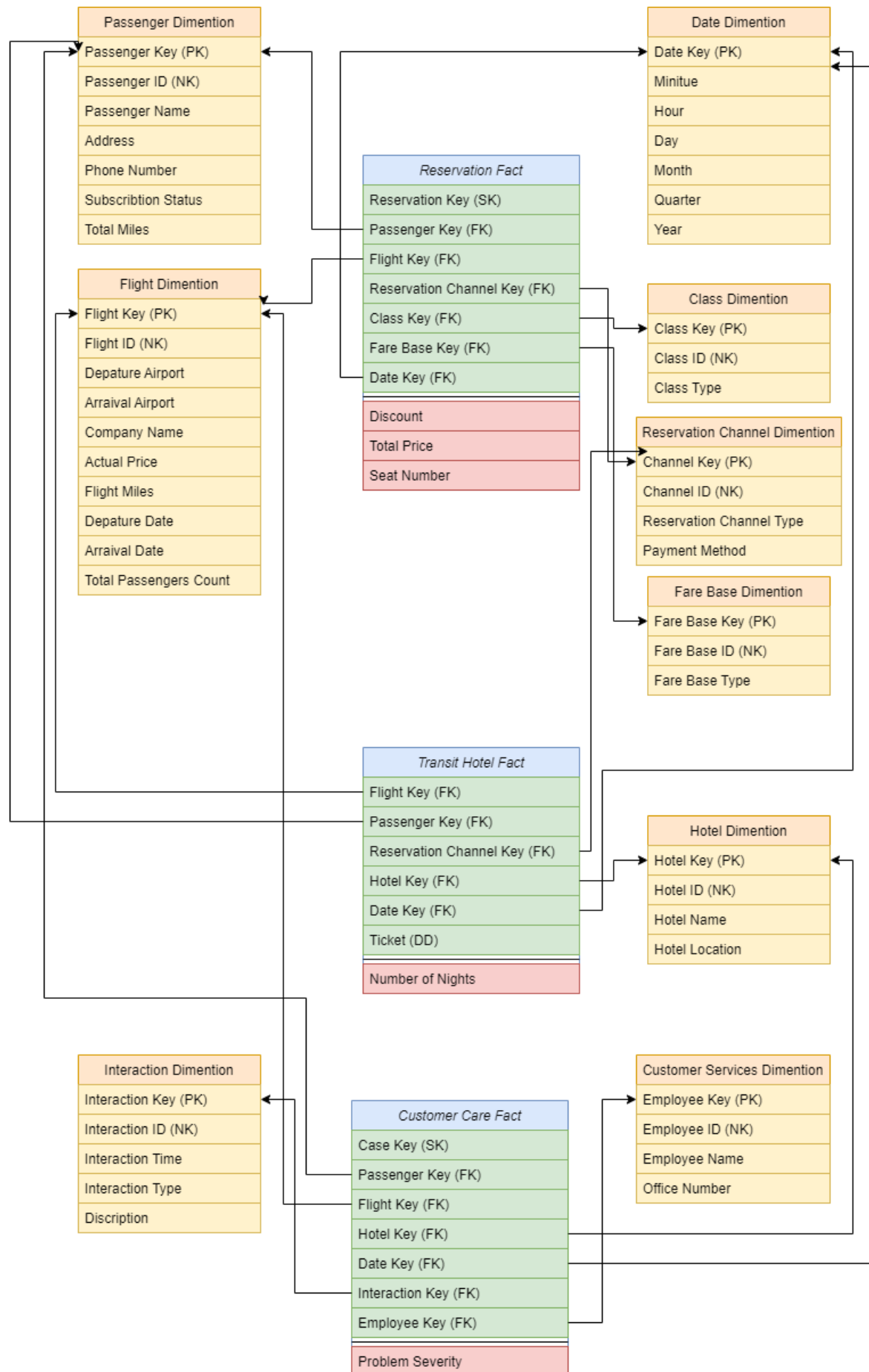
Using **Kimball's** Philosophy.



There will be one row in each fact table for each different transaction collected from passengers. The dimensionality associated with this data is quite extensive. The **Role-Playing** technique is used as the date dimension table was used more than one time in the facts table as different dates.

### Logical Model:

Represents the facts and dimensions tables, and the relations between them.



## Tables Identifications:

### 1- Passenger Dimension.

Represents data about passengers, like passenger **Name**, **ID**, **Address**, and **Phone Number**. In addition to their **Subscription Type**: **GOLD**, **PLATINUM**, or **TITANUM** provide different benefits, and the **Total Miles** for each passenger.

### 2- Flight Dimension.

Represents data about the flight, like flight **ID**, **Departure Airport**, **Arrival Airport**, **Airline Company Name**, the **Miles of the Flight**, the **Departure Date**, the **Arrival Date**, and the **Total Number of Passengers** on this flight.

### 3- Date Dimension.

Represents the **Minute**, **Hour**, **Day**, **Month**, **Quarter**, and **Year** for each date stored for different business needs.

### 4- Class Dimension.

Represents the passengers' reservation class for each flight, it can be **First Class**, **Business Class**, or **Economy Class**.

### 5- Reservation Channel Dimension.

Represents the channel that the ticket was purchased from. It can be **Website**, **Office**, or in the **Airport**. It also includes the payment method if it was via **Visa**, **Bank**, or **Cash**.

### 6- Fare Base Dimension.

Represents if the passenger fully paid the ticket price, or with a discount. this represents using code numbers which:

**111** means Full Fare First Class.

**112** means Discount Fare First Class.

**221** means Full Fare Business Class.

**222** means Discount Fare Business Class.

**331** means Full Fare Economy Class.

**332** means Discount Fare Economy Class.

### 7- Hotel Dimension.

Represents data about the hotels that the passengers stayed on during the transit flights, like hotel **ID**, **Name**, and its **Location**.

## 8- Customer Services Dimension.

Represents data about the customer services employees who are responsible for customer care interactions, like employee **ID**, **Name**, and his **Office Number**.

## 9- Interaction Dimension.

Represents the passenger's different interactions with customer services, it contains the **Interaction Time** if it was **Before**, **Within**, or **After** the flight, and the **Interaction Type** if it was **Feedback**, **Inquiry**, or **Complaint**.

## 10- Reservation Fact.

Represents the reservation processes and the data needed for it, it measures the **Discount** for each passenger based on his subscription:

**10% discount for GOLD subscribers.**

**15% discount for PLATINUM subscribers.**

**25% discount for TITANIUM subscribers.**

It also measures the **Total Price** after the discount and generates the passenger's **Seat Number** after the reservation process is done.

## 11- Transit Hotel Fact.

Represents the hotel reservation process and the **Number of Nights** that each passenger stayed based on the **Arrival Time** to the hotel, and the **Departure Time** from it. Each passenger is assigned to a hotel based on the **Ticket Number** he owned.

## 12- Customer Care Fact.

Represents the passenger's different interactions and the customer service's employee who is responsible for it. It measures the **Problem Severity** based on its **Type** and **Time** to work on different interactions based on its priorities. It represents like:

**1 for Complaint Within the flight.**

**2 for Complaint Before the flight.**

**3 for Complaint After the flight.**

**4 for Inquiry Within the flight.**

**5 for Feedback Within the flight.**

**6 for Inquiry Before the flight.**

**7 for Feedback Before the flight.**

**8 for Feedback After the flight.**

**9 for Inquiry After the flight.**

## Assumptions:

- 1- The passengers are considered as frequent flyers based on the **Number of Miles** that are assigned to their profiles.
- 2- The model assume that each passenger has a different subscription: **GOLD**, **PLATINUM**, or **TITANUM**. Each passenger can subscribe to one subscription that provides different benefits to him and can change it at any time.
- 3- The passenger table can apply the **Slowly Changed Dimension** concept on the **Subscription** attribute, that it may change its value over the time.
- 4- To detect if the passenger paid the ticket using a discount or not, we used the **Fare Base** dimension that stored this information for some business needs
- 5- The discount is computed for each passenger based on his subscription, that the **GOLD** subscribers can have a **10%** discount on their ticket price, the **PLATINUM** subscribers can have a **15%** discount on their ticket price, and the **TITANUM** subscribers can have a **25%** on their ticket price.
- 6- The **Total Price = Actual Price – (Discount \* Actual Price)**; that the actual price is the ticket price stored in the Flight dimension, and the discount is the value that computed based on the passengers' subscription.
- 7- The **Seat Number** is a unique number generated for each passenger with each reservation process.
- 8- The **Number of Nights = Arrival Date – Departure date** from the hotel.
- 9- The **Problem Severity** computes based on the **Interaction Time** and **Interaction Type**. It assumes that **Complaints** always have the highest priority even if it is before or after the flight.

## Physical Model:

The creation of the tables using SQL queries. In this project, PL SQL is used for all queries.

### 1- Create Passenger Dimension:

```
CREATE TABLE passenger_dim (  
  passenger_key NUMBER CONSTRAINT pk_cons PRIMARY KEY,  
  passenger_id NUMBER CONSTRAINT uniq_cons UNIQUE,  
  passenger_name VARCHAR(250),  
  address VARCHAR(250),  
  phone VARCHAR(250),  
  sub_stat VARCHAR(250),  
  total_miles NUMBER );
```



Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
► PASSENGER_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PASSENGER_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PASSENGER_NAME	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
ADDRESS	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
PHONE	5		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
SUB_STAT	6		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
TOTAL_MILES	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

## 2- Create Flight Dimension:

```
CREATE TABLE flight_dim (
flight_key NUMBER CONSTRAINT flight_pk_cons PRIMARY KEY,
flight_id NUMBER CONSTRAINT flight_unq_cons UNIQUE,
depature_airport VARCHAR(250),
arraival_airport VARCHAR(250),
comany_name VARCHAR(250),
actual_price NUMBER,
flight_miles NUMBER,
depature_date DATE,
arraival_date DATE,
total_passenger_count NUMBER );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
► FLIGHT_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FLIGHT_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DEPATURE_AIRPORT	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
ARRAIVAL_AIRPORT	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
COMANY_NAME	5		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
ACTUAL_PRICE	6		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FLIGHT_MILES	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DEPATURE_DATE	8		Y	DATE		None		<input type="checkbox"/>	<input type="checkbox"/>
ARRAIVAL_DATE	9		Y	DATE		None		<input type="checkbox"/>	<input type="checkbox"/>
TOTAL_PASSENGER_COUNT	10		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

## 3- Create Date Dimension:

```
CREATE TABLE date_dim (
date_key NUMBER CONSTRAINT date_pk_cons PRIMARY KEY,
minitue NUMBER,
hours NUMBER,
days NUMBER,
months NUMBER,
quarter NUMBER,
years NUMBER );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
DATE_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
MINUTUE	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
HOURS	3		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DAYS	4		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
MONTHS	5		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
QUARTER	6		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
YEARS	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

#### 4- Create Class Dimension:

```
CREATE TABLE class_dim (
class_key NUMBER CONSTRAINT class_pk_cons PRIMARY KEY,
class_id NUMBER CONSTRAINT class_unq_cons UNIQUE,
class_type VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
CLASS_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
CLASS_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
CLASS_TYPE	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

#### 5- Create Reservation Channel Dimension:

```
CREATE TABLE reservation_channel_dim (
channel_key NUMBER CONSTRAINT channel_pk_con PRIMARY KEY,
channel_id NUMBER CONSTRAINT channel_unq_cond UNIQUE,
reservation_channel_type VARCHAR(250),
payment_method VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
CHANNEL_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
CHANNEL_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
RESERVATION_CHANNEL_TYPE	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
PAYMENT_METHOD	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

#### 6- Create Fare Base Dimension:

```
CREATE TABLE fare_base_dim (
fare_base_key NUMBER CONSTRAINT fare_pk_cons PRIMARY KEY,
fare_base_id NUMBER CONSTRAINT fare_unq_cons UNIQUE,
fare_base_type VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
FARE_BASE_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FARE_BASE_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FARE_BASE_TYPE	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

## 7- Create Hotel Dimension:

```
CREATE TABLE hotel_dim (
  hotel_key NUMBER CONSTRAINT hotel_pk_cons PRIMARY KEY,
  hotel_id NUMBER CONSTRAINT hotel_unq_cons UNIQUE,
  hotel_name VARCHAR(250),
  hotel_location VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
HOTEL_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
HOTEL_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
HOTEL_NAME	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
HOTEL_LOCATION	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

## 8- Create Customer Services Dimension:

```
CREATE TABLE cutomer_services_dim (
  employee_key NUMBER CONSTRAINT emp_pk_cons PRIMARY KEY,
  employee_id NUMBER CONSTRAINT emp_unq_cons UNIQUE,
  employee_name VARCHAR(250),
  office_number VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
EMPLOYEE_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
EMPLOYEE_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
EMPLOYEE_NAME	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
OFFICE_NUMBER	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

## 9- Create Interaction Dimension:

```
CREATE TABLE interation_dim (
  interaction_key NUMBER CONSTRAINT inter_pk_cons PRIMARY KEY,
  interaction_id NUMBER CONSTRAINT inter_unq_cons UNIQUE,
  interaction_time VARCHAR(250),
  interaction_type VARCHAR(250),
  discription VARCHAR(250) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
INTERACTION_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
INTERACTION_ID	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
INTERACTION_TIME	3		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
INTERACTION_TYPE	4		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>
DISCRPTION	5		Y	VARCHAR2 (250 Byte)		None		<input type="checkbox"/>	<input type="checkbox"/>

## 10- Create Reservation Fact:

```
CREATE TABLE reservation_fact (
reservation_key NUMBER CONSTRAINT res_pk_cons PRIMARY KEY,
passenger_key NUMBER,
flight_key NUMBER,
reservation_channel_key NUMBER,
class_key NUMBER,
fare_base_key NUMBER,
date_key NUMBER,
discount NUMBER,
total_price NUMBER,
seat_number NUMBER );
```

```
ALTER TABLE reservation_fact
ADD (
CONSTRAINT pass_fk_cons FOREIGN KEY(passenger_key) REFERENCES
passenger_dim(passenger_key),
CONSTRAINT flight_fk_con FOREIGN KEY(flight_key) REFERENCES flight_dim(flight_key),
CONSTRAINT channel_fk_cons FOREIGN KEY(reservation_channel_key) REFERENCES
reservation_channel_dim(channel_key),
CONSTRAINT class_fk_cons FOREIGN KEY(class_key) REFERENCES class_dim(class_key),
CONSTRAINT fare_fk_cons FOREIGN KEY(fare_base_key) REFERENCES
fare_base_dim(fare_base_key),
CONSTRAINT date_fk_cons FOREIGN KEY(date_key) REFERENCES date_dim(date_key) );
```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
RESERVATION_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PASSENGER_KEY	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FLIGHT_KEY	3		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
RESERVATION_CHANNEL_KEY	4		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
CLASS_KEY	5		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FARE_BASE_KEY	6		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DATE_KEY	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DISCOUNT	8		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
TOTAL_PRICE	9		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
SEAT_NUMBER	10		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

## 11- Create Transit Hotel Fact:

```
CREATE TABLE transit_hotel_fact (
flight_key NUMBER,
passenger_key NUMBER,
reservation_channel_key NUMBER,
hotel_key NUMBER,
date_key NUMBER,
ticket_number NUMBER,
number_of_nights NUMBER );
```

```

ALTER TABLE transit_hotel_fact
ADD (
CONSTRAINT flight_fk_conss FOREIGN KEY(flight_key) REFERENCES flight_dim(flight_key),
CONSTRAINT pass_fk_conss FOREIGN KEY(passenger_key) REFERENCES
passenger_dim(passenger_key),
CONSTRAINT rese_fk_conss FOREIGN KEY(reservation_channel_key) REFERENCES
reservation_channel_dim(channel_key),
CONSTRAINT hotel_fk_conss FOREIGN KEY(hotel_key) REFERENCES hotel_dim(hotel_key),
CONSTRAINT date_fk_conss FOREIGN KEY(date_key) REFERENCES date_dim(date_key));

```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
FLIGHT_KEY	1		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PASSENGER_KEY	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
RESERVATION_CHANNEL_KEY	3		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
HOTEL_KEY	4		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DATE_KEY	5		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
TICKET_NUMBER	6		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
NUMBER_OF_NIGHTS	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

## 12- Create Customer Care Fact:

```

CREATE TABLE customer_care_fact(
case_key NUMBER CONSTRAINT case_pk_cons PRIMARY KEY,
passenger_key NUMBER,
flight_key NUMBER,
hotel_key NUMBER,
date_key NUMBER,
interaction_key NUMBER,
employee_key NUMBER,
problem_severity NUMBER );

```

```

ALTER TABLE customer_care_fact
ADD (
CONSTRAINT pass_fk_consss FOREIGN KEY(passenger_key) REFERENCES
passenger_dim(passenger_key),
CONSTRAINT flight_fk_consss FOREIGN KEY(flight_key) REFERENCES flight_dim(flight_key),
CONSTRAINT hotel_fk_consss FOREIGN KEY(hotel_key) REFERENCES hotel_dim(hotel_key),
CONSTRAINT date_fk_consss FOREIGN KEY(date_key) REFERENCES date_dim(date_key),
CONSTRAINT inter_fk_consss FOREIGN KEY(interaction_key) REFERENCES
interaction_dim(interaction_key),
CONSTRAINT emp_fk_consss FOREIGN KEY(employee_key) REFERENCES
customer_services_dim(employee_key) );

```

Column Name	ID	Pk	Null?	Data Type	Default	Histogram	Encryption Alg	Salt	Seq/Trigger
CASE_KEY	1	1	N	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PASSENGER_KEY	2		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
FLIGHT_KEY	3		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
HOTEL_KEY	4		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
DATE_KEY	5		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
INTERACTION_KEY	6		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
EMPLOYEE_KEY	7		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>
PROBLEM_SEVERITY	8		Y	NUMBER		None		<input type="checkbox"/>	<input type="checkbox"/>

## Insert Dummy Data into The Physical Tables:

Based on the HR schema that ORACLE provides for learning, and using some other techniques and functions, some dummy data inserted into the tables for testing and answering some business questions.

### 1- Class Dimension Data:

	CLASS_KEY	CLASS_ID	CLASS_TYPE
▶	1	111	First Class
	2	222	Business Class
	3	333	Economy Class

### 2- Customer Services Dimension Data:

	EMPLOYEE_KEY	EMPLOYEE_ID	EMPLOYEE_NAME	OFFICE_NUMBER
▶	1	100	King	AD_PRES
	2	101	Kochhar	AD_VP
	3	102	De Haan	AD_VP
	4	103	Hunold	IT_PROG
	5	104	Ernst	IT_PROG
	6	105	Austin	FI_ACCOUNT
	7	106	Pataballa	IT_PROG
	8	107	Lorentz	IT_PROG
	9	108	Greenberg	FI_MGR
	10	109	Faviet	FI_ACCOUNT
	11	110	Chen	FI_ACCOUNT
	12	111	Sciarra	FI_ACCOUNT
	13	112	Urman	FI_ACCOUNT
	14	113	Popp	FI_ACCOUNT
	15	114	Raphaely	PU_MAN
	16	115	Khoo	PU_CLERK
	17	116	Baida	PU_CLERK

### 3- Date Dimension Data:

	DATE_KEY	MINITUE	HOURS	DAYS	MONTHS	QUARTER	YEARS
▶	1	44	23	1	1	2	2000
	2	9	22	21	9	1	2005
	3	4	4	13	1	2	2001
	4	35	21	3	1	3	2006
	5	29	8	21	5	2	2007
	6	44	9	1	1	1	2000
	7	8	21	5	2	3	2006
	8	35	9	7	2	1	2007
	9	53	11	17	8	3	2002
	10	6	0	16	8	3	2002
	11	15	0	28	9	2	2005
	12	49	11	30	9	1	2005
	13	23	22	7	3	2	2006
	14	14	1	7	12	2	2007
	15	20	0	7	12	2	2002
	16	57	10	18	5	2	2003
	17	31	23	24	12	1	2005
	18	57	10	24	7	2	2005
	19	26	3	15	11	2	2006
	20	40	19	10	8	2	2007

### 4- Fare Base Dimension Data:

	FARE_BAS...	FARE_BASE_ID	FARE_BASE_TYPE
▶	1	111	Full Fare First Class
	2	112	Discount Fare First Class
	3	221	Full Fare Business Class
	4	222	Discount Fare Business Class
	5	331	Full Fare Economy Class
	6	332	Discount Fare Economy Class

## 5- Flight Dimension Data:

FLIGHT_KEY	FLIGHT_ID	DEPARTURE_AIRPORT	ARRIVAL_AIRPORT	COMANY_NAME	ACTUAL_PRICE	FLIGHT_MILES	DEPARTURE_DATE	ARRIVAL_DATE	TOTAL_PASSENGER_COUNT
107	25	Roma	Argentina	Aegean Airlines	2200	2550.52454208148	09/17/2003	11/17/2003	53
108	40	Venice	Australia	Hawaiian Airlines	3300	4378.05026460416	02/17/2004	04/17/2004	244
110	70	Hiroshima	Brazil	American Airlines	2500	4521.41180066077	12/07/2002	02/07/2003	275
111	85	Southlake	Canada	American Airlines	2100	9309.28354831194	05/18/2003	07/18/2003	261
112	100	South San Francisco	Switzerland	American Airlines	3300	6127.09804189225	12/24/2005	02/24/2006	277
113	115	South Brunswick	China	American Airlines	2900	2025.0177570767	07/24/2005	09/24/2005	167
114	130	Seattle	Germany	American Airlines	2400	7721.78824303379	11/15/2006	01/15/2007	195
115	145	Toronto	Denmark	American Airlines	2200	6358.17177119324	08/10/2007	10/10/2007	233
116	160	Whitehorse	Egypt	Emirates	3600	7870.03632521858	06/07/2002	08/07/2002	112
117	175	Beijing	France	Egyptair	3200	4865.18117994981	07/18/2004	09/18/2004	298
118	190	Bombay	Israel	Egyptair	2700	8215.50599505266	04/10/2005	06/10/2005	164
119	205	Sydney	India	Egyptair	2500	5258.40168994574	05/01/2003	07/01/2003	213
120	220	Singapore	Italy	Egyptair	3500	7389.91537045954	10/10/2005	12/10/2005	243
122	250	Oxford	Kuwait	Egyptair	2600	8541.00630073775	07/16/2005	09/16/2005	251
123	265	Stretford	Malaysia	Egyptair	2500	5468.31801802918	09/28/2006	11/28/2006	115
124	280	Munich	Mexico	Egyptair	14000	2591.14058542827	01/14/2007	03/14/2007	291
125	295	Sao Paulo	Nigeria	Egyptair	13500	4242.05048898465	03/08/2008	05/08/2008	105
126	310	Geneva	Netherlands	Egyptair	12000	8565.463242847	08/20/2005	10/20/2005	215
127	325	Bern	Singapore	Egyptair	11000	4956.18960364778	10/30/2005	12/30/2005	134
128	340	Utrecht	United Kingdom	Egyptair	10500	2065.6748818056	02/16/2005	04/16/2005	123

## 6- Hotel Dimension Data:

HOTEL_KEY	HOTEL_ID	HOTEL_NAME	HOTEL_LOCATION
2	1000		1297 Via Cola di Rie
3	1100		93091 Calle della Testa
4	1200	Tokyo Prefecture	2017 Shinjuku-ku
5	1300		9450 Kamiya-cho
6	1400	Texas	2014 Jabberwocky Rd
7	1500	California	2011 Interiors Blvd
8	1600	New Jersey	2007 Zagora St
9	1700	Washington	2004 Charade Rd
10	1800	Ontario	147 Spadina Ave
11	1900	Yukon	6092 Boxwood St
12	2000		40-5-12 Laogianggen
13	2100	Maharashtra	1298 Vileparle (E)
14	2200	New South Wales	12-98 Victoria Street
15	2300		198 Clementi North
16	2400		8204 Arthur St
17	2500	Oxford	Magdalen Centre, The Oxford Science Park
18	2600	Manchester	9702 Chester Road
19	2700	Bavaria	Schwanthalerstr. 7031
20	2800	Sao Paulo	Rua Frei Caneca 1360



## 7- Interaction Dimension Data:

INTERACTIO...	INTERACTION_ID	INTERACTION_TIME	INTERACTION_TYPE	DISCRIPTION
1	10	after flight	feedback	twBplzwlTxgMZAHkiAuvJRdUhGyVQOKnXSsm
2	20	after flight	complaint	eXJUtaqxLLsuDdMtgliaDzKCNHYqPTqwJc
3	30	within flight	complaint	eJqrHVJYYLTsBCzsOaRxglQtkqJoIUUZMUE
4	40	within flight	complaint	BBvKGmXrMENznRhQXhZVPzAgPpeylfYtDxJ
5	50	within flight	inquiry	zAxJlyPgBXZMptqoMUIRrZIrSviPCWzoPr
6	60	within flight	complaint	ZJfpCRXZsFjUmxAJLuXRrStvZnVPURPzWVj
7	70	before flight	complaint	sGjueWbtupSFwKZoqGJmyxAEXimZOaSMtgV
8	80	after flight	complaint	niVQMglzsjNJFbaByTVnPoVTbtGoVntegq
9	90	before flight	complaint	BWSeiKKKyJnXiwGchCOgFPXABzsWQDKbTI
10	100	after flight	inquiry	ghexOxJrbuAIVYcaKwNByrBBYUUmSsUEMc
11	110	after flight	feedback	ErfMFJaQGwacThMQMALkhfsAkhpevoXFbI
12	120	within flight	inquiry	iikMTYyTUiLegmecBkFdoYVqBwxVgYMKBSZ
13	130	after flight	inquiry	jkzfEQrXhWLSfgtAtyXcjiizwLeWBkOQIJ
14	140	within flight	inquiry	hSbcKKJoWhpyPfvouhBHJgKJbRbmXHuzyHV
15	150	within flight	complaint	LEiLyRUhXerzYDCsoxSikDOISEnUpLJTNG
16	160	within flight	complaint	mLBiENXJifUTjuDnJiKkTXnzTGGqRsymkFk
17	170	after flight	complaint	wLUEAnEWyCvrvDzwcRBGGSiZWGVvSkEQvPH
18	180	within flight	complaint	owmhjVtfvWGCtItEsCnJSRqATYJntSdgtph
19	190	before flight	inquiry	fykgBcYKsezanVXOjMGIHUwUcoWLoMnISsa
20	200	after flight	feedback	GubyCxRQnovYWFYInNmBInGsqoMTLXagJwV

## 8- Passenger Dimension Data:

PASSENGE...	PASSENGER_ID	PASSENGER_NAME	ADDRESS	PHONE	SUB_STAT	TOTAL_MILES
216	101	Neena	2014 Jabbberwocky Rd	515.123.4568	platinum	88502.6956301504
217	102	Lex	2011 Interiors Blvd	515.123.4569	titanium	96726.2442806679
218	103	Alexander	2007 Zagora St	590.423.4567	platinum	98023.1417284964
219	104	Bruce	2004 Charade Rd	590.423.4568	titanium	27171.7787238698
220	105	David	147 Spadina Ave	590.423.4569	titanium	61680.2845805925
221	106	Valli	6092 Boxwood St	590.423.4560	titanium	95237.492012167
222	107	Diana	40-5-12 Laogianggen	590.423.5567	titanium	56980.6415776557
224	109	Daniel	12-98 Victoria Street	515.124.4169	titanium	35633.9425789926
225	110	John	198 Clementi North	515.124.4269	platinum	23385.9039002057
226	111	Ismael	8204 Arthur St	515.124.4369	titanium	48468.5064517751
227	112	Jose Manuel	Magdalen Centre, The Oxford Science Park	515.124.4469	gold	56754.8932721566
228	113	Luis	9702 Chester Road	515.124.4567	titanium	5505.59977834346
229	114	Den	Schwanthalerstr. 7031	515.127.4561	titanium	91556.9903815056
230	115	Alexander	Rua Frei Caneca 1360	515.127.4562	gold	26946.5924708267
231	116	Shelli	20 Rue des Corps-Saints	515.127.4563	titanium	57916.3565328519
232	117	Sigal	Murtenstrasse 921	515.127.4564	platinum	13016.1135220223
234	119	Karen	Mariano Escobedo 9991	515.127.4566	titanium	59007.5497927365
235	120	Matthew	helmy gomaa	650.123.1234	gold	43402.5403056213

## 9- Reservation Channel Dimension Data:

ini	CHANNEL_KEY	CHANNEL_ID	RESERVATION_CHANNEL_TYPE	PAYMENT_METHOD
▶	1	111	website	visa
	2	112	website	cash
	3	113	website	bank
	4	221	office	visa
	5	222	office	cash
	6	223	office	bank
	7	331	airport	visa
	8	332	airport	cash
	9	333	airport	bank

## 10- Customer Care Fact Data:

ini	CASE_KEY	PASSENGER_KEY	FLIGHT_KEY	HOTEL_KEY	DATE_KEY	INTERACTION_KEY	EMPLOYEE_KEY	PROBLEM_SEVERITY
▶	1000	216	107	18	76	1	52	8
	1001	217	108	14	80	2	75	3
	1002	218	109	10	95	3	85	1
	1003	219	110	13	12	4	82	1
	1004	220	111	19	25	5	80	4
	1005	221	112	11	30	6	44	1
	1006	222	113	9	29	7	72	2
	1007	223	114	24	102	8	94	3
	1008	224	115	23	42	9	91	2
	1009	225	116	11	27	10	58	9
	1010	226	117	24	26	11	61	8
	1011	227	118	18	38	12	74	4
	1012	228	119	16	87	13	56	9
	1013	229	120	12	81	14	5	4
	1014	230	121	24	13	15	41	1
	1015	231	122	20	80	16	96	1
	1016	232	123	3	88	17	68	3

## 11- Reservation Fact Data:

RESERVATION_KEY	PASSENGER_KEY	FLIGHT_KEY	RESERVATION_CHANNEL_KEY	CLASS_KEY	FARE_BASE_KEY	DATE_KEY	DISCOUNT	TOTAL_PRICE	SEAT_NUMBER
1001	216	107	7	2	2	95	0.15	330	241
1101	217	108	8	2	5	83	0.25	825	214
1201	218	109	6	2	5	47	0.15	420	7
1301	219	110	3	2	3	68	0.25	625	242
1401	220	111	4	1	3	91	0.25	525	41
1501	221	112	7	2	5	76	0.25	825	216
1601	222	113	1	2	4	4	0.25	725	186
1701	223	114	8	2	1	94	0.25	600	63
1801	224	115	1	1	3	40	0.25	550	157
1901	225	116	6	2	1	50	0.15	540	165
2001	226	117	7	1	1	47	0.25	800	23
2101	227	118	6	2	4	84	0.1	270	181
2201	228	119	7	1	1	101	0.25	625	254
2301	229	120	3	1	4	10	0.25	875	56
2401	230	121	8	2	5	61	0.1	310	103
2501	231	122	8	2	1	20	0.25	650	78
2601	232	123	5	1	3	44	0.15	375	71

## 12- Transit Hotel Fact Data:

FLIGHT_KEY	PASSENGER_KEY	RESERVATION_CHANNEL_KEY	HOTEL_KEY	DATE_KEY	TICKET_NUMBER	NUMBER_OF_NIGHTS
107	216	2	11	2	28126	6982
108	217	5	5	10	47359	6830
109	218	1	21	2	89152	6282
110	219	8	12	39	21255	7265
111	220	7	18	82	55081	7104
112	221	4	8	17	84854	6152
113	222	4	6	82	43055	6305
114	223	3	24	15	51003	5827
115	224	3	15	9	38231	5559
116	225	5	24	82	68256	7449
117	226	8	9	95	86347	6676
118	227	1	24	52	65321	6411
119	228	3	15	34	57771	7121
120	229	8	20	59	62483	6228
121	230	5	19	48	74244	5461
122	231	2	7	89	43292	6313

## Insert The Measured Data Into Fact Tables

### 1- Insert Data into Discount Major in Reservation Fact:

```
DECLARE
```

```
    v_sub VARCHAR(250);
```

```
CURSOR Cur IS
```

```
    SELECT RESERVATION_KEY, PASSENGER_KEY, DISCOUNT
    from RESERVATION_FACT
    FOR UPDATE OF DISCOUNT;
```

```
BEGIN
```

```
    FOR Rec IN Cur
    LOOP
```

```
        SELECT sub_stat
        INTO v_sub
        FROM passenger_dim
        WHERE passenger_key = Rec.passenger_key;
```

```
        IF(v_sub = 'gold') THEN
            UPDATE RESERVATION_FACT
            SET discount = 0.10
            WHERE CURRENT OF Cur;
```

```
        ELSIF(v_sub = 'platinum') THEN
            UPDATE RESERVATION_FACT
            SET discount = 0.15
            WHERE CURRENT OF Cur;
```

```
        ELSIF(v_sub = 'titanium') THEN
            UPDATE RESERVATION_FACT
            SET discount = 0.25
            WHERE CURRENT OF Cur;
```

```
    END IF;
```

```
END LOOP;
```

```
END;
```

## 2- Insert Data into Total Price Major in Reservation Fact:

```
DECLARE

v_price NUMBER;
v_discount NUMBER;

CURSOR Cur IS
SELECT RESERVATION_KEY, flight_key , total_price, discount
  from RESERVATION_FACT
  FOR UPDATE OF total_price;

BEGIN

FOR Rec IN Cur

LOOP

  SELECT actual_price
  INTO v_price
  FROM flight_dim
  WHERE flight_key = Rec.flight_key;

  UPDATE RESERVATION_FACT
  SET total_price = v_price - (v_price*Rec.discount)
  WHERE CURRENT OF Cur;

END LOOP;

END;
```

## 3- Insert Data into Seat Number Major in Reservation Fact:

```
DECLARE

CURSOR Cur IS
SELECT flight_key , total_price, seat_number
  from RESERVATION_FACT
  FOR UPDATE OF seat_number;

BEGIN

FOR Rec IN Cur

LOOP

  IF (Rec.flight_key IS IN (SELECT flight_key
                           FROM flight_dim))
  THEN

    UPDATE RESERVATION_FACT
    SET seat_number = TRUNC(dbms_random.value(1,300),0)
```

```

        WHERE CURRENT OF Cur;

    END IF;

    END LOOP;

END;

```

#### 4- Insert Data into Number of Nights Major in Transit Hotel Fact:

```

DECLARE

    v_date DATE;

    CURSOR Cur IS
        SELECT FLIGHT_KEY ,NUMBER_OF_NIGHTS
        from TRANSIT_HOTEL_FACT
        FOR UPDATE OF NUMBER_OF_NIGHTS;

BEGIN

    FOR Rec IN Cur

    LOOP

        SELECT arraival_date
        INTO v_date
        FROM flight_dim
        WHERE flight_key = Rec.flight_key;

        UPDATE TRANSIT_HOTEL_FACT
        SET NUMBER_OF_NIGHTS = TRUNC(SYSDATE - v_date ,0)
        WHERE CURRENT OF Cur;

    END LOOP;

END;

```

#### 5- Insert Data into Problem Severity Major in Customer Care Fact:

```

DECLARE

    v_type VARCHAR(250);
    v_time VARCHAR(250);

    CURSOR Cur IS
        SELECT CASE_KEY ,PROBLEM_SEVERITY, INTERACTION_KEY
        from CUSTOMER_CARE_FACT
        FOR UPDATE OF PROBLEM_SEVERITY;

```

BEGIN

FOR Rec IN Cur

LOOP

```
SELECT INTERACTION_TIME, INTERACTION_TYPE
INTO v_time, v_type
FROM INTERACTION_DIM
WHERE INTERACTION_KEY = Rec.INTERACTION_KEY;
```

```
IF (v_time = 'within flight' AND v_type = 'complaint')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 1
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'before flight' AND v_type = 'complaint')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 2
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'after flight' AND v_type = 'complaint')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 3
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'within flight' AND v_type = 'inquiry')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 4
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'within flight' AND v_type = 'feedback')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 5
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'before flight' AND v_type = 'inquiry')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 6
    WHERE CURRENT OF Cur;
```

```
ELSIF (v_time = 'before flight' AND v_type = 'feedback')
THEN
```

```
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 7
    WHERE CURRENT OF Cur;
```

```

ELSEIF (v_time = 'after flight' AND v_type = 'feedback')
THEN
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 8
    WHERE CURRENT OF Cur;

ELSEIF (v_time = 'after flight' AND v_type = 'inquiry')
THEN
    UPDATE CUSTOMER_CARE_FACT
    SET PROBLEM_SEVERITY = 9
    WHERE CURRENT OF Cur;

END IF;

END LOOP;

END;

```

## SQL Queries to Answer Business Questions:

### 1- Find the Most Common Arrival Destinations:

```

SELECT ARRIVAL_AIRPORT, COUNT(ARRIVAL_AIRPORT) AS "Number of Visits"
FROM flight_dim
GROUP BY ARRIVAL_AIRPORT
ORDER BY COUNT(ARRIVAL_AIRPORT) DESC;

```

ARRIVAL_AIRPORT	Number of Visits
Egypt	9
United States of America	7
Italy	7
Mexico	5
United Kingdom	5
Germany	5
France	5
Argentina	5
China	5
Australia	4
Nigeria	4
India	4
Malaysia	4
Denmark	4
Kuwait	4



## 2- Find the Interactions with Type “Complaint” and the Employee Who Works on It:

```
SELECT i.INTERACTION_TYPE, i.DISCRIPTION, c.EMPLOYEE_NAME
FROM INTERACTION_DIM i JOIN CUSTOMER_CARE_FACT f
ON i.INTERACTION_KEY = f.INTERACTION_KEY
JOIN CUTOMER_SERVICES_DIM c
ON F.EMPLOYEE_KEY = C.EMPLOYEE_KEY
WHERE I.INTERACTION_TYPE = 'complaint';
```

INTERACTION_...	DISCRIPTION	EMPLOYEE_NAME
complaint	eXJUtaqxLLsuDdMtglliaDzKCNHYqPTqwJc	Abel
complaint	eJqrHVJYYLTsBCzsOaRxglQtkqJoIUUZMUE	Sarchand
complaint	BBvKGmXrMENznRhQXhZVPzAgPpeylfYtDxJ	Fleaur
complaint	ZJfpCRXZsFjUmxAJLuXRrStvZnVPURPzWVj	Matos
complaint	sGjueWbtupSFwKZoqGJmyxAEXimZOaSMtgv	Smith
complaint	niVQMglzsjNJFbaByTVnPoVTbtGoVNtegq	Everett
complaint	BWsIeiKKKyJnXiwGchCOgFPXABzsWQDKbTi	Gates
complaint	LEiLyRUhxyzYDCsxxvSIkDOISEnUpLJTNG	Patel
complaint	mLBiENXJifUTjuDnIikKTXnzTGgqRsymkFk	Jones
complaint	wLUEAnEWyCVrwDzwcrBGGSiZWGVvSkEQvPH	Banda
complaint	owmhjVtfvWGcTItEsCnJSRqATYJntSdgtph	Seo
complaint	iGFOJqwpAeqjaOVEUFDPaQMqpIIFZaskqVk	Olsen

## 3- Find the Top Passengers with Largest Total Miles:

```
SELECT PASSENGER_NAME, ROUND(TOTAL_MILES,2)
FROM PASSENGER_DIM
WHERE ROWNUM <= 10
ORDER BY TOTAL_MILES DESC;
```

PASSENGER_NAME	ROUND(TOTAL_MILES,2)
Alexander	98023.14
Lex	96726.24
Valli	95237.49
Neena	88502.7
David	61680.28
Diana	56980.64
Ismael	48468.51
Daniel	35633.94
Bruce	27171.78
John	23385.9

#### 4- Find the Total Airline Companies and Their Profit:

```
SELECT f.COMANY_NAME, SUM(r.TOTAL_PRICE) profit
FROM FLIGHT_DIM f JOIN RESERVATION_FACT r
ON f.FLIGHT_KEY = r.FLIGHT_KEY
GROUP BY COMANY_NAME
ORDER BY SUM(r.TOTAL_PRICE) DESC;
```

COMANY_NAME	PROFIT
Egyptair	58920
Sunclass Airlines	26141.2
Japan Airlines	11825
tigerair Australia	4335
Nordwind Airlines	3875
American Airlines	3850
Transavia Airlines	2075
Hawaiian Airlines	1245
Lion Airlines	750
Emirates	540
Aegean Airlines	330

#### 5- Find the Number of Passengers for Each Airline Company:

```
SELECT f.COMANY_NAME, COUNT(p.PASSENGER_KEY) "Number of Passengers"
FROM FLIGHT_DIM f JOIN RESERVATION_FACT r
ON f.FLIGHT_KEY = r.FLIGHT_KEY
JOIN PASSENGER_DIM p
ON r.PASSENGER_KEY = p.PASSENGER_KEY
GROUP BY COMANY_NAME
ORDER BY COUNT(p.PASSENGER_KEY) DESC;
```

COMANY_NAME	Number of Passengers
Egyptair	45
Sunclass Airlines	34
American Airlines	6
Japan Airlines	6
tigerair Australia	4
Transavia Airlines	4
Nordwind Airlines	2
Hawaiian Airlines	2
Aegean Airlines	1
Emirates	1
Lion Airlines	1

## 6- Find the Number of Flights for Each Airline Company:

```
SELECT COMANY_NAME, COUNT(FLIGHT_KEY) "Number of Flights"  
FROM FLIGHT_DIM  
GROUP BY COMANY_NAME  
ORDER BY COUNT(FLIGHT_KEY) DESC;
```

COMANY_NAME	Number of Flights
Egyptair	45
Sunclass Airlines	34
American Airlines	6
Japan Airlines	6
tigerair Australia	4
Transavia Airlines	4
Nordwind Airlines	2
Hawaiian Airlines	2
Aegean Airlines	1
Emirates	1
Lion Airlines	1

## 7- Find The Total Sales from Each Payment Method:

```
SELECT c.PAYMENT_METHOD "Payment Method", SUM(r.TOTAL_PRICE) profit  
FROM RESERVATION_CHANNEL_DIM c JOIN RESERVATION_FACT r  
ON c.CHANNEL_KEY = r.RESERVATION_CHANNEL_KEY  
GROUP BY c.PAYMENT_METHOD  
ORDER BY SUM(r.TOTAL_PRICE) DESC;
```

Payment Method	PROFIT
visa	47805
cash	38741.2
bank	27340

## 8- Find the Total Sales from Each Reservation Channel:

```
SELECT reservation_channel_type AS "channel type", SUM(total_price) AS "total cost"  
FROM reservation_fact rf  
JOIN RESERVATION_CHANNEL_DIM rd  
ON rf.reservation_channel_key = rd.channel_key  
GROUP BY reservation_channel_type  
ORDER BY "total cost" DESC;
```

channel type	total cost
office	44020
website	36836.2
airport	33030

### 9- Find the Average Number of Nights That Passengers Stayed in Each Hotel:

```
SELECT h.HOTEL_NAME, TRUNC(AVG(n.NUMBER_OF_NIGHTS),0) "AVG number of nights"
FROM HOTEL_DIM h JOIN TRANSIT_HOTEL_FACT n
ON H.HOTEL_KEY = N.HOTEL_KEY
GROUP BY h.hotel_name
ORDER BY AVG(n.NUMBER_OF_NIGHTS) DESC;
```

HOTEL_NAME	AVG number of nights
Texas	6190
Distrito Federal,	6188
Maharashtra	6182
Geneve	6150
Yukon	6135
Utrecht	6112
New Jersey	6101
Ontario	6063
Sao Paulo	5887
BE	5849
California	5839
Bavaria	5754
Oxford	5657

### 10- Find The Most Received Classes at Each Company:

```
SELECT comany_name AS "company" , class_type AS "class"
FROM class_dim cd
JOIN reservation_fact rf
ON CD.CLASS_KEY = RF.CLASS_KEY
JOIN flight_dim fd
ON rf.flight_key= FD.FLIGHT_KEY
GROUP BY comany_name , class_type ;
```

iii	company	class
►	Transavia Airlines	Business Class
	American Airlines	Business Class
	Egyptair	First Class
	tigerair Australia	First Class
	Nordwind Airlines	First Class
	Emirates	Business Class
	tigerair Australia	Business Class
	Transavia Airlines	First Class
	Japan Airlines	First Class
	Aegean Airlines	Business Class
	Hawaiian Airlines	Business Class
	Egyptair	Business Class
	Lion Airlines	First Class
	Sunclass Airlines	Business Class
	Nordwind Airlines	Business Class
	American Airlines	First Class
	Sunclass Airlines	First Class
	Japan Airlines	Business Class

# 11- Find the Frequent Passengers and How Much Miles they Passed:

```

SELECT PASSENGER_ID,SUM( FLIGHT_MILES)as "flight miles "
FROM FLIGHT_dim
join reservation_fact on reservation_fact.flight_key = FLIGHT_DIM.FLIGHT_KEY
join passenger_dim on PASSENGER_DIM.PASSENGER_key =
RESERVATION_FACT.PASSENGER_KEY
GROUP BY PASSENGER_ID;

```

iii	PASSENGER_ID	flight miles
►	123	2752.89780
	129	5396.80030
	138	8856.18063
	151	3963.78714
	152	1591.63962
	153	6803.93392
	167	3379.95273
	168	5705.93007
	173	4125.95607
	179	2280.17409
	180	9999.04133
	198	3894.71997

## 12- Find The Percentage of Each Subscription Type:

```
SELECT SUB_STAT , ROUND(COUNT(SUB_STAT)/ (SELECT COUNT(*) FROM PASSENGER_dim)
*100,1)|| '%'
FROM PASSENGER_dim
GROUP BY SUB_STAT;
```

SUB_STAT	ROUND(COUNT(SUB_STAT)/(SELECTCO...
gold	28.3%
titanium	46.2%
platinum	25.5%