

**Project Report**

**On**

**Airline Ticket Reservation System**

(CSD 2206-5 DATABASE DESIGN AND SQL)

**Submitted to: Submitted by:**

Ms. Rachida Amjoun Lovepreet Kaur (C0763499)

Minh Phuong Ly (C0757038)

Aneesha Ann Aloysious (C0769743)

Honey Mathew (C0765178)

**CONTENTS**

[INTRODUCTION 3](#_Toc24760165)

[REQUIREMENT ANALYSIS 4](#_Toc24760166)

[**Overview**: 4](#_Toc24760167)

[Entity, Attribute and Relationship Identification 5](#_Toc24760168)

[**Entities and Attributes:** 5](#_Toc24760169)

[**Relationships:** 5](#_Toc24760170)

[CONCEPTUAL MODEL (E-R DIAGRAM) 6](#_Toc24760171)

[LOGICAL DESIGN 8](#_Toc24760172)

[**Relational schemas:** 8](#_Toc24760173)

[**Third Normalization Form:** 8](#_Toc24760174)

[PHYSICAL TABLE REPREESENTATION 9](#_Toc24760175)

[PHYSICAL MODEL 13](#_Toc24760172)

[**Database Creation:** 13](#_Toc24760173)

[**Business Rules:** 23](#_Toc24760174)

[**Tools Used:** 23](#_Toc24760173)

[**Applying Business Rules using Constraints:** 24](#_Toc24760174)

[**Records Insertion:** 26](#_Toc24760173)

[**Constraint Testing:** 36](#_Toc24760174)

[**SQL Queries:** 3](#_Toc24760174)9

# INTRODUCTION

Airline Ticket Reservation System is a database project that facilitate the reservation of the online air tickets through an effective and yet simple GUI for a normal passenger intending to travel in airways. The project is basically targeted all those people who would like to travel through air. Apart from reserving tickets, through this system a passenger can compare fares ‘from’ various cities ‘to’ various cities.

# REQUIREMENT ANALYSIS

## Overview:

The Airline Ticket Reservation System should fulfill the following requirements:

* Keeping records of different flights of a particular airline at various places.
* Keeping the records of a specific passengers like passport number, address and contact number etc.
* Keep record of total hours of a specific aircraft has served during a specific time interval.
* Provides details related to number of aircrafts belonging to a specific airline.
* The system will keep record of the total number of hours that a specific passenger has travelled during a specific time interval.
* The system will give information about the total number of hours that a specific airline has been running during a specific time interval.
* It provides list of all passengers who flew to a specific city during a specific time interval.
* Point out the most visited city during the last month.
* Generating a list of aircrafts that have not been in used from a specific source location.
* It creates list of airlines that run flight from a specific source to a destination.
* It provides the list of all options that a passenger can have when travelling from a source to a destination. This includes a connecting flight, for instance, a passenger is travelling from Toronto to Delhi and there is no direct flight, therefore, you have to find the options for this passenger.
* It provides the detailed description of minimum hours that it will take for a passenger to travel from a source city to a destination city. Again, consider the connecting flights as mentioned in item number 9, e.g. travelling from Toronto to Delhi.

# Entity, Attribute and Relationship Identification

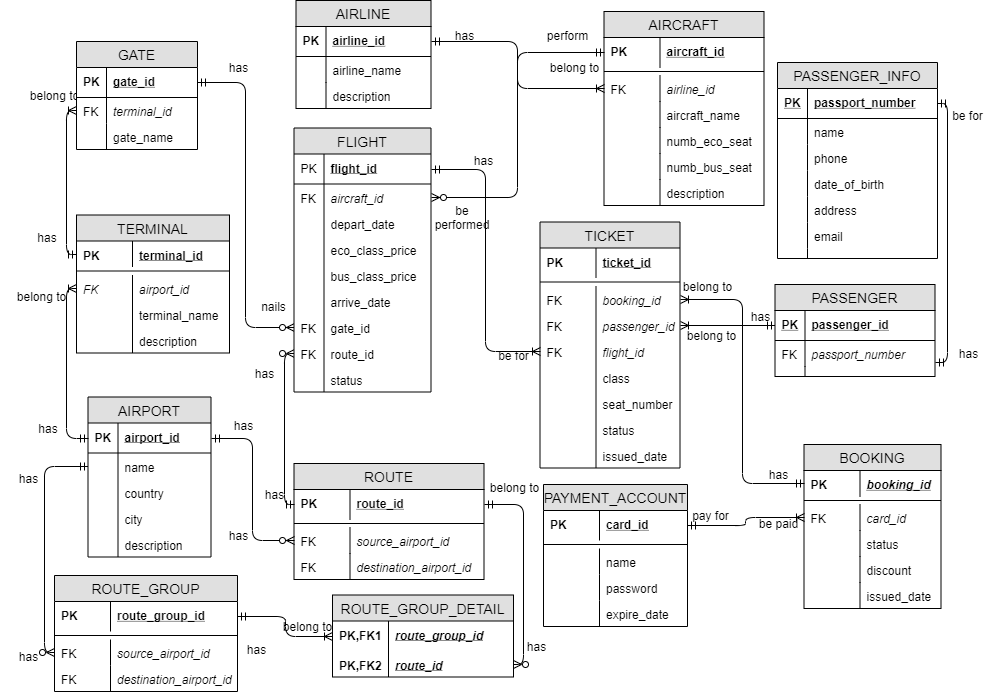
## **Entities and Attributes:**

1. TICKET has attributes: ticket\_id (primary key), booking\_id (foreign key), passenger\_id(foreign key), flight\_id(foreign key), class, seat\_number, status, issued\_date
2. FLIGHT stores all information of all flights and it has attributes: flight\_id(primary key), aircraft\_id(foreign key), depart\_date, eco\_class\_price, bus\_class\_price, arrival\_date, gate\_id(foreign key), route\_id(foreign key), status
3. AIRLINE has attributes: airline\_id(primary key), airline\_name, description
4. AIRCRAFT has attributes: aircraft\_id(primary key), airline\_id (foreign key), aircraft\_name, number\_eco\_seat, number\_bus\_seat, description
5. PASSENGER has attributes: passenger\_id(primary key), passport\_number(foreign key)
6. PASSENGER\_INFO has attributes: passport\_number(primary key), name, phone, date\_of\_birth, address, email
7. TERMINAL has attributes: terminal\_id(primary key), airport\_id(foreign key), terminal\_name, description
8. GATE has attributes: gate\_id(primary key), terminal\_id(foreign key), gate\_name
9. AIRPORT has attributes: airport\_id(primary key), name, country, city, description
10. PAYMENT\_ACCOUNT has attributes: card\_id(primary key), name, password, expire\_date
11. BOOKING has attributes: booking\_id(primary\_key), card\_id(foreign key), status, discount, issued\_date
12. ROUTE entity provides details about direct route from one source to a destination. It has attributes: route\_id(primary key), source\_airport\_id(foreign key), destination\_airport\_id(foreign key)
13. ROUTE\_GROUP entity is especially for getting information about connecting flights from one destination to other. It has attributes: route\_group\_id(primary key), source\_airport\_id(foreign key), destination\_airport\_id(foreign key)

## **Relationships:**

1. ROUTE\_GROUP\_DETAIL plays a role of solving many to many relationship problem between ROUTE and ROUTE\_GROUP entities. It has attributes: route\_group\_id ( primary key, foreign key), route\_id (primary key, foreign key)

# CONCEPTUAL MODEL (E-R DIAGRAM)

****

The ER diagram fulfils the following user requirements:

* The airline ticket reservation system can keep track of different flights of a particular airline at various places.
* Also, this system can keep the records of a specific passengers like passport number, address and contact number etc.
* The system can keep record of total hours of a specific aircraft has served during a specific time interval
* This can provide details related to number of aircrafts belonging to a specific airline
* The system can keep record of the total number of hours that a specific passenger has travelled during a specific time interval
* The system can give information about the total number of hours that a specific airline has been running during a specific time interval
* It provides list of all passengers who flew to a specific city during a specific time interval
* This ERD point out the most visited city during the last month
* It can generate a list of aircrafts that have **not** been in used from a specific source location
* It can create list of airlines that run flight from a specific source to a destination
* This airline system can provide the list of all options that a passenger can have when travelling from a source to a destination. This includes a connecting flight, for instance, a passenger is travelling from Toronto to Delhi and there is no direct flight, therefore, you have to find the options for this passenger
* This ERD results the detailed description of minimum hours that it will take for a passenger to travel from a source city to a destination city. Again, consider the connecting flights as mentioned in item number 9, e.g. travelling from Toronto to Delhi.

# LOGICAL DESIGN

**Relational schemas:**

TICKET (**ticket\_id**, *booking\_id, passenger\_id, flight\_id,* class, seat\_number, status, issued\_date)

PASSENGER (**passenger\_id**, *passport\_number*)

PASSENGER\_INFO (**passport\_number**, name, phone, date\_of\_birth, address, email)

GATE (**gate\_id**, *terminal\_id*, gate\_name)

TERMINAL (**terminal\_id**, *airport\_id*, terminal\_name, description)

BOOKING (**booking\_id**, *card\_id,* status, discount, issued\_date)

PAYMENT\_ACCOUNT (**card\_id**, name, password, expire\_date)

FLIGHT (**flight\_id,** *aircraft\_id,* depart\_date, eco\_class\_price, bus\_class\_price, arrive\_date, *route\_id*, *gate\_id*, status)

AIRLINE (**airline id,** airline\_name, description)

AIRCRAFT (**aircraft\_id,** *airline\_id,* aircraft\_name, numb\_eco\_seat, numb\_bus\_seat, description)

ROUTE (**route id,** *source\_airport\_id, destination\_airport\_id*)

ROUTE\_GROUP (**route group id,** *source\_airport\_id, destination\_airport\_id*)

ROUTE\_GROUP\_DETAIL (***route group id***, ***route id***)

AIRPORT (**airport id**, name, country, city, description)

## **Third Normalization Form:**

The ERD is already in third normalization form.

# Physical Table Representation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TERMINAL** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Terminal ID (PK) | NOT NULL | SMALLINT | 2 | terminal\_id |
| IATA Airport ID (FK) | NOT NULL | CHAR | 3 | airport\_id |
| Terminal name | NOT NULL | CHAR | 3 | terminal\_name |
| Description | NULL | VARCHAR | 10 | description |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GATE** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Gate ID (PK) | NOT NULL | SMALLINT | 2 | gate\_id |
| Terminal ID (FK) | NOT NULL | SMALLINT | 2 | terminal\_id |
| Name of the gate | NOT NULL | VARCHAR | 10 | gate\_name |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AIRPORT** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| IATA Airport ID (PK) | NOT NULL | CHAR | 3 | airport\_id |
| Name of airport | NOT NULL | VARCHAR | 50 | name |
| Country | NOT NULL | VARCHAR | 20 | country |
| City | NOT NULL | VARCHAR | 20 | city |
| Description | NULL | VARCHAR | 10 | description |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AIRLINE** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| IATA ID of Airline (PK) | NOT NULL | CHAR | 2 | airline\_id |
| Airline name | NOT NULL | VARCHAR | 50 | airline\_name |
| Description | NULL | VARCHAR | 50 | description |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AIRCRAFT** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Aircraft ID (PK) | NOT NULL | VARCHAR | 10 | aircraft\_id |
| IATA ID of Airline (FK) | NOT NULL | CHAR | 2 | airline\_id |
| Aircraft name | NOT NULL | VARCHAR | 50 | aircraft\_name |
| Number of eco class seat | NOT NULL | SMALLINT | 3 | number\_eco\_seat |
| Number of business class seat | NOT NULL | SMALLINT | 3 | number\_bus\_seat |
| Description | NULL | VARCHAR | 50 | description |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FLIGHT** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Flight ID (PK) | NOT NULL | VARCHAR | 10 | flight\_id |
| Aircraft ID (FK) | NOT NULL | VARCHAR | 10 | aircraft\_id |
| Route ID (FK) | NOT NULL | VARCHAR | 10 | route\_id |
| Gate ID (FK) | NOT NULL | SMALLINT | 2 | gate\_id |
| Departure date and time | NOT NULL | DATETIME |  | depart\_date |
| Arrival date and time | NOT NULL | DATETIME |  | arrive\_date |
| Price for economic class | NOT NULL | FLOAT | 15 | eco\_class\_price |
| Price for business class | NOT NULL | FLOAT | 15 | bus\_class\_price |
| Status of the flight | NOT NULL | VARCHAR | 10 | status |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROUTE** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Route ID (PK) | NOT NULL | VARCHAR | 10 | route\_id |
| IATA source Airport ID (FK) | NOT NULL | CHAR | 3 | source\_airport\_id |
| IATA destination Airport ID (FK) | NOT NULL | CHAR | 3 | destination\_airport\_id |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROUTE\_GROUP\_DETAIL** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Route ID (PK,FK) | NOT NULL | VARCHAR | 10 | route\_id |
| Route group ID (PK,FK) | NOT NULL | VARCHAR | 10 | route\_group\_id |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROUTE\_GROUP** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Route group ID (PK) | NOT NULL | VARCHAR | 10 | route\_group\_id |
| IATA source Airport ID (FK) | NOT NULL | CHAR | 3 | source\_airport\_id |
| IATA destination Airport ID (FK) | NOT NULL | CHAR | 3 | destination\_airport\_id |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TICKET** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Ticket ID (PK) | NOT NULL | VARCHAR | 20 | ticket\_id |
| Booking ID (FK) | NOT NULL | VARCHAR | 20 | booking\_id |
| Passenger ID (FK) | NOT NULL | VARCHAR | 20 | passenger\_id |
| Flight ID (FK) | NOT NULL | VARCHAR | 10 | flight\_id |
| Class of the ticket | NOT NULL | VARCHAR | 5 | class |
| The seat number | NULL | VARCHAR | 3 | seat\_number |
| Status of the ticket | NOT NULL | VARCHAR | 10 | status |
| Issued date and time | NOT NULL | DATETIME |  | issued\_date |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PASSENGER** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Passenger ID (PK) | NOT NULL | VARCHAR | 20 | passenger\_id |
| Passort number(FK) | NOT NULL | VARCHAR | 20 | passport\_number |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PASSENGER\_INFO** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Passort number (PK) | NOT NULL | VARCHAR | 20 | passport\_number |
| Passenger name | NOT NULL | VARCHAR | 30 | name |
| Contact phone number | NOT NULL | CHAR | 10 | phone |
| Date of birth | NOT NULL | DATE |  | date\_of\_birth |
| Address | NOT NULL | VARCHAR | 50 | address |
| Contact email | NOT NULL | VARCHAR | 30 | email |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BOOKING** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Booking ID (PK) | NOT NULL | VARCHAR | 20 | booking\_id |
| Card ID (FK) | NOT NULL | VARCHAR | 20 | card\_id |
| Status of booking | NULL | VARCHAR | 10 | status |
| Discount | NOT NULL | FLOAT | 4 | discount |
| Booking issued date and time | NOT NULL | DATETIME |  | issued\_date |

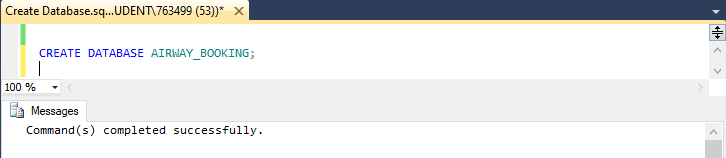
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PAYMENT\_ACCOUNT** | | | | |
| **Column Description** | **Optionality** | **Data Type** | **Size, Digits** | **SQL short name** |
| Card ID (PK) | NOT NULL | VARCHAR | 20 | card\_id |
| Name on the card | NOT NULL | VARCHAR | 30 | name |
| Password | NOT NULL | SMALLINT | 3 | password |
| Card expire date | NOT NULL | DATE |  | expire\_date |

# PHYSICAL MODEL

# Database Creation:

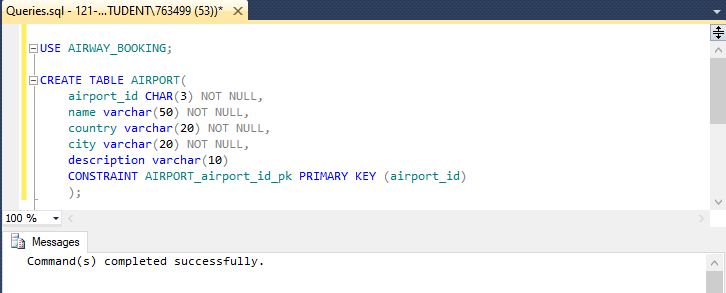
1. Create database with name AIRWAY\_BOOKING.

|  |
| --- |
| CREATE DATABASE AIRWAY\_BOOKING; |



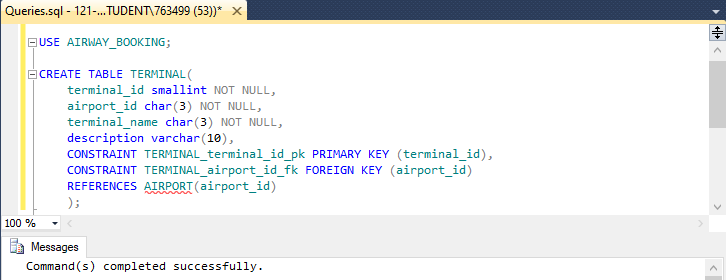
1. Create table AIRPORT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE AIRPORT(  airport\_id CHAR(3) NOT NULL,  name varchar(50) NOT NULL,  country varchar(20) NOT NULL,  city varchar(20) NOT NULL,  description varchar(10)  CONSTRAINT AIRPORT\_airport\_id\_pk PRIMARY KEY (airport\_id)  ); |



1. Create table TERMINAL.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE TERMINAL(  terminal\_id smallint NOT NULL,  airport\_id char(3) NOT NULL,  terminal\_name char(3) NOT NULL,  description varchar(10),  CONSTRAINT TERMINAL\_terminal\_id\_pk PRIMARY KEY (terminal\_id),  CONSTRAINT TERMINAL\_airport\_id\_fk FOREIGN KEY (airport\_id)  REFERENCES AIRPORT(airport\_id)  ); |



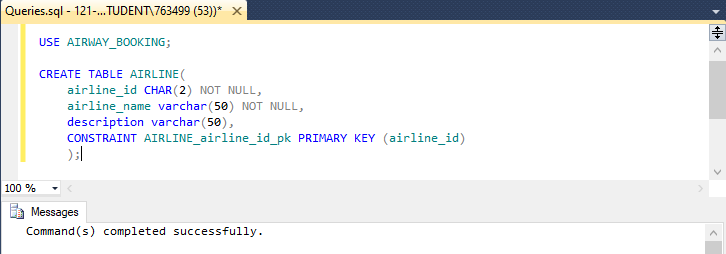
1. Create table GATE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE GATE(  gate\_id smallint NOT NULL,  terminal\_id smallint NOT NULL,  gate\_name varchar(10) NOT NULL,  CONSTRAINT GATE\_gate\_id\_pk PRIMARY KEY (gate\_id),  CONSTRAINT GATE\_terminal\_id\_fk FOREIGN KEY (terminal\_id)  REFERENCES TERMINAL(terminal\_id)  ); |

|  |
| --- |
|  |

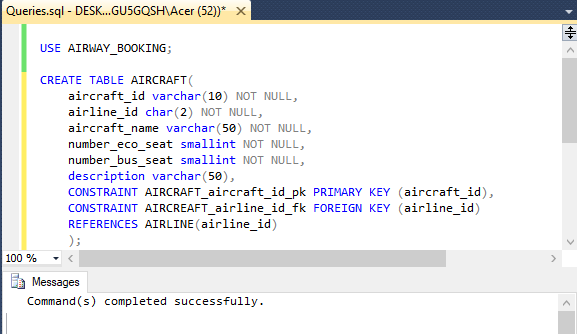
1. Create table AIRLINE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE AIRLINE(  airline\_id CHAR(2) NOT NULL,  airline\_name varchar(50) NOT NULL,  description varchar(50),  CONSTRAINT AIRLINE\_airline\_id\_pk PRIMARY KEY (airline\_id)  ); |



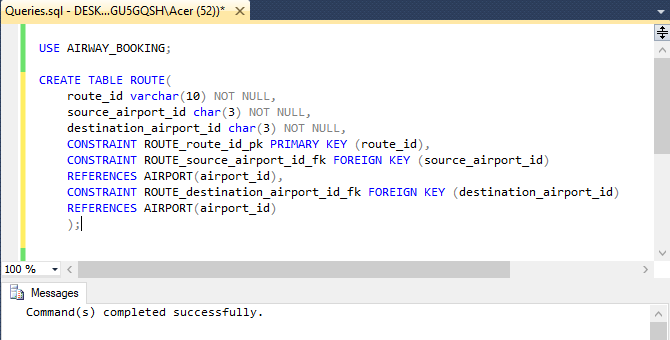
1. Create table AIRCRAFT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE AIRCRAFT(  aircraft\_id varchar(10) NOT NULL,  airline\_id char(2) NOT NULL,  aircraft\_name varchar(50) NOT NULL,  number\_eco\_seat smallint NOT NULL,  number\_bus\_seat smallint NOT NULL,  description varchar(50),  CONSTRAINT AIRCRAFT\_aircraft\_id\_pk PRIMARY KEY (aircraft\_id),  CONSTRAINT AIRCREAFT\_airline\_id\_fk FOREIGN KEY (airline\_id)  REFERENCES AIRLINE(airline\_id)  ); |



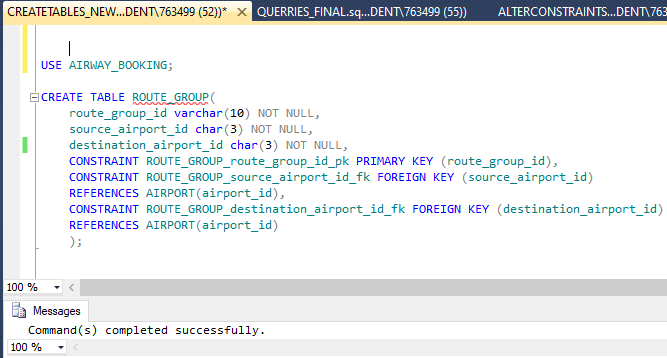
1. Create table ROUTE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE ROUTE(  route\_id varchar(10) NOT NULL,  source\_airport\_id char(3) NOT NULL,  destination\_airport\_id char(3) NOT NULL,  CONSTRAINT ROUTE\_route\_id\_pk PRIMARY KEY (route\_id),  CONSTRAINT ROUTE\_source\_airport\_id\_fk FOREIGN KEY (source\_airport\_id)  REFERENCES AIRPORT(airport\_id),  CONSTRAINT ROUTE\_destination\_airport\_id\_fk FOREIGN KEY(destination\_airport\_id)  REFERENCES AIRPORT(airport\_id)  ); |



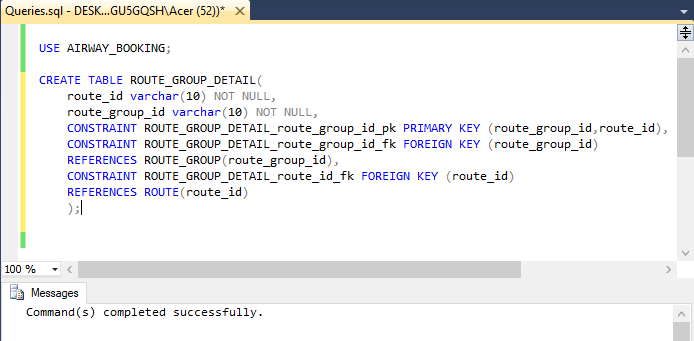
1. Create table ROUTE\_GROUP.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE ROUTE\_GROUP(  route\_group\_id varchar(10) NOT NULL,  source\_airport\_id char(3) NOT NULL,  destination\_airport\_id char(3) NOT NULL,  CONSTRAINT ROUTE\_GROUP\_route\_group\_id\_pk PRIMARY KEY (route\_group\_id),  CONSTRAINT ROUTE\_GROUP\_source\_airport\_id\_fk FOREIGN KEY (source\_airport\_id)  REFERENCES AIRPORT(airport\_id),  CONSTRAINT ROUTE\_GROUP\_destination\_airport\_id\_fk FOREIGN KEY (destination\_airport\_id)  REFERENCES AIRPORT(airport\_id)  ); |



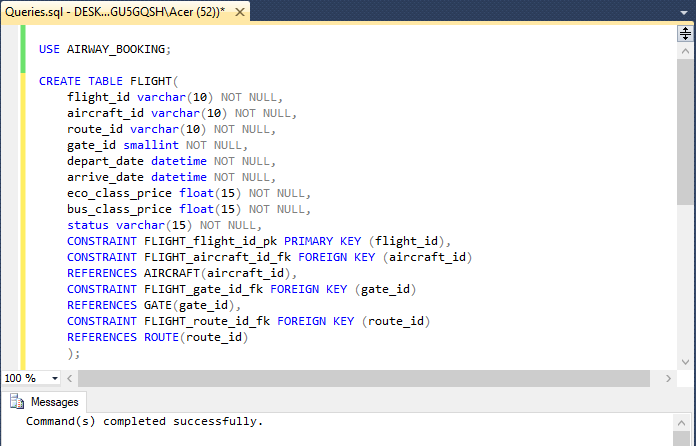
1. Create table ROUTE\_GROUP\_DETAIL.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE ROUTE\_GROUP\_DETAIL(  route\_id varchar(10) NOT NULL,  route\_group\_id varchar(10) NOT NULL,  CONSTRAINT ROUTE\_GROUP\_DETAIL\_route\_group\_id\_pk PRIMARY KEY (route\_group\_id,route\_id),  CONSTRAINT ROUTE\_GROUP\_DETAIL\_route\_group\_id\_fk FOREIGN KEY (route\_group\_id)  REFERENCES ROUTE\_GROUP(route\_group\_id),  CONSTRAINT ROUTE\_GROUP\_DETAIL\_route\_id\_fk FOREIGN KEY (route\_id)  REFERENCES ROUTE(route\_id)  ); |



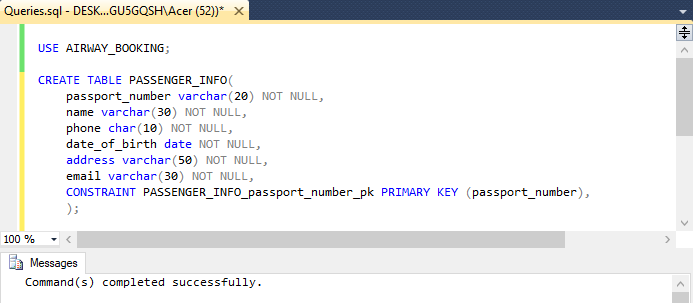
1. Create table FLIGHT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE FLIGHT(  flight\_id varchar(10) NOT NULL,  aircraft\_id varchar(10) NOT NULL,  route\_id varchar(10) NOT NULL,  gate\_id smallint NOT NULL,  depart\_date datetime NOT NULL,  arrive\_date datetime NOT NULL,  eco\_class\_price float(15) NOT NULL,  bus\_class\_price float(15) NOT NULL,  status varchar(15) NOT NULL,  CONSTRAINT FLIGHT\_flight\_id\_pk PRIMARY KEY (flight\_id),  CONSTRAINT FLIGHT\_aircraft\_id\_fk FOREIGN KEY (aircraft\_id)  REFERENCES AIRCRAFT(aircraft\_id),  CONSTRAINT FLIGHT\_gate\_id\_fk FOREIGN KEY (gate\_id)  REFERENCES GATE(gate\_id),  CONSTRAINT FLIGHT\_route\_id\_fk FOREIGN KEY (route\_id)  REFERENCES ROUTE(route\_id)  ); |



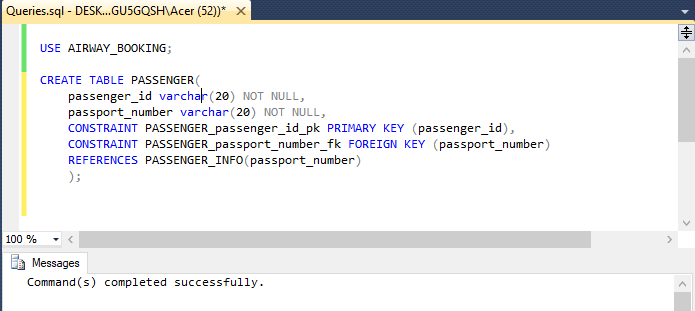
1. Create table PASSENGER\_INFO.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE PASSENGER\_INFO(  passport\_number varchar(20) NOT NULL,  name varchar(30) NOT NULL,  phone char(10) NOT NULL,  date\_of\_birth date NOT NULL,  address varchar(50) NOT NULL,  email varchar(30) NOT NULL,  CONSTRAINT PASSENGER\_INFO\_passport\_number\_pk PRIMARY KEY (passport\_number),  ); |



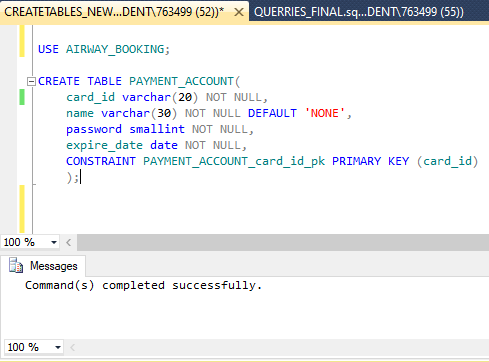
1. Create table PASSENGER.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE PASSENGER(  passenger\_id varchar(20) NOT NULL,  passport\_number varchar(20) NOT NULL,  CONSTRAINT PASSENGER\_passenger\_id\_pk PRIMARY KEY (passenger\_id),  CONSTRAINT PASSENGER\_passport\_number\_fk FOREIGN KEY (passport\_number)  REFERENCES PASSENGER\_INFO(passport\_number)  ); |



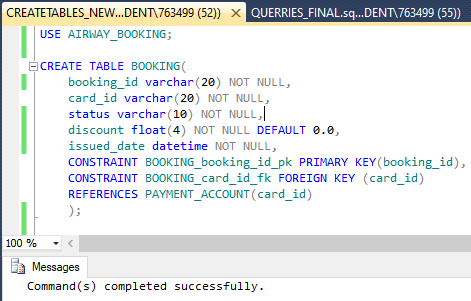
1. Create table PAYMENT\_ACCOUNT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE PAYMENT\_ACCOUNT(  card\_id varchar(20) NOT NULL,  name varchar(30) NOT NULL DEFAULT 'NONE',  password smallint NOT NULL,  expire\_date date NOT NULL,  CONSTRAINT PAYMENT\_ACCOUNT\_card\_id\_pk PRIMARY KEY (card\_id)  ); |



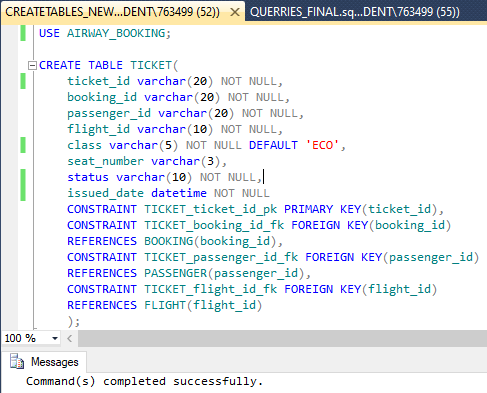
1. Create table BOOKING.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE BOOKING(  booking\_id varchar(20) NOT NULL,  card\_id varchar(20) NOT NULL,  status varchar(10) NOT NULL,  discount float(4) NOT NULL DEFAULT 0.0,  issued\_date datetime NOT NULL,  CONSTRAINT BOOKING\_booking\_id\_pk PRIMARY KEY(booking\_id),  CONSTRAINT BOOKING\_card\_id\_fk FOREIGN KEY (card\_id)  REFERENCES PAYMENT\_ACCOUNT(card\_id)  ); |



1. Create table TICKET.

|  |
| --- |
| USE AIRWAY\_BOOKING;  CREATE TABLE TICKET(  ticket\_id varchar(20) NOT NULL,  booking\_id varchar(20) NOT NULL,  passenger\_id varchar(20) NOT NULL,  flight\_id varchar(10) NOT NULL,  class varchar(5) NOT NULL DEFAULT 'ECO',  seat\_number varchar(3),  status varchar(10) NOT NULL,  issued\_date datetime NOT NULL  CONSTRAINT TICKET\_ticket\_id\_pk PRIMARY KEY(ticket\_id),  CONSTRAINT TICKET\_booking\_id\_fk FOREIGN KEY(booking\_id)  REFERENCES BOOKING(booking\_id),  CONSTRAINT TICKET\_passenger\_id\_fk FOREIGN KEY(passenger\_id)  REFERENCES PASSENGER(passenger\_id),  CONSTRAINT TICKET\_flight\_id\_fk FOREIGN KEY(flight\_id)  REFERENCES FLIGHT(flight\_id)  ); |



# BUSINESS RULES:

Every database is consisting of some rules and constraints to work properly and to avoid any incorrect data to get store. Business rules are a set of statements that specify what the user can or cannot do with the data inserted. Here are some constraints imposed on specific aspects of Airline Reservation System Database:

1. During payment, card expire date should be greater than current date
2. While booking a ticket, discount should not be less than 10.
3. In particular passenger information, every passenger must have unique email address.

# TOOLS USED

* SQL Server:

Every application requires a database system to store some data. SQL Server is a relational database management system (RDBMS) developed by Microsoft. It is primarily designed and developed to compete with MySQL and Oracle database. We have created our database using mysql queries.

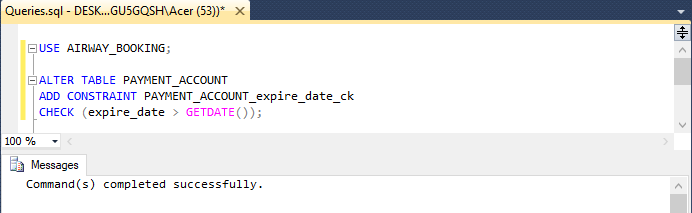
* DRAW.IO:

It is an open source, browser-based application used for drawing diagrams by user end. We have used this tool in creating ER diagram of the project.

# Applying Business Rules Using Constraints:

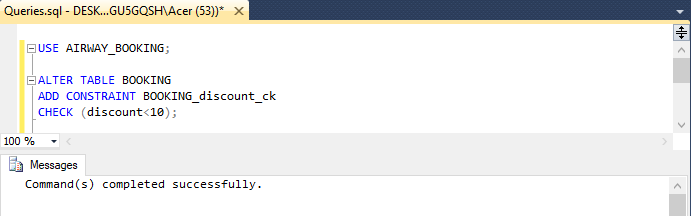
1. Setting constraint on card expiry date.

|  |
| --- |
| USE AIRWAY\_BOOKING;  ALTER TABLE PAYMENT\_ACCOUNT  ADD CONSTRAINT PAYMENT\_ACCOUNT\_expire\_date\_ck  CHECK (expire\_date > GETDATE()); |



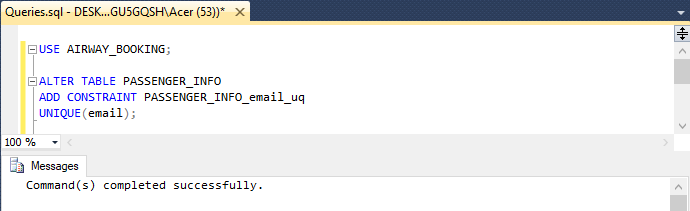
1. Setting constraint on booking discount.

|  |
| --- |
| USE AIRWAY\_BOOKING;  ALTER TABLE BOOKING  ADD CONSTRAINT BOOKING\_discount\_ck  CHECK (discount<10); |



1. Setting constraint on passenger’s email address.

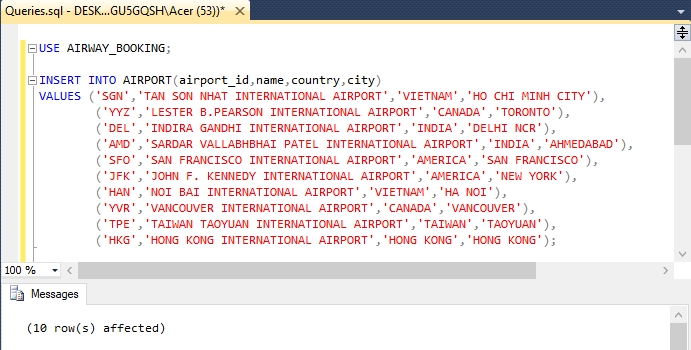
|  |
| --- |
| USE AIRWAY\_BOOKING;  ALTER TABLE PASSENGER\_INFO  ADD CONSTRAINT PASSENGER\_INFO\_email\_uq  UNIQUE(email); |



# Records insertion

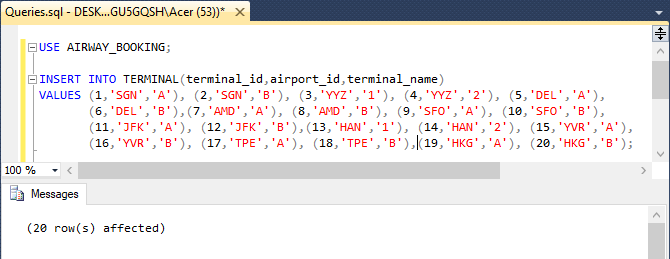
1. Insert data to AIRPORT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO AIRPORT(airport\_id,name,country,city)  VALUES ('SGN','TAN SON NHAT INTERNATIONAL AIRPORT','VIETNAM','HO CHI MINH CITY'),  ('YYZ','LESTER B.PEARSON INTERNATIONAL AIRPORT','CANADA','TORONTO'),  ('DEL','INDIRA GANDHI INTERNATIONAL AIRPORT','INDIA','DELHI NCR'),  ('AMD','SARDAR VALLABHBHAI PATEL INTERNATIONAL AIRPORT','INDIA','AHMEDABAD'),  ('SFO','SAN FRANCISCO INTERNATIONAL AIRPORT','AMERICA','SAN FRANCISCO'),  ('JFK','JOHN F. KENNEDY INTERNATIONAL AIRPORT','AMERICA','NEW YORK'),  ('HAN','NOI BAI INTERNATIONAL AIRPORT','VIETNAM','HA NOI'),  ('YVR','VANCOUVER INTERNATIONAL AIRPORT','CANADA','VANCOUVER'),  ('TPE','TAIWAN TAOYUAN INTERNATIONAL AIRPORT','TAIWAN','TAOYUAN'),  ('HKG','HONG KONG INTERNATIONAL AIRPORT','HONG KONG','HONG KONG'); |



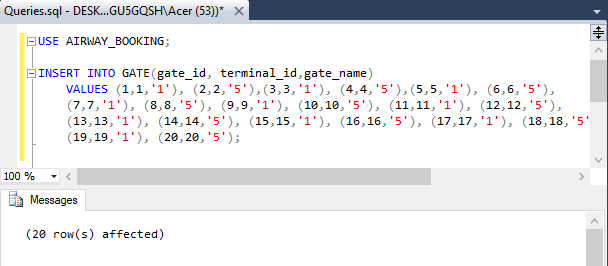
1. Insert data to TERMINAL.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO TERMINAL(terminal\_id,airport\_id,terminal\_name)  VALUES (1,'SGN','A'), (2,'SGN','B'), (3,'YYZ','1'), (4,'YYZ','2'),(5,'DEL','A'),  (6,'DEL','B'),(7,'AMD','A'), (8,'AMD','B'), (9,'SFO','A'), (10,'SFO','B'),  (11,'JFK','A'), (12,'JFK','B'),(13,'HAN','1'), (14,'HAN','2'), (15,'YVR','A'),  (16,'YVR','B'), (17,'TPE','A'), (18,'TPE','B'),(19,'HKG','A'), (20,'HKG','B'); |



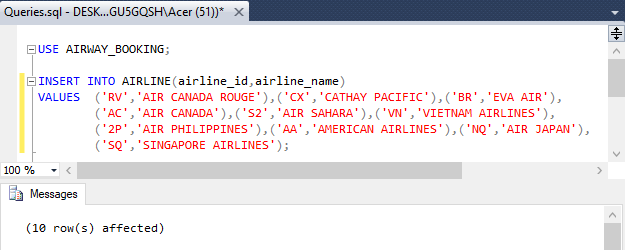
1. Insert data to GATE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO GATE(gate\_id, terminal\_id,gate\_name)  VALUES (1,1,'1'), (2,2,'5'),(3,3,'1'), (4,4,'5'),(5,5,'1'), (6,6,'5'),  (7,7,'1'), (8,8,'5'), (9,9,'1'), (10,10,'5'), (11,11,'1'), (12,12,'5'),  (13,13,'1'), (14,14,'5'), (15,15,'1'), (16,16,'5'), (17,17,'1'), (18,18,'5'),  (19,19,'1'), (20,20,'5'); |



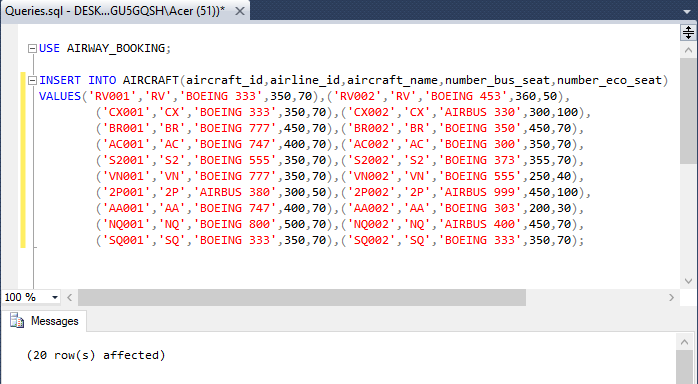
1. Insert data to AIRLINE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO AIRLINE(airline\_id,airline\_name)  VALUES('RV','AIR CANADA ROUGE'),('CX','CATHAY PACIFIC'),('BR','EVA AIR'),  ('AC','AIR CANADA'),('S2','AIR SAHARA'),('VN','VIETNAM AIRLINES'),  ('2P','AIR PHILIPPINES'),('AA','AMERICAN AIRLINES'),('NQ','AIR JAPAN'),  ('SQ','SINGAPORE AIRLINES'); |



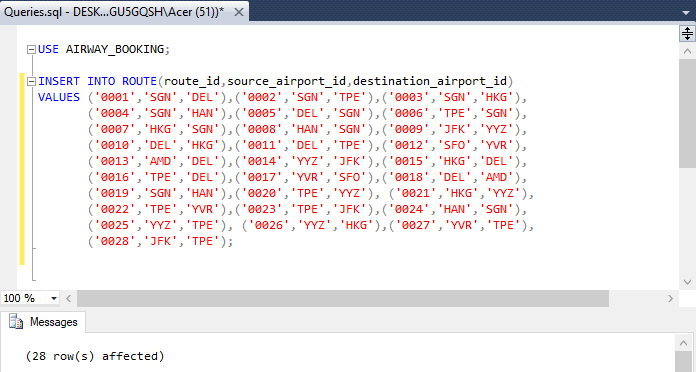
1. Insert data to AIRCRAFT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO AIRCRAFT(aircraft\_id,airline\_id,aircraft\_name,number\_bus\_seat,number\_eco\_seat)  VALUES('RV001','RV','BOEING 333',350,70),('RV002','RV','BOEING 453',360,50),  ('CX001','CX','BOEING 333',350,70),('CX002','CX','AIRBUS 330',300,100),  ('BR001','BR','BOEING 777',450,70),('BR002','BR','BOEING 350',450,70),  ('AC001','AC','BOEING 747',400,70),('AC002','AC','BOEING 300',350,70),  ('S2001','S2','BOEING 555',350,70),('S2002','S2','BOEING 373',355,70),  ('VN001','VN','BOEING 777',350,70),('VN002','VN','BOEING 555',250,40),  ('2P001','2P','AIRBUS 380',300,50),('2P002','2P','AIRBUS 999',450,100),  ('AA001','AA','BOEING 747',400,70),('AA002','AA','BOEING 303',200,30),  ('NQ001','NQ','BOEING 800',500,70),('NQ002','NQ','AIRBUS 400',450,70),  ('SQ001','SQ','BOEING 333',350,70),('SQ002','SQ','BOEING 333',350,70); |



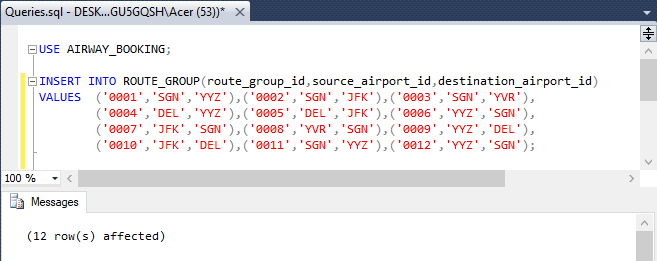
1. Insert data to ROUTE.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO ROUTE(route\_id,source\_airport\_id,destination\_airport\_id)  VALUES ('0001','SGN','DEL'),('0002','SGN','TPE'),('0003','SGN','HKG'),  ('0004','SGN','HAN'),('0005','DEL','SGN'),('0006','TPE','SGN'),  ('0007','HKG','SGN'),('0008','HAN','SGN'),('0009','JFK','YYZ'),  ('0010','DEL','HKG'),('0011','DEL','TPE'),('0012','SFO','YVR'),  ('0013','AMD','DEL'),('0014','YYZ','JFK'),('0015','HKG','DEL'),  ('0016','TPE','DEL'),('0017','YVR','SFO'),('0018','DEL','AMD'),  ('0019','SGN','HAN'),('0020','TPE','YYZ'), ('0021','HKG','YYZ'),  ('0022','TPE','YVR'),('0023','TPE','JFK'),('0024','HAN','SGN'),  ('0025','YYZ','TPE'), ('0026','YYZ','HKG'),('0027','YVR','TPE'),  ('0028','JFK','TPE'); |



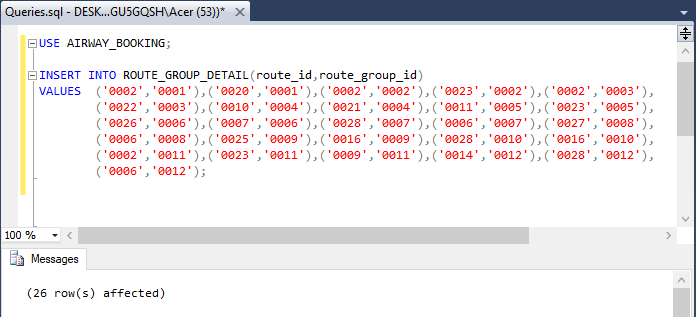
1. Insert data to ROUTE\_GROUP.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO ROUTE\_GROUP(route\_group\_id,source\_airport\_id,destination\_airport\_id)  VALUES ('0001','SGN','YYZ'),('0002','SGN','JFK'),('0003','SGN','YVR'),  ('0004','DEL','YYZ'),('0005','DEL','JFK'),('0006','YYZ','SGN'),  ('0007','JFK','SGN'),('0008','YVR','SGN'),('0009','YYZ','DEL'),  ('0010','JFK','DEL'),('0011','SGN','YYZ'),('0012','YYZ','SGN'); |



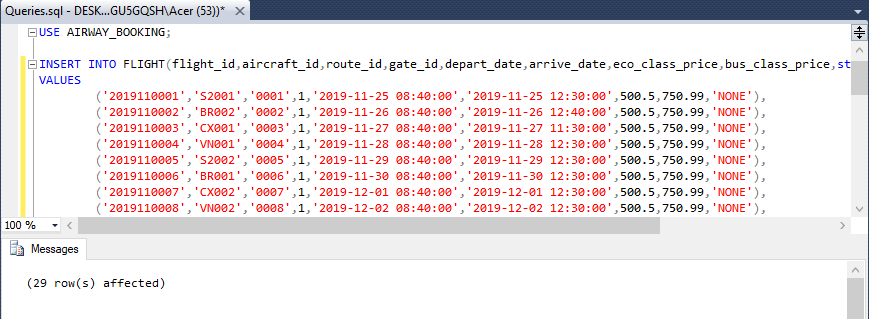
1. Insert data to ROUTE\_GROUP\_DETAIL.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO ROUTE\_GROUP\_DETAIL(route\_id,route\_group\_id)  VALUES ('0002','0001'),('0020','0001'),('0002','0002'),('0023','0002'), ('0002','0003'),('0022','0003'),('0010','0004'),('0021','0004'),  ('0011','0005'),('0023','0005'),('0026','0006'),('0007','0006'),  ('0028','0007'),('0006','0007'),('0027','0008'),('0006','0008'),  ('0025','0009'),('0016','0009'),('0028','0010'),('0016','0010'),  ('0002','0011'),('0023','0011'),('0009','0011'),('0014','0012'),  ('0028','0012'),('0006','0012'); |



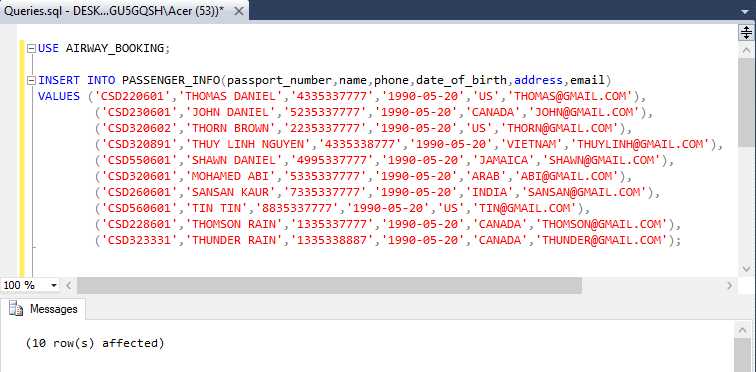
1. Insert data to FLIGHT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO FLIGHT(flight\_id,aircraft\_id,route\_id,gate\_id,depart\_date,arrive\_date,eco\_class\_price,bus\_class\_price,status)  VALUES  ('2019110001','S2001','0001',1,'2019-11-25 08:40:00','2019-11-25 12:30:00', 500.5,750.99,'NONE'),  ('2019110002','BR002','0002',1,'2019-11-26 08:40:00','2019-11-26 12:40:00', 500.5,750.99,'NONE'),  ('2019110003','CX001','0003',1,'2019-11-27 08:40:00','2019-11-27 11:30:00', 500.5,750.99,'NONE'),  ('2019110004','VN001','0004',1,'2019-11-28 08:40:00','2019-11-28 12:30:00', 500.5,750.99,'NONE'),  ('2019110005','S2002','0005',1,'2019-11-29 08:40:00','2019-11-29 12:30:00', 500.5,750.99,'NONE'),  ('2019110006','BR001','0006',1,'2019-11-30 08:40:00','2019-11-30 12:30:00', 500.5,750.99,'NONE'),  ('2019110007','CX002','0007',1,'2019-12-01 08:40:00','2019-12-01 12:30:00', 500.5,750.99,'NONE'),  ('2019110008','VN002','0008',1,'2019-12-02 08:40:00','2019-12-02 12:30:00', 500.5,750.99,'NONE'),  ('2019110009','AA001','0009',1,'2019-12-03 08:40:00','2019-12-03 12:30:00', 500.5,750.99,'NONE'),  ('2019110010','CX001','0010',1,'2019-12-04 08:40:00','2019-12-04 12:30:00', 500.5,750.99,'NONE'),  ('2019110011','BR001','0011',1,'2019-12-05 08:40:00','2019-12-05 12:30:00', 500.5,750.99,'NONE'),  ('2019110012','AC001','0012',1,'2019-12-06 08:40:00','2019-12-06 12:30:00', 500.5,750.99,'NONE'),  ('2019110013','S2001','0013',1,'2019-12-07 08:40:00','2019-12-07 12:30:00', 500.5,750.99,'NONE'),  ('2019110014','AC002','0014',1,'2019-12-08 08:40:00','2019-12-08 12:30:00', 500.5,750.99,'NONE'),  ('2019110015','CX002','0015',1,'2019-12-09 08:40:00','2019-12-09 12:30:00', 500.5,750.99,'NONE'),  ('2019110016','BR002','0016',1,'2019-12-10 08:40:00','2019-12-10 12:30:00', 500.5,750.99,'NONE'),  ('2019110017','RV001','0017',1,'2019-12-11 08:40:00','2019-12-11 12:30:00', 500.5,750.99,'NONE'),  ('2019110018','S2002','0018',1,'2019-12-12 08:40:00','2019-12-12 12:30:00', 500.5,750.99,'NONE'),  ('2019110019','VN002','0019',1,'2019-12-13 08:40:00','2019-12-13 12:30:00', 500.5,750.99,'NONE'),  ('2019110020','BR001','0020',1,'2019-12-14 08:40:00','2019-12-14 12:30:00', 500.5,750.99,'NONE'),  ('2019110021','CX001','0021',1,'2019-12-15 08:40:00','2019-12-15 12:30:00', 500.5,750.99,'NONE'),  ('2019110022','NQ001','0022',1,'2019-12-16 08:40:00','2019-12-16 12:30:00', 500.5,750.99,'NONE'),  ('2019110023','NQ002','0023',1,'2019-12-01 08:40:00','2019-12-01 12:30:00', 500.5,750.99,'NONE'),  ('2019110024','VN001','0024',1,'2019-12-18 08:40:00','2019-12-18 12:30:00', 500.5,750.99,'NONE'),  ('2019110025','RV002','0025',1,'2019-12-19 08:40:00','2019-12-19 12:30:00', 500.5,750.99,'NONE'),  ('2019110026','2P001','0026',1,'2019-12-20 08:40:00','2019-12-20 12:30:00', 500.5,750.99,'NONE'),  ('2019110027','2P002','0027',1,'2019-12-21 08:40:00','2019-12-21 12:30:00', 500.5,750.99,'NONE'),  ('2019110028','BR001','0028',1,'2019-12-22 08:40:00','2019-12-22 12:30:00', 500.5,750.99,'NONE'),  ('2019110029','2P001','0011',1,'2019-12-23 08:40:00','2019-12-23 12:30:00', 500.5,750.99,'NONE'); |



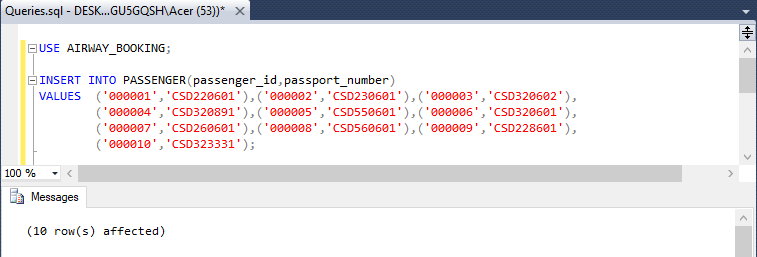
1. Insert data to PASSENGER\_INFO.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO PASSENGER\_INFO(passport\_number,name,phone,date\_of\_birth,address,email)  VALUES ('CSD220601','THOMAS DANIEL','4335337777','1990-05-20', 'US', 'THOMAS@GMAIL.COM'),  ('CSD230601','JOHN DANIEL','5235337777','1990-05-20','CANADA','JOHN@GMAIL.COM'),  ('CSD320602','THORN BROWN','2235337777','1990-05-20','US','THORN@GMAIL.COM'),  ('CSD320891','THUY LINH NGUYEN','4335338777','1990-05-20','VIETNAM','THUYLINH@GMAIL.COM'),  ('CSD550601','SHAWN DANIEL','4995337777','1990-05-20','JAMAICA','SHAWN@GMAIL.COM'),  ('CSD320601','MOHAMED ABI','5335337777','1990-05-20','ARAB','ABI@GMAIL.COM'),  ('CSD260601','SANSAN KAUR','7335337777','1990-05-20','INDIA','SANSAN@GMAIL.COM'),  ('CSD560601','TIN TIN','8835337777','1990-05-20','US','TIN@GMAIL.COM'),  ('CSD228601','THOMSON RAIN','1335337777','1990-05-20','CANADA','THOMSON@GMAIL.COM'),  ('CSD323331','THUNDER RAIN','1335338887','1990-05-20','CANADA','THUNDER@GMAIL.COM'); |



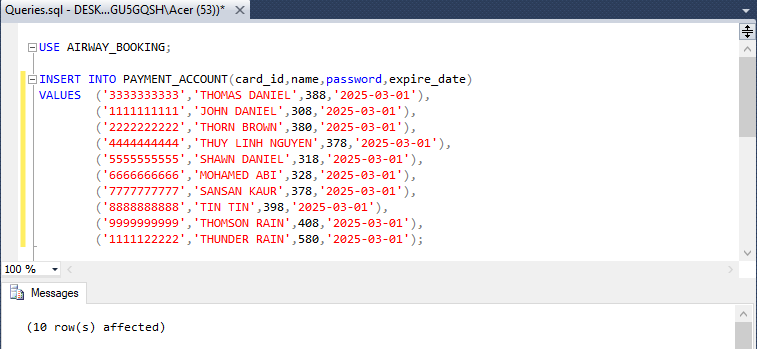
1. Insert data to PASSENGER.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO PASSENGER(passenger\_id,passport\_number)  VALUES ('000001','CSD220601'),('000002','CSD230601'),('000003','CSD320602'),  ('000004','CSD320891'),('000005','CSD550601'), ('000006','CSD320601'),  ('000007','CSD260601'),('000008','CSD560601'),('000009','CSD228601'),  ('000010','CSD323331'); |



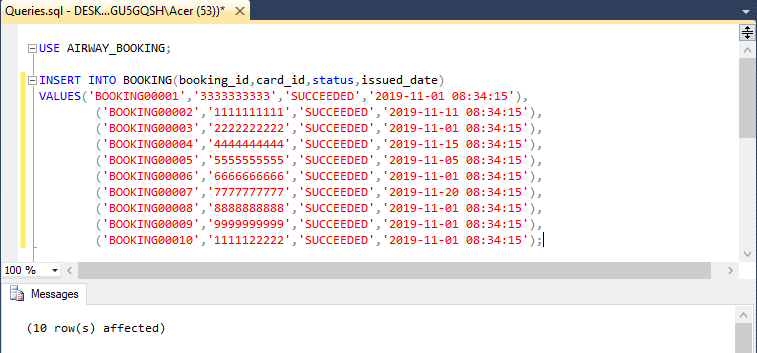
1. Insert to data to PAYMENT\_ACCOUNT.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO PAYMENT\_ACCOUNT(card\_id,name,password,expire\_date)  VALUES ('3333333333','THOMAS DANIEL',388,'2025-03-01'),  ('1111111111','JOHN DANIEL',308,'2025-03-01'),  ('2222222222','THORN BROWN',380,'2025-03-01'),  ('4444444444','THUY LINH NGUYEN',378,'2025-03-01'),  ('5555555555','SHAWN DANIEL',318,'2025-03-01'),  ('6666666666','MOHAMED ABI',328,'2025-03-01'),  ('7777777777','SANSAN KAUR',378,'2025-03-01'),  ('8888888888','TIN TIN',398,'2025-03-01'),  ('9999999999','THOMSON RAIN',408,'2025-03-01'),  ('1111122222','THUNDER RAIN',580,'2025-03-01'); |



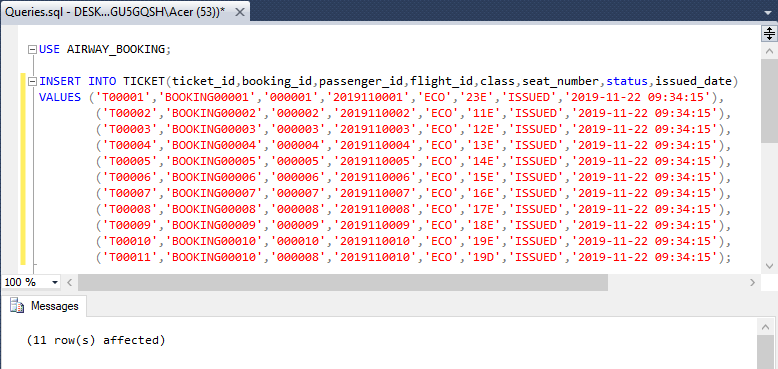
1. Insert data to BOOKING.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO BOOKING(booking\_id,card\_id,status,issued\_date)  VALUES('BOOKING00001','3333333333','SUCCEEDED','2019-11-01 08:34:15'),  ('BOOKING00002','1111111111','SUCCEEDED','2019-11-11 08:34:15'),  ('BOOKING00003','2222222222','SUCCEEDED','2019-11-01 08:34:15'),  ('BOOKING00004','4444444444','SUCCEEDED','2019-11-15 08:34:15'),  ('BOOKING00005','5555555555','SUCCEEDED','2019-11-05 08:34:15'),  ('BOOKING00006','6666666666','SUCCEEDED','2019-11-01 08:34:15'),  ('BOOKING00007','7777777777','SUCCEEDED','2019-11-20 08:34:15'),  ('BOOKING00008','8888888888','SUCCEEDED','2019-11-01 08:34:15'),  ('BOOKING00009','9999999999','SUCCEEDED','2019-11-01 08:34:15'),  ('BOOKING00010','1111122222','SUCCEEDED','2019-11-01 08:34:15'); |



1. Insert data to TICKET.

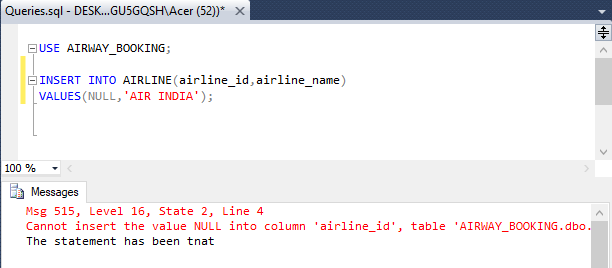
|  |
| --- |
| INSERT INTO TICKET(ticket\_id,booking\_id,passenger\_id,flight\_id,class,seat\_number,status,issued\_date)  VALUES ('T00001','BOOKING00001','000001','2019110001','ECO','23E','ISSUED','2019-11-22 09:34:15'),  ('T00002','BOOKING00002','000002','2019110002','ECO','11E','ISSUED','2019-11-22 09:34:15'),  ('T00003','BOOKING00003','000003','2019110003','ECO','12E','ISSUED','2019-11-22 09:34:15'),  ('T00004','BOOKING00004','000004','2019110004','ECO','13E','ISSUED','2019-11-22 09:34:15'),  ('T00005','BOOKING00005','000005','2019110005','ECO','14E','ISSUED','2019-11-22 09:34:15'),  ('T00006','BOOKING00006','000006','2019110006','ECO','15E','ISSUED','2019-11-22 09:34:15'),  ('T00007','BOOKING00007','000007','2019110007','ECO','16E','ISSUED','2019-11-22 09:34:15'),  ('T00008','BOOKING00008','000008','2019110008','ECO','17E','ISSUED','2019-11-22 09:34:15'),  ('T00009','BOOKING00009','000009','2019110009','ECO','18E','ISSUED','2019-11-22 09:34:15'),  ('T00010','BOOKING00010','000010','2019110010','ECO','19E','ISSUED','2019-11-22 09:34:15'),  ('T00011','BOOKING00010','000008','2019110010','ECO','19D','ISSUED','2019-11-22 09:34:15'); |



# Constraints Testing

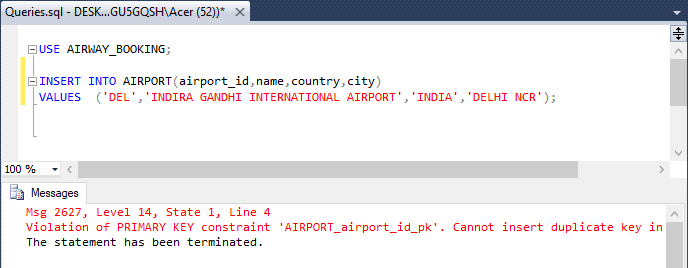
1. Testing of NULL constraint.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO AIRLINE(airline\_id,airline\_name) VALUES(NULL,'AIR INDIA'); |



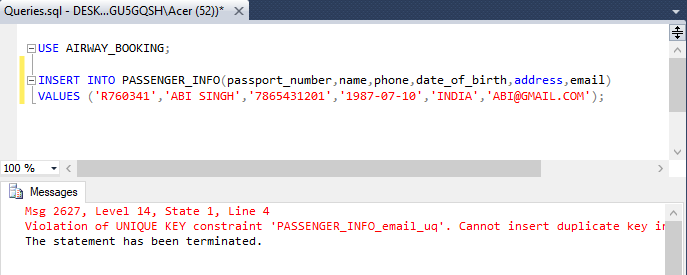
1. Testing of PRIMARY KEY constraint.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO AIRPORT(airport\_id,name,country,city)  VALUES ('DEL','INDIRA GANDHI INTERNATIONAL AIRPORT','INDIA','DELHI NCR'); |



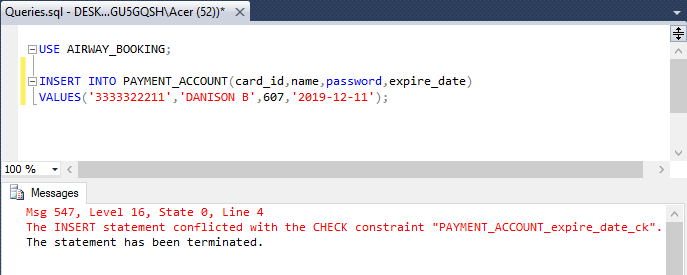
1. Testing of UNIQUE constraint.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO PASSENGER\_INFO(passport\_number,name,phone,date\_of\_birth,address,email)  VALUES ('R760341','ABI SINGH','7865431201','1987-07-10','INDIA','ABI@GMAIL.COM'); |



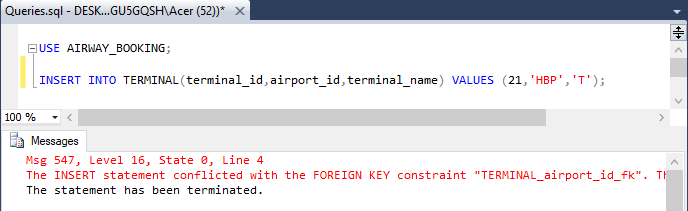
1. Testing of CHECK constraint.

|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO PAYMENT\_ACCOUNT(card\_id,name,password,expire\_date)  VALUES('3333322211','DANISON B',607,'2019-12-11'); |



1. Testing of FOREIGN KEY.

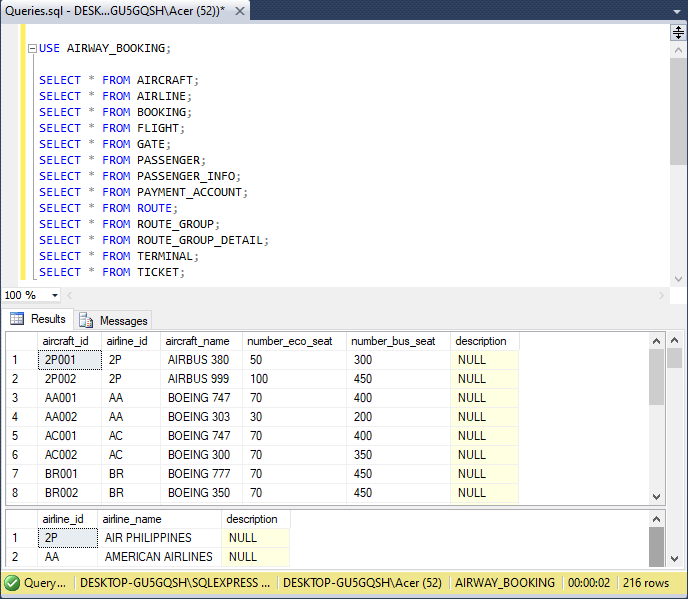
|  |
| --- |
| USE AIRWAY\_BOOKING;  INSERT INTO TERMINAL(terminal\_id,airport\_id,terminal\_name) VALUES (21,'HBP','T'); |

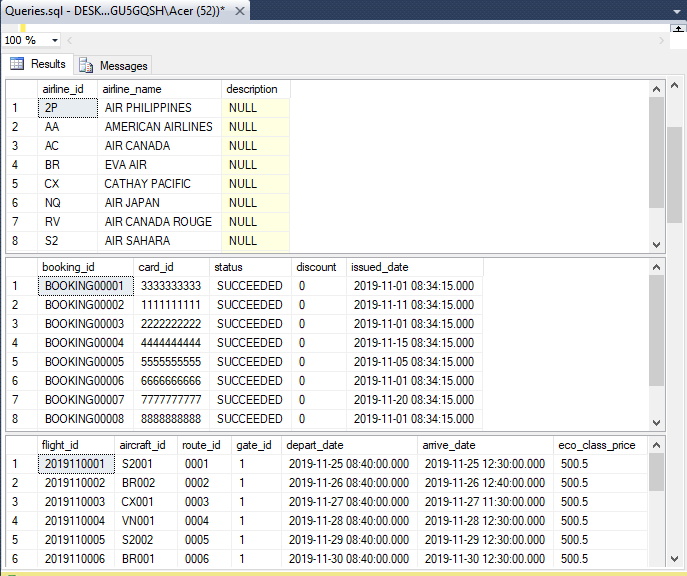


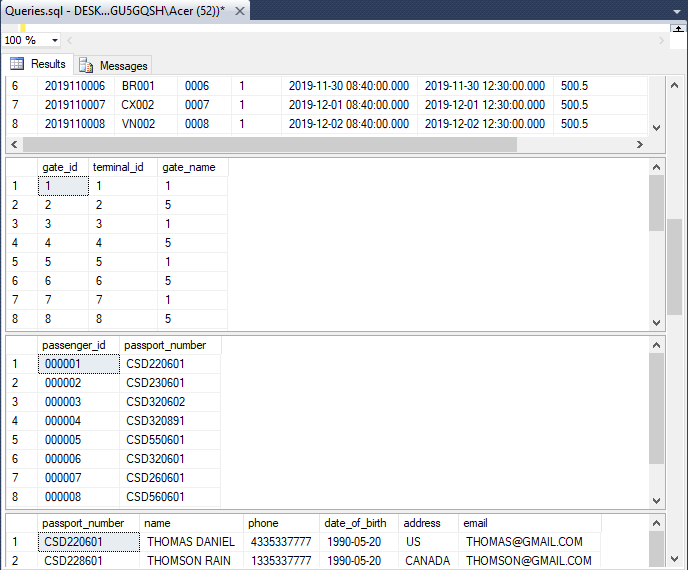
# SQL Queries

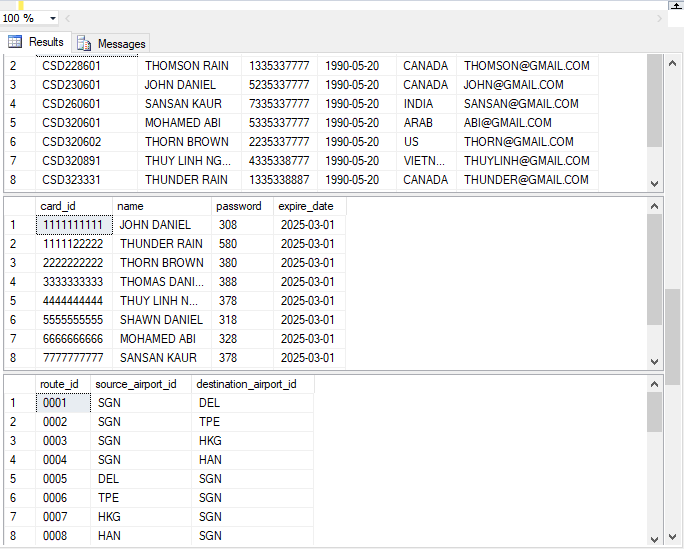
1. Data entry into tables that you have created.

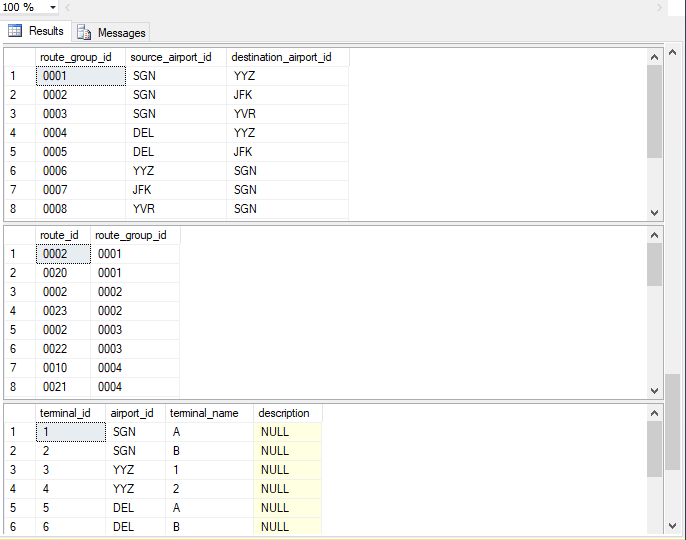
|  |
| --- |
| USE AIRWAY\_BOOKING;  SELECT \* FROM AIRCRAFT;  SELECT \* FROM AIRLINE;  SELECT \* FROM BOOKING;  SELECT \* FROM FLIGHT;  SELECT \* FROM GATE;  SELECT \* FROM PASSENGER;  SELECT \* FROM PASSENGER\_INFO;  SELECT \* FROM PAYMENT\_ACCOUNT;  SELECT \* FROM ROUTE;  SELECT \* FROM ROUTE\_GROUP;  SELECT \* FROM ROUTE\_GROUP\_DETAIL;  SELECT \* FROM TERMINAL;  SELECT \* FROM TICKET; |

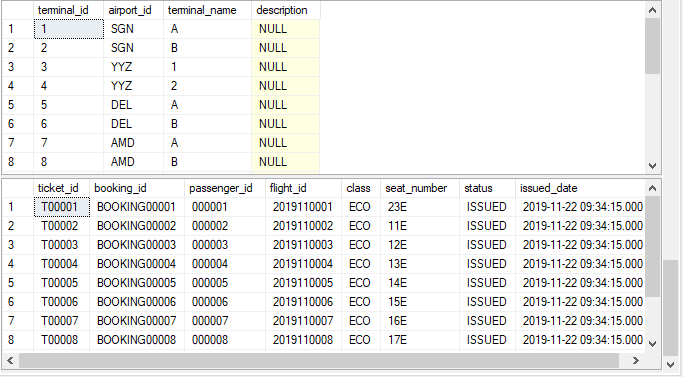






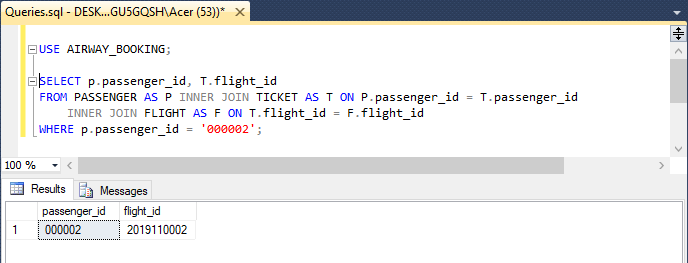






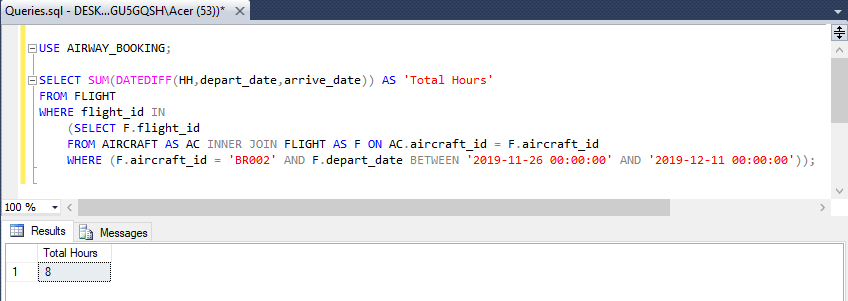
1. Travel history of a specific passenger.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q2 Travel history of a specific passenger\*/  /\*Performing: All flights that passenger\_id 000002 has flied\*/  SELECT p.passenger\_id, T.flight\_id  FROM PASSENGER AS P INNER JOIN TICKET AS T ON P.passenger\_id = T.passenger\_id  INNER JOIN FLIGHT AS F ON T.flight\_id = F.flight\_id  WHERE p.passenger\_id = '000002'; |



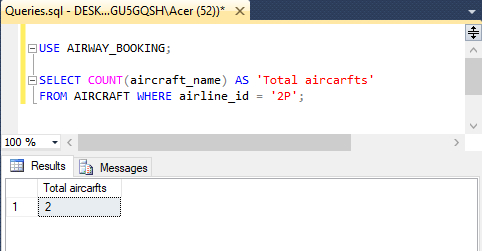
1. Total hours that a specific aircraft has served during a specific time interval.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q3 total hours that a specific aircarft has served during a specific time interval\*/  /\*Performing: Total hours aircraft\_if BR002 has served from Nov 26th 2019 to Dec 11th 2019\*/  SELECT SUM(DATEDIFF(HH,depart\_date,arrive\_date)) AS 'Total Hours'  FROM FLIGHT  WHERE flight\_id IN  (SELECT F.flight\_id  FROM AIRCRAFT AS AC INNER JOIN FLIGHT AS F ON AC.aircraft\_id = F.aircraft\_id  WHERE (F.aircraft\_id = 'BR002' AND F.depart\_date BETWEEN '2019-11-26 00:00:00' AND '2019-12-11 00:00:00')); |



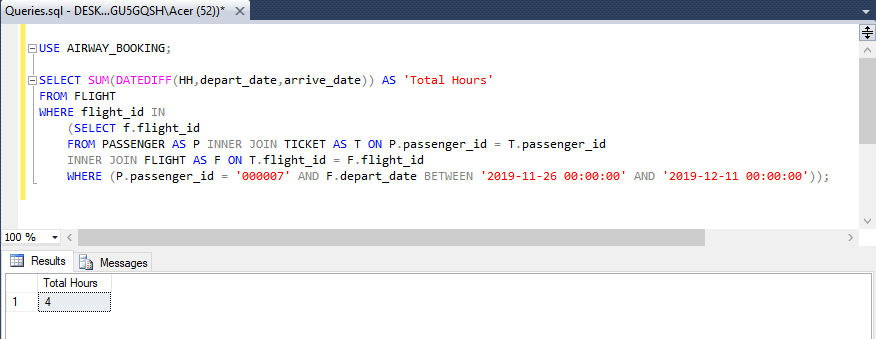
1. Total number of aircrafts belonging to a specific airline.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q4 number of aircarfts belong to a specific airline\*/  /\*Performing: Total aircraft that Philipines Airline(2P) possessing\*/  SELECT COUNT(aircraft\_name) AS 'Total aircarfts'  FROM AIRCRAFT WHERE airline\_id = '2P'; |



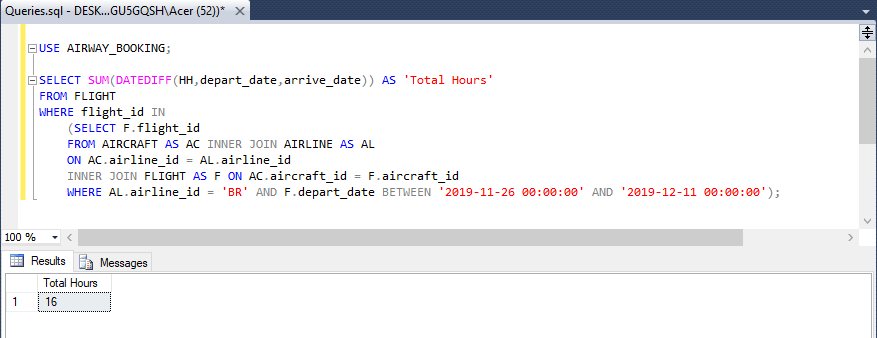
1. Total number of hours that a specific passenger has travelled during a specific time interval.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q5 total hours that a specific passenger has travel during a specific time interval\*/  /\*Performing: Total hours passesger\_if 000007 has traveled during the period of Nov 26th 219 - Dec 11th 2019\*/  SELECT SUM(DATEDIFF(HH,depart\_date,arrive\_date)) AS 'Total Hours'  FROM FLIGHT  WHERE flight\_id IN  (SELECT f.flight\_id  FROM PASSENGER AS P INNER JOIN TICKET AS T ON P.passenger\_id = T.passenger\_id  INNER JOIN FLIGHT AS F ON T.flight\_id = F.flight\_id  WHERE (P.passenger\_id = '000007' AND F.depart\_date BETWEEN '2019-11-26 00:00:00' AND '2019-12-11 00:00:00')); |



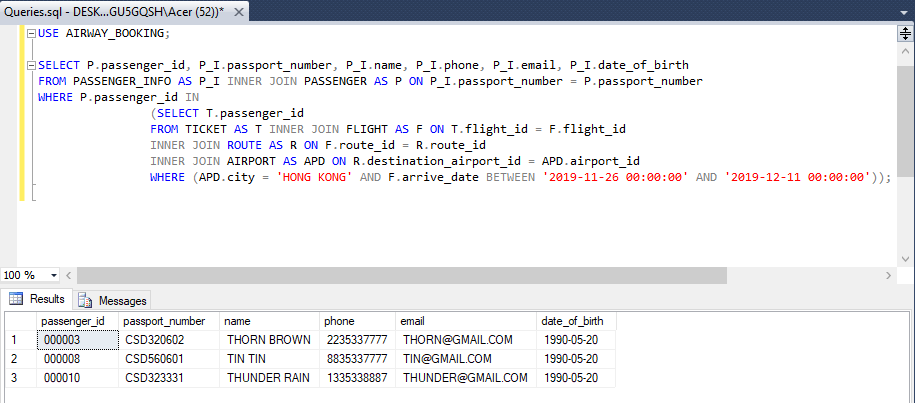
1. Total number of hours that a specific airline has been running during a specific time interval.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q6 Total number of hours that a specific airline has been running during a specific time interval\*/  /\*Performing: Total number of hours that Eva Airline has been running between Nov 26 2019 and Dec 11 2019\*/  SELECT SUM(DATEDIFF(HH,depart\_date,arrive\_date)) AS 'Total Hours'  FROM FLIGHT  WHERE flight\_id IN (SELECT F.flight\_id  FROM AIRCRAFT AS AC INNER JOIN AIRLINE AS AL  ON AC.airline\_id = AL.airline\_id  INNER JOIN FLIGHT AS F ON AC.aircraft\_id = F.aircraft\_id  WHERE AL.airline\_id = 'BR' AND F.depart\_date BETWEEN '2019-11-26 00:00:00'AND '2019-12-11 00:00:00'); |



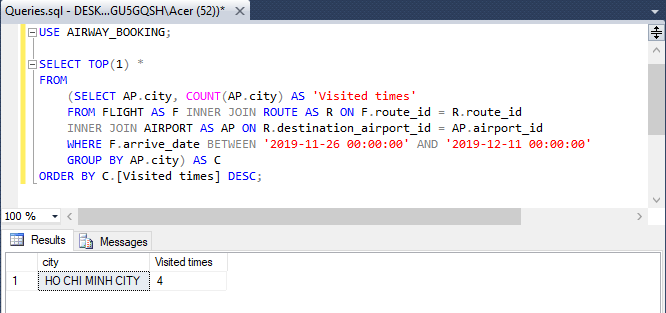
1. List of all passengers who flew to a specific city during a specific time interval.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q7 List of all passengers who flew to a specific city during a specific time interval\*/  /\*Performing: List all passgener who flew to Hong Kong between Nov 26 2019 and Dec 11 2019\*/  SELECT P.passenger\_id, P\_I.passport\_number, P\_I.name, P\_I.phone, P\_I.email, P\_I.date\_of\_birth  FROM PASSENGER\_INFO AS P\_I INNER JOIN PASSENGER AS P ON P\_I.passport\_number = P.passport\_number  WHERE P.passenger\_id IN  (SELECT T.passenger\_id FROM TICKET AS T INNER JOIN FLIGHT AS F ON T.flight\_id = F.flight\_id INNER JOIN ROUTE AS R ON F.route\_id = R.route\_id  INNER JOIN AIRPORT AS APD ON R.destination\_airport\_id = APD.airport\_id  WHERE (APD.city = 'HONG KONG' AND F.arrive\_date BETWEEN '2019-11-26 00:00:00' AND '2019-12-11 00:00:00')); |



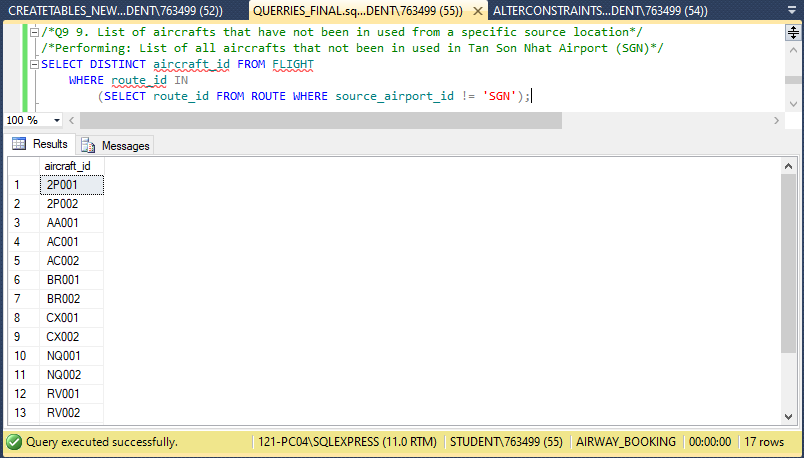
1. Most visited city during the last month.

|  |
| --- |
| USE AIRWAY\_ BOOKING;  /\*Q8 Most visited city during the last month.\*/  /\*Performing: Show the city was visited the most in the period from Nov 26 2019 to Dec 11 2019\*/  SELECT TOP(1) \*  FROM  (SELECT AP.city, COUNT(AP.city) AS 'Visited times'  FROM FLIGHT AS F INNER JOIN ROUTE AS R ON F.route\_id = R.route\_id  INNER JOIN AIRPORT AS AP ON R.destination\_airport\_id = AP.airport\_id  WHERE F.arrive\_date BETWEEN '2019-11-26 00:00:00' AND '2019-12-11 00:00:00'  GROUP BY AP.city) AS C  ORDER BY C.[Visited times] DESC; |



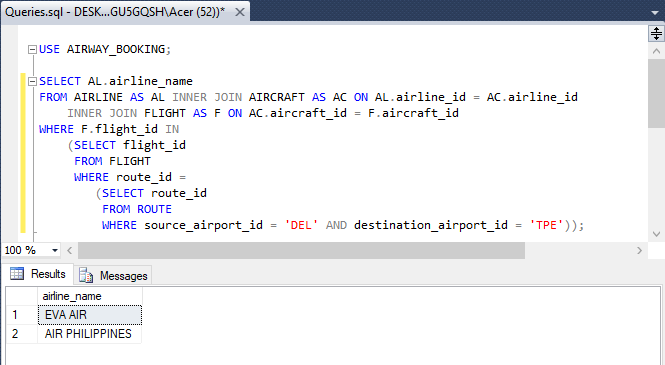
1. List of aircrafts that have **not** been in used from a specific source location.

|  |
| --- |
| USE AIRWAY\_BOOKING;  SELECT DISTINCT aircraft\_id FROM FLIGHT  WHERE route\_id IN  (SELECT route\_id FROM ROUTE WHERE source\_airport\_id != 'SGN'); |



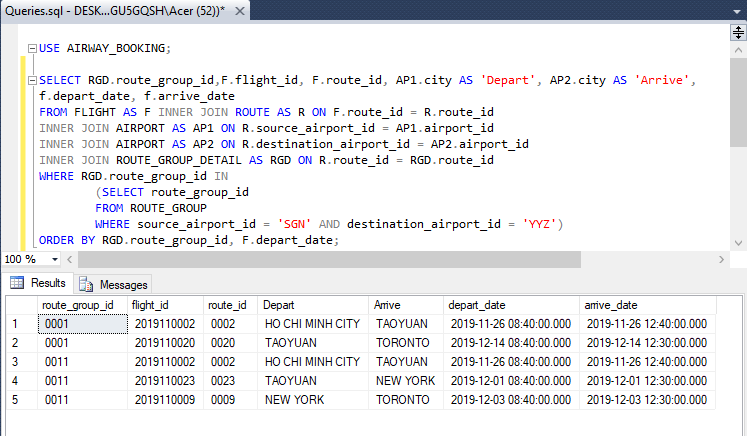
1. List of airlines that run flight from a specific source to a destination.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q10 List of airlines that run flight from a specific source to a destination.\*/  /\*Performing: List of all airlines have the flight from DEL (Delhi City) to TPE (TAIWAN)\*/  SELECT AL.airline\_name  FROM AIRLINE AS AL INNER JOIN AIRCRAFT AS AC ON AL.airline\_id = AC.airline\_id  INNER JOIN FLIGHT AS F ON AC.aircraft\_id = F.aircraft\_id  WHERE F.flight\_id IN  (SELECT flight\_id  FROM FLIGHT  WHERE route\_id =  (SELECT route\_id FROM ROUTE  WHERE source\_airport\_id = 'DEL' AND destination\_airport\_id = 'TPE')); |



1. The list of all options that a passenger can have when travelling from a source to a destination. This includes a connecting flight, for instance, a passenger is travelling from Toronto to Dehli and there is no direct flight, therefore, you have to find the options for this passenger.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q11 The list of all options that a passenger can have when travelling from a source to a destination.  This includes a connecting flight, for instance, a passenger is travelling from Toronto to Dehli and there is no  direct flight, therefore, you have to find the options for this passenger.\*/  /\*Performing: List all in-direct flights from SGN(Ho Chi Minh City) to YYZ(Toronto)\*/  SELECT RGD.route\_group\_id,F.flight\_id, F.route\_id, AP1.city AS 'Depart', AP2.city AS 'Arrive', f.depart\_date, f.arrive\_date  FROM FLIGHT AS F INNER JOIN ROUTE AS R ON F.route\_id = R.route\_id  INNER JOIN AIRPORT AS AP1 ON R.source\_airport\_id = AP1.airport\_id  INNER JOIN AIRPORT AS AP2 ON R.destination\_airport\_id = AP2.airport\_id  INNER JOIN ROUTE\_GROUP\_DETAIL AS RGD ON R.route\_id = RGD.route\_id  WHERE RGD.route\_group\_id IN  (SELECT route\_group\_id  FROM ROUTE\_GROUP  WHERE source\_airport\_id = 'SGN' AND destination\_airport\_id = 'YYZ')  ORDER BY RGD.route\_group\_id, F.depart\_date; |



1. What is the minimum number of hours that it will take for a passenger to travel from a source city to a destination city? Again, consider the connecting flights as mentioned in item number 9, e.g. travelling from Toronto to Dehli.

|  |
| --- |
| USE AIRWAY\_BOOKING;  /\*Q12 What is the minimum number of hours that it will take for a passenger to travel from a source city to a  destination city. Again, consider the connecting flights as mentioned in item number 9, e.g. travelling from Toronto to Dehli.\*/  /\*Performing: Amount of minimum hour it takes for flying from SGN to YYZ\*/  SELECT MIN(B.Total\_hours) AS 'Minimum number of hours' FROM  (SELECT STR(CAST(SUM(DATEDIFF(MINUTE,C.depart\_date,C.arrive\_date)) AS DECIMAL(10,2))/60,10,2) AS 'Total\_hours'  FROM  (SELECT RGD.route\_group\_id,F.flight\_id, F.route\_id, F.arrive\_date, F.depart\_date  FROM FLIGHT AS F INNER JOIN ROUTE AS R ON F.route\_id = R.route\_id  INNER JOIN AIRPORT AS AP1 ON R.source\_airport\_id = AP1.airport\_id  INNER JOIN AIRPORT AS AP2 ON R.destination\_airport\_id = AP2.airport\_id  INNER JOIN ROUTE\_GROUP\_DETAIL AS RGD ON R.route\_id = RGD.route\_id  WHERE RGD.route\_group\_id IN  (SELECT route\_group\_id  FROM ROUTE\_GROUP  WHERE source\_airport\_id = 'SGN' AND destination\_airport\_id = 'YYZ')) AS C  GROUP BY C.route\_group\_id) AS B; |

