1. **INTRODUCTION**

The purpose of this project is to depict how Indian Airports manage their databases. This projects aims to show a prototype of the actual database in use by the Airports. This project has been implemented in MySQL.

1. **PROBLEM STATEMENT**

The world which we have inherited from our forefathers relies increasingly upon a growing set of data and information to manage and prioritize the work around it. We, as future Software Developers, have the onus on ourselves to take up this challenge of creating and maintaining this data in an organized form. One such place where a huge dataset is created, manipulated and destroyed is the Indian airports. The Resulting Data Inconsistency by all of the processes mentioned above has to be solved by us. This is what we have tried to achieve in this project.

1. **FUNCTIONAL REQUIREMENTS**
2. **Admin Module**
   1. **Log in**

This module is used to log into the airport management system.

Input- Username, Password

Output- Admin dashboard

* 1. **Maintenance Schedule**

This module is used to maintain the flights and repairing as well as refueling.

Input- Maintenance\_id

Output- Hanger assigned and refueling required

* 1. **Manage and assign new Airport employees**

This module is used to add/remove and manage existing employees for the airport

Input- Employee and company name as the airport name

Output- Employee details or new employee added

1. **Employee Module**
   1. **Log in**

This module is used by the employees to access the airport management system.

Input- Username, Password

Output- Employee dashboard

* 1. **Booking and Customer Service**

This module lets the employees to assist the customer in there booking of tickets

Input- Customer\_id and Ticket\_id

Output- Booking details and customer details

* 1. **Payment Module**

This module helps the customer to pay for the flights and airport and the company to receive their fair share of the price paid. Also it enables the company to have a secure payment gateway.

Input- Transction\_id and amount paid and flight\_id

Output- Payment details and payment gateway used

1. **Company module:**
2. **Login:**

The company manger can login and view the current flight details, customers on the flight and the cabin crew for the flight.

1. **Manage Staff:**

The company manager can add and delete the staff for the company. He/she can also schedule the staff for the flights.

1. **Maintenance Schedule:**

The company manager can also manage the maintenance schedule for the flights.

1. **Customer module:**
2. **Login:**

The customer can login and view flight and ticket details at the airport check-in points.

1. **Book Tickets**

In this module the customer books tickets for the flights he/she wishes for.

1. **ENTITIES AND RELATIONSHIP**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity | Relationship | | Entity | |
| Customer | **A customer can book n no. of tickets and there is only 1 record of customer** | | **Ticket** | |
| Customer | **A customer can carry n no. of baggage** | | **Baggage** | |
| Customer | **A customer can have 1 parking spot in the airport parking** | | **Parking** | |
| Ticket | **Every ticket has only 1 transaction or payment gateway** | | **Transaction** | |
| Ticket | **Every ticket can have a flight corresponding to it** | | **Flight** | |
| Flight | **Every flight has an airplane crew** | | **Airplane crew** | |
| Airplane companies | **Every airplane company owns n no. of airplanes** | | **Airplanes** | |
| Airplane companies | | **Every airplane company owns an aviation clearance to fly in a particular area** | | **Aviation Clearance** | |
| Airplanes | | **Every airplane has n no. of maintenance scheduled over a period of time** | | **Maintenance Schedule** | |
| Flights | | **Every Flight has n no. of runway schedule for takeoff and landing** | | **Runway Schedule** | |
| Fights | | **Every flight has list of associated airports to which they are allowed to fly to for International Flights** | | **Associated airports** | |
| Airport Vendors | | **Every Airport Vendor has a Staff** | | **Airport Vendor Staff** | |
| Employees | | **Employees work for the Associated Companies operating at the airports.** | | **Airplane Companies** | |
| Employees | | **Responsible for Operation of flights** | | **Flights** | |

1. **RELATIONAL SCHEMA**
   1. **Airport (**Airport\_id, Airport\_name, City, No\_of\_runways**);**
   2. **Customer (**Customer\_id, Passport\_id, Customer\_name**);**
   3. **Baggage (**Baggage\_id, Belt\_no, Luggage\_items, Cabin\_item, Total\_weight, Exceeded\_Weight, Total\_amount\_exceeded, Customer\_id**);**
   4. **Flights (**Flight\_id, Source, Destination, Start\_time, End\_time**);**
   5. **Flight\_details (**Flight\_id, Airplane\_id, Airplane\_company\_id**);**
   6. **Transactions (**Transaction\_id, Payment\_Gateway, Amount\_paid, Ticket\_id**);**
   7. **Parking (**Parking\_id, Vehicle\_no, Time\_from, Time\_to, Total\_amount**);**
   8. **Tickets (**Ticket\_id, Seat\_no,Transaction\_id, Customer\_id, Flight\_id**);**
   9. **Employee (**Employee\_id, Employee\_name, Designation, Salary, Company\_id**);**
   10. **Airplane\_crew (**Crew\_id,Flight\_id, Captain\_emp\_id, FO\_emp\_id, SO\_emp\_id,**,** Air\_marshall\_name**);**
   11. **Airport\_vendor (**Vendor\_id, Vendor\_name**);**
   12. **Runway\_schedule (**Runway\_id, landing\_or\_takeoff, Runway\_no, flight\_id**);**
   13. **Aviation\_clearance (**Clearance\_id, From, To, Form\_no**);**
   14. **Airplanes (**Airplane\_id, Manufacture, No\_of\_flights, Estabilished, Clearance\_id**);**
   15. **Maintainance\_schedule (**Maintainance\_id, From, To, Hanger\_no, Problem, Fuel\_refill, Airplane\_id**);**
   16. **Airplane\_companies (**Company\_id, Company\_name, Company\_manager, Company\_owner, Clearance\_id**);**
   17. **Airplane\_hostess (**Hostess\_id,Crew\_id**);**
   18. **Associated\_airports (**Assoc\_airport\_id, City, Assoc\_airport\_name**);**

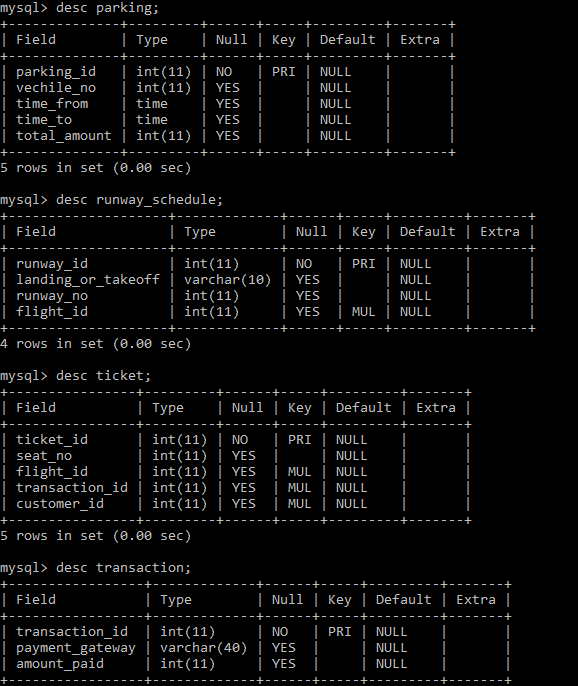
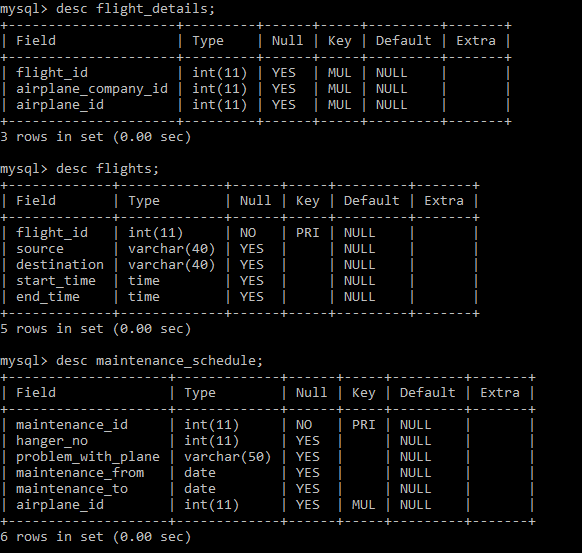
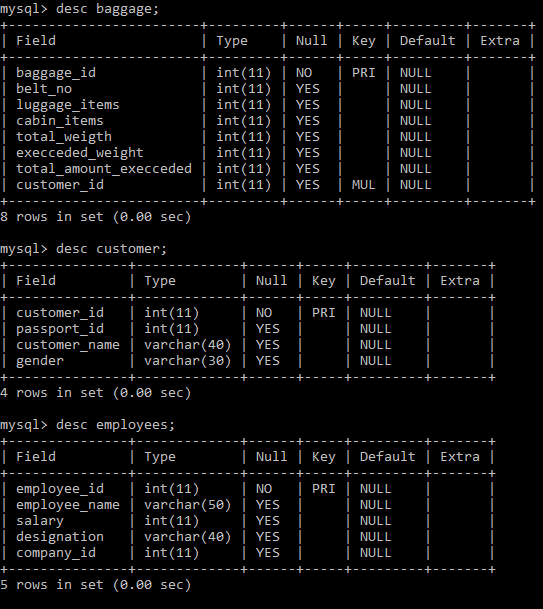
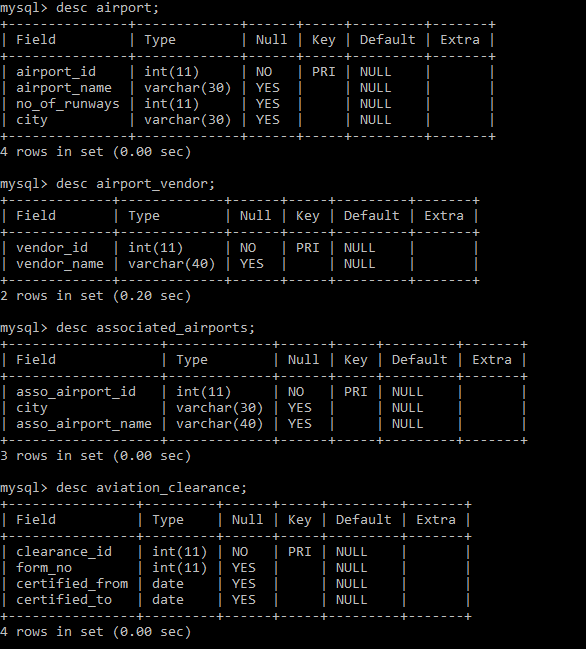
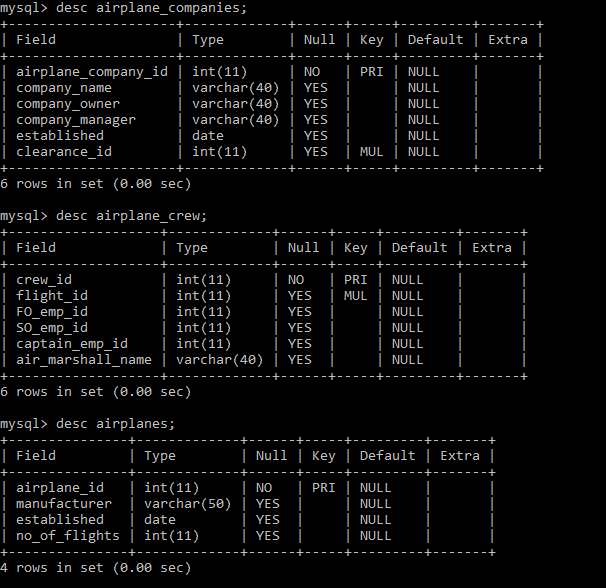
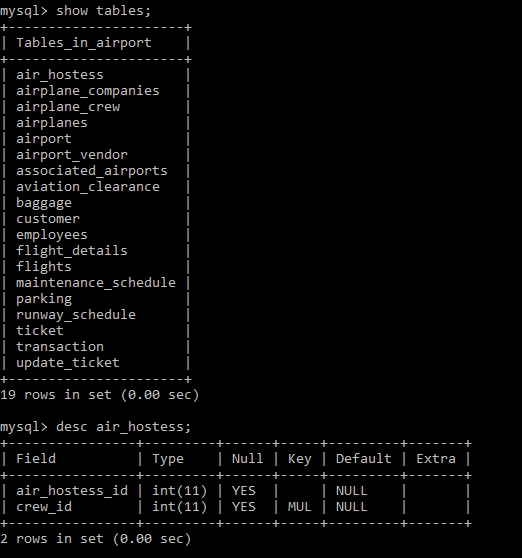
1. **KIND OF ANOMALIES IN THE TABLE**

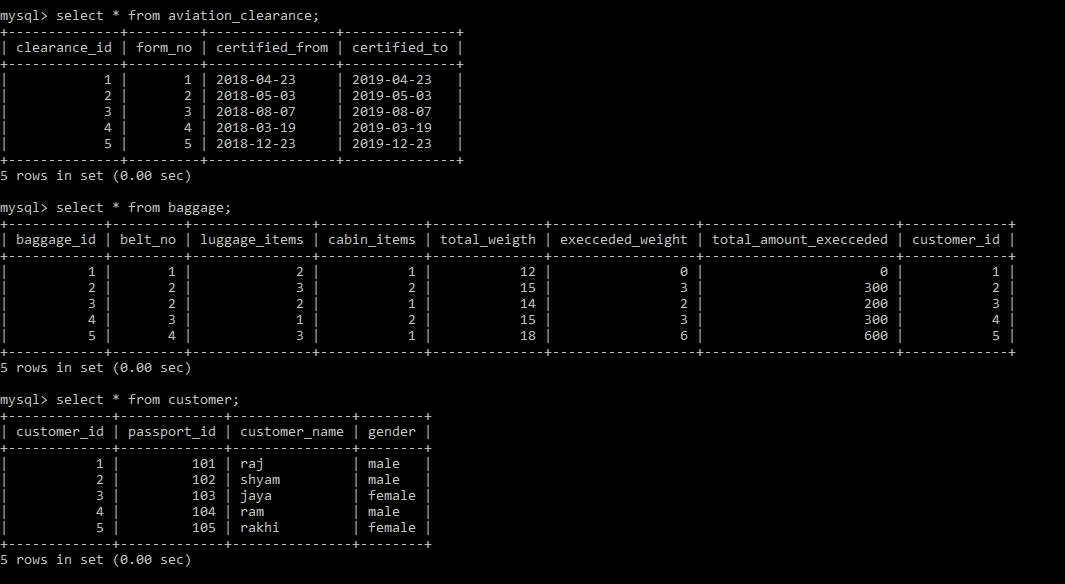
There are no anomalies in the relational schema.

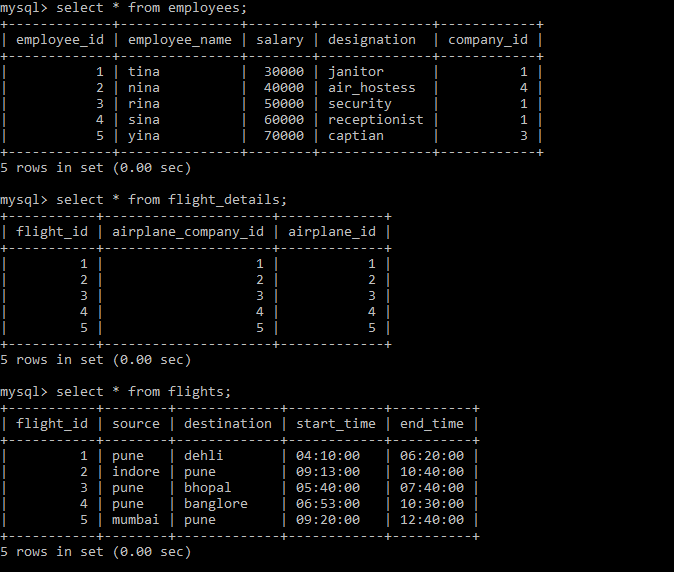
1. **FUNCTIONAL DEPENDENCY CHART OF EACH RELATION**

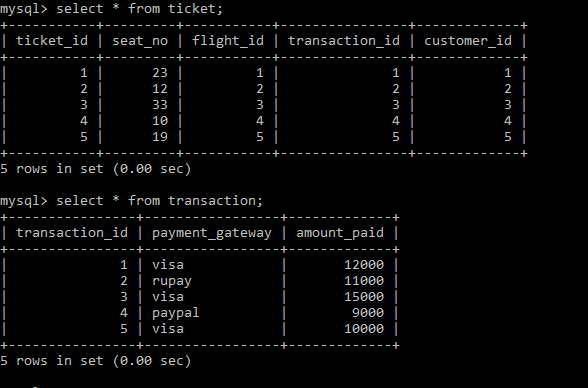
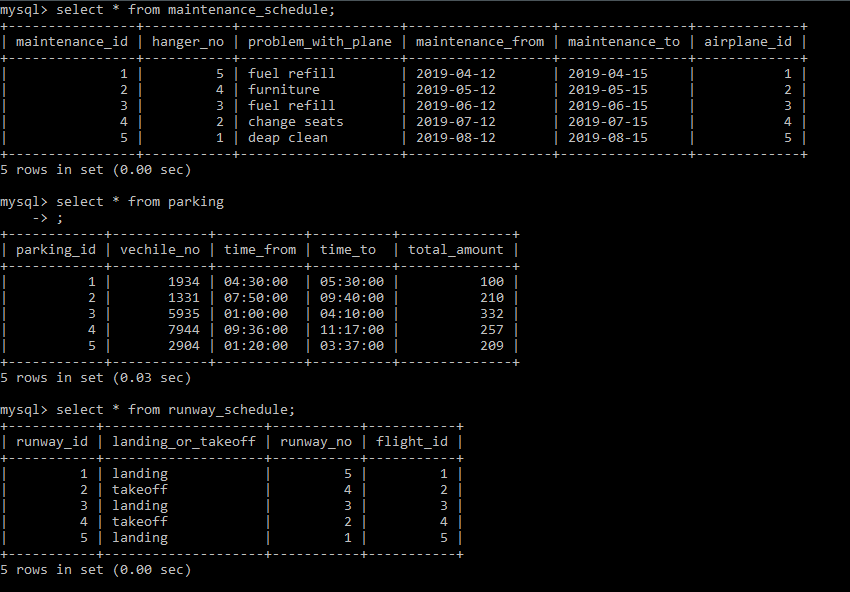
1. **NORMALISATION OF RELATIONAL SCHEMA**

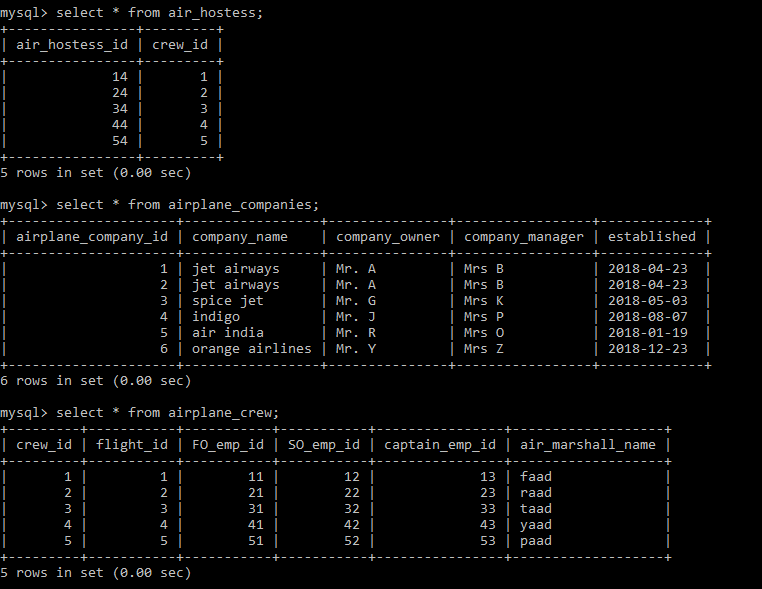
Since there are no repeating groups in the table, nor there is partial or transitive dependency, therefore the tables are in 3NF form.

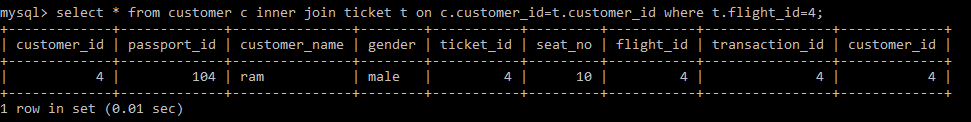
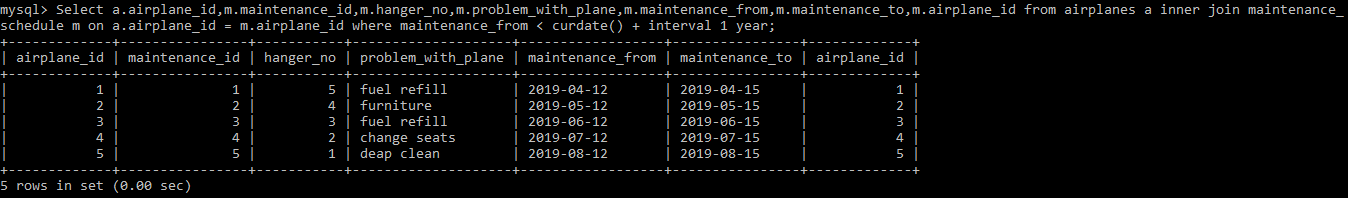
1. **DATABASE IMPLEMENTATION  
   Table descriptions:**

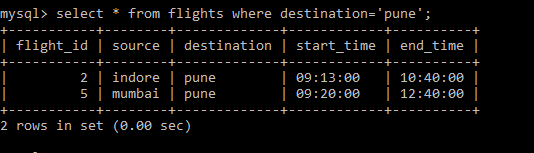
**Table Data:**

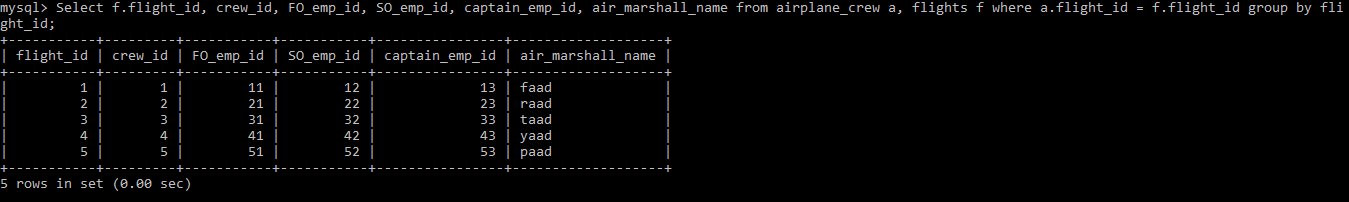
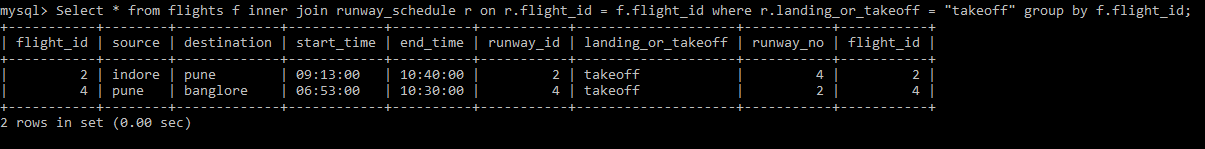
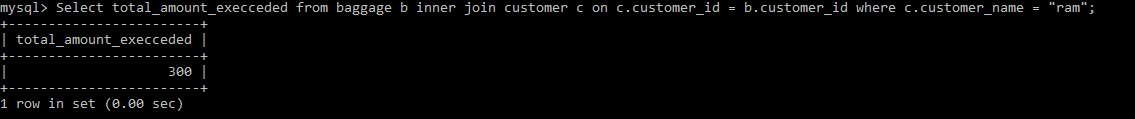
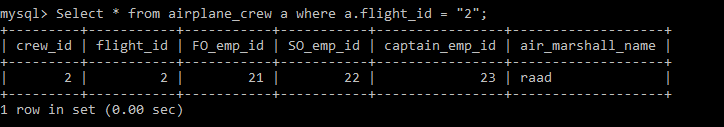
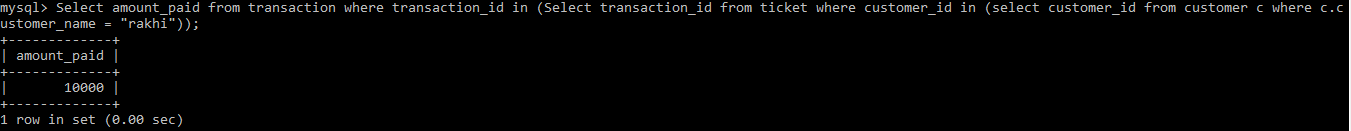
****

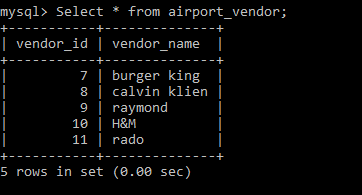
****

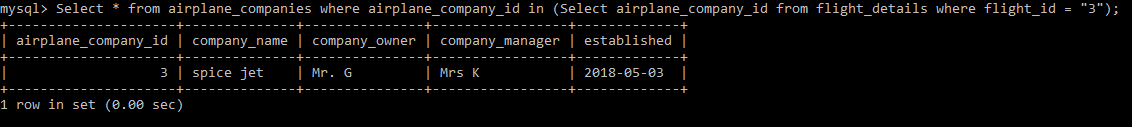
****

1. **QUERY EXECUTION**
   1. **Retrieve the set of customers who are travelling on flight number 4.**
   2. **Retrieve the details of airplanes that have their maintenance within the next 1 year.**
   3. **Retrieve the set of flights which are flying to Pune.**

****

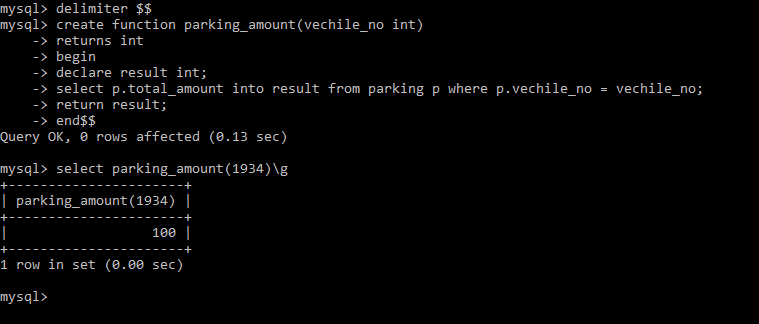
* 1. **Show flight wise crew details.**
  2. **Show the flight details for flights which are taking off.**
  3. **Show the extra charge the customer “Ram” had to pay on baggage.**
  4. **Retrieve crew details for the flight id 2.**
  5. **Retrieve the amount which Rakhi paid for his ticket to Delhi**
  6. **Show the vendors that have been allowed on the Airport.**

****

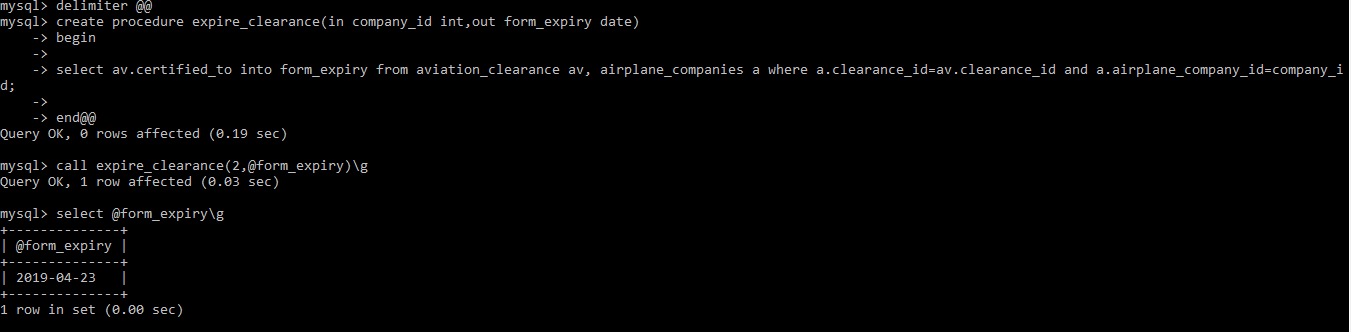
* 1. ** Get the details of the airline catering to flight number 3.**

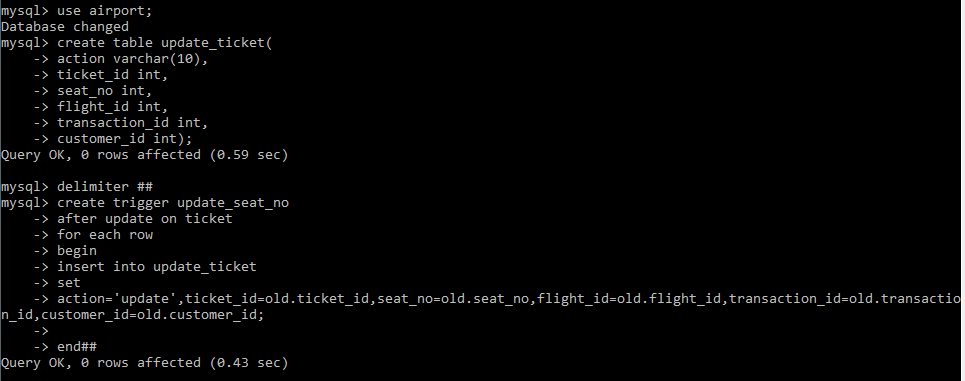
1. **FUNCTIONS**

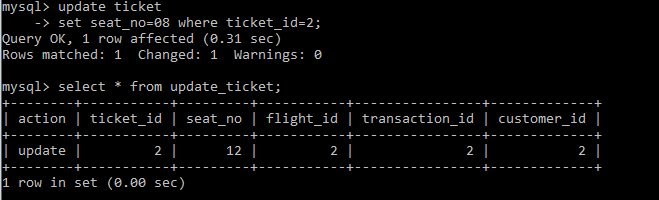
Function to retrieve the parking amount paid for a given vehicle number.



1. **PROCEDURES**

Procedure to retrieve the expiry date of the clearance form for a given company.

1. ** TRIGGERS**

****Trigger to update the ticket when the seat number is changed for particular ticket.