

A  
Project Report  
on  
**CAR SERVICE Management System**

Developed by

**Ravi Piparva - Department of IT, DD University**  
**Jay Solanki - Department of IT, DD University**

**Guided By**

**Internal Guide: Prof. Sunil K. Vithlani**

**Department of Information Technology**

**Faculty of Technology, DD University**



**Department of Information Technology**  
**Faculty of Technology, Dharmsinh Desai University**  
**College Road, Nadiad-387001**  
**October – 2019**

**DHARMSINH DESAI UNIVERSITY**  
**NADIAD-387001, GUJARAT**



**CERTIFICATE**

This is to certify that the project entitled “CAR SERVICE MANAGEMENT SYSTEM” is a bonafied report of the work carried out by

1) **Mr. RAVI PIPARVA,**

Student ID No: **17ITUOS090**

2) **Mr. JAY SOLANKI,**

Student ID No: **17ITUN018**

of Department of Information Technology, semester V, under the guidance and supervision for the subject Database Management System. They were involved in Project training during academic year 2019-2020.

**Prof. Sunil K. Vithlani**

(Project Guide)

Department of Information Technology,

Faculty of Technology,

Dharmsinh Desai University, Nadiad

Date:

**Prof. Vipul Dabhi**

Head, Department of Information Technology,

Faculty of Technology,

Dharmsinh Desai University, Nadiad

Date:

## Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project. All that we have have done is only due to such supervision and assistance and we would not forget to thank them.

We respect and thank **Prof. Sunil K. Vithlani** for providing us an opportunity to do the project work in DBMS and giving us all support and guidance, which made us complete the project duly. We are extremely thankful to her for providing such a nice support, guidance by taking keen interest in our project, although she had busy schedule managing the lectures.

We would also like to express our special thanks of gratitude to our HOD Prof. Vipul Dabhi who gave us the golden opportunity to do this wonderful project on the **Topic: Car Service Management**. We would also like to thank him for including such things in curriculum making it more interesting and useful practically.

Finally, we would like to thank each and every person who was there around us helping more or less in our project and keeping us motivated to work hard and complete the project. We would also thank them for inspiring us and sharing their ideas and views to make this project a success.

Thanking you.

Yours Sincerely,

**RAVI PIPARVA** (IT - 096)

**JAY SOLANKI** (IT - 111)

## TABLE OF CONTENTS

<b>I. Certificate</b>	<b>II</b>
<b>II. Acknowledgement</b>	<b>III</b>
<b>1. SYSTEM OVERVIEW</b>	<b>2</b>
1.1 Current system	5
1.2 Objectives of the Proposed System	5
1.3 Advantages of the Proposed system (over current)	5
<b>2. E-R DIAGRAM</b>	<b>7</b>
2.1 Entities	8
2.2 Relationships Mapping Constraints	9
<b>3. DATA DICTIONARY</b>	<b>10</b>
<b>4. SCHEMA DIAGRAM</b>	<b>13</b>
<b>5. DATABASE IMPLEMENTATION</b>	<b>10</b>
5.1 Create Schema	10
5.2 Insert Data values	16
5.3 Queries (Based on functions, group by, having, joins, sub query etc.)	23
5.4 PL/SQL Blocks (Procedures and Functions)	34
5.5 Exception	41
5.6 Triggers	42
5.7 Cursors	44
<b>6. FUTURE ENHANCEMENTS OF THE SYSTEM</b>	<b>47</b>
<b>7. BIBLIOGRAPHY</b>	<b>48</b>

## **1. System Overview**

### **1.1 Current System**

- This is Car Service Management System.
- It also allows the customer and user to Register Themselves.
- There are Multiple Car Service System from which Customer can select.
- The Customer can also select the type and time of Service they want.
- The Customer can also select the Particular and particular city from anywhere they want to service they want.
- The Customer can also avail different Offers Available E.g. At the time of festivals or some our center anniversary.
- The Customer can also give their valuable feedback and Suggestion to that Our System can improved as per the customer's valuable feedback.
- For all these facilities, i.e. registering, placing service at our center, giving a feedback, the customer gets an acknowledgement.

### **1.2 Objectives of the Proposed System**

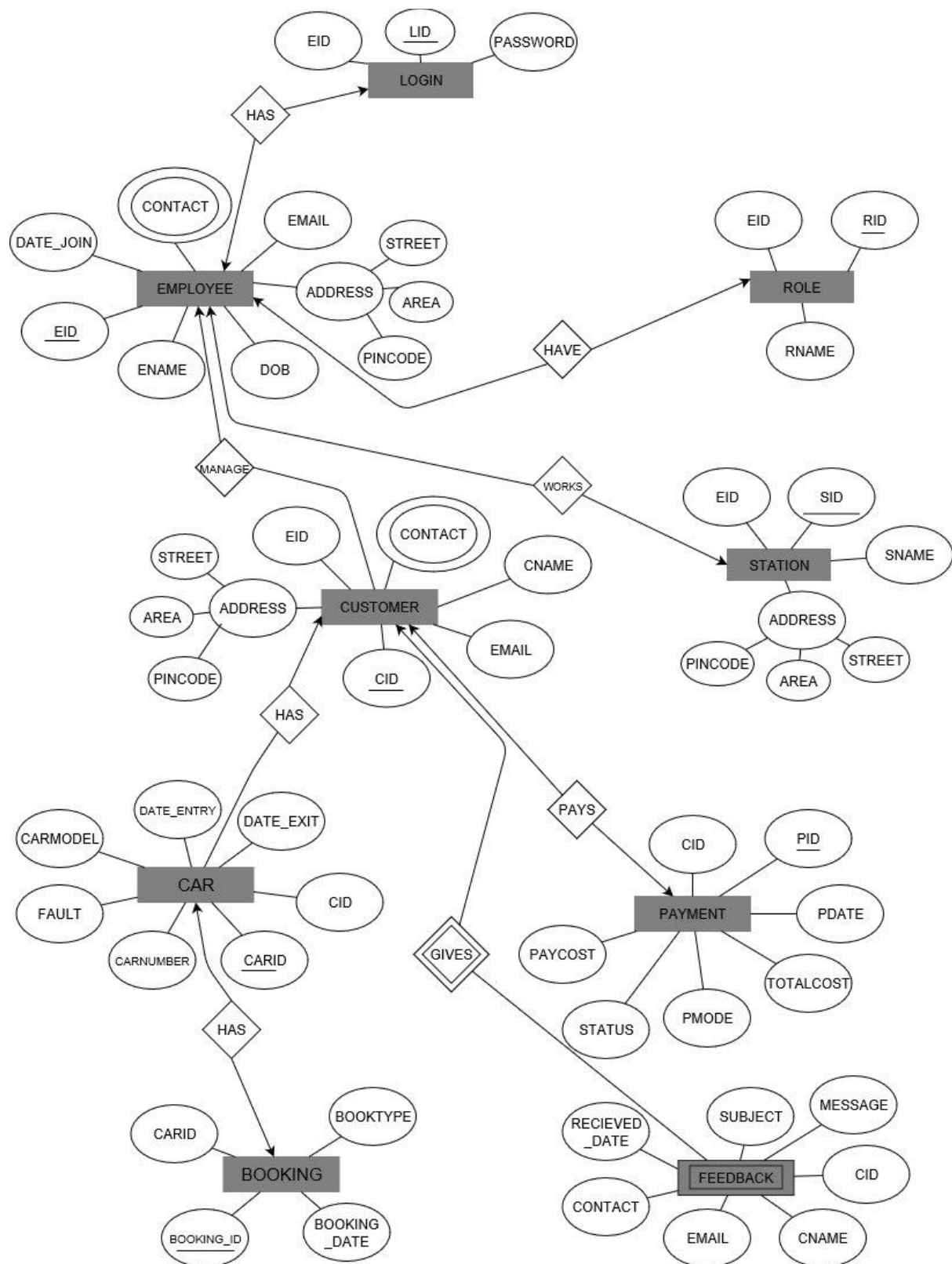
- Good and Efficient Service can provide to our customer in turn increasing the number of Customer's.
- The best feature of this system is to that the if the customer is not available in particular then also, he can service his car irrespective of place and time.
- The system suits for the people who cannot go and afford to waste time as the system is providing the doorstep services.
- Also, we in future we would try to give a quick notification for those customers who need the car service in particular period of time.

### **1.3 Advantages of the proposed system (Over Current)**

- Fast and efficient as well as affordable service
- The Customer will be notified when the Request for the service is placed, Done, delivered.
- The customer can pay through credit card online or by cash when the order is delivered.

- The customer can pay through credit card online or by cash when the order is delivered.
- There is no minimum amount of service required for doorstep delivery.
- This system covers all our Service center so that our customer can get access of Mobility of All our service on go.

## 2. E-R Diagram



## 2.1 Entities

- Login
- Employee
- Customer
- Role
- Station
- Payment
- Booking
- Car
- Feedback



## 2.2 Relationship and Mapping constraints

- Login has One to One relationship with Employee Table.  
Relation name: Has
- Employee has One to One relationship with Role Table
- Relation name: Have
- Employee has One to One relationship with Station Table  
Relation name: Works
- Employee has One to Many relationships with Customer  
Relation name: Manage
- Customer has One to One relationship with Payment  
Relation name: Pays
- Customer has One to Many relationships with Feedback  
Relation name: Gives
- Customer has One to Many relationships with Car  
Relation name: Has
- Car has One to One relationship with Booking  
Relation name: Pay Amount

### 3. Data Dictionary

#### 1. Login Table

Column	Type	Null	Default	Links to
LID ( <i>Primary</i> )	varchar(10)	No		
EID	varchar(10)	No		employee -> EID
PASSWORD	varchar(10)	No		

#### 2. Customer

Column	Type	Null	Default	Links to
CID ( <i>Primary</i> )	varchar(10)	No		
EID	varchar(10)	No		employee -> EID
CNAME	varchar(10)	No		
EMAIL	varchar(10)	No		
CONTACT	int(15)	No		
STREET	varchar(15)	No		
AREA	varchar(10)	No		
PINCODE	int(6)	No		

#### 3. Employee

Column	Type	Null	Default	Links to
EID ( <i>Primary</i> )	varchar(10)	No		
ENAME	varchar(10)	No		
EMAIL	varchar(10)	No		
CONTACT	int(10)	No		
DOB	date	No		
DATE_JOIN	date	No		
STREET	varchar(10)	No		
AREA	varchar(10)	No		
PINCODE	varchar(10)	No		

#### 4. Booking

Column	Type	Null	Default	Links to
BOOKING_ID ( <i>Primary</i> )	varchar(10)	No		
CARID	varchar(10)	No		car -> CARID
BOOKING_DATE	date	No		
BOOKTYPE	varchar(10)	No		

#### 5. Role

Column	Type	Null	Default	Links to
RID ( <i>Primary</i> )	varchar(10)	No		
EID	varchar(10)	No		employee -> EID
NAME	varchar(10)	No		

## 6. Car

Column	Type	Null	Default	Links to
CARID ( <i>Primary</i> )	varchar(10)	No		
CID	varchar(10)	No		customer -> CID
CARNUMBER	varchar(10)	No		
CARMODEL	varchar(10)	No		
FAULT	varchar(10)	No		
DATE_ENTRY	date	No		
DATE_EXIT	date	No		

## 7. Payment

Column	Type	Null	Default	Links to
PID ( <i>Primary</i> )	varchar(10)	No		
CID	varchar(10)	No		customer -> CID
PDATE	date	No		
PMODE	varchar(10)	No		
TOTALCOST	int(15)	No		
PAYCOST	int(15)	No		
STATUS	int(10)	No		

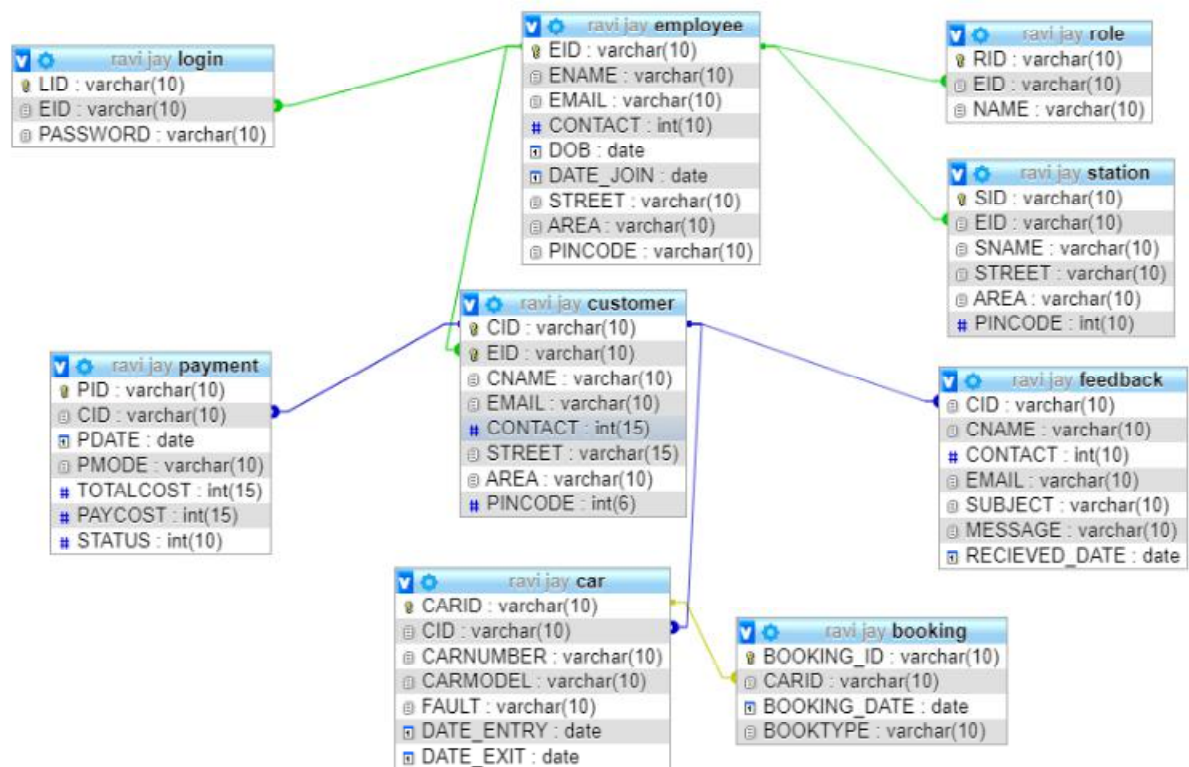
## 8. Station

Column	Type	Null	Default	Links to
SID ( <i>Primary</i> )	varchar(10)	No		
EID	varchar(10)	No		employee -> EID
SNAME	varchar(10)	No		
STREET	varchar(10)	No		
AREA	varchar(10)	No		
PINCODE	int(10)	No		

## 9. Feedback

Column	Type	Null	Default	Links to
CID	varchar(10)	No		customer -> CID
CNAME	varchar(10)	No		
CONTACT	int(10)	No		
EMAIL	varchar(10)	No		
SUBJECT	varchar(10)	No		
MESSAGE	varchar(10)	No		
RECIEVED_DATE	date	No		

## 4. Schema Diagram



## 5. Database Implementation

### 5.1 Create Schema

#### 1. Employee Table

```
CREATE TABLE EMPLOYEE (
    EID VARCHAR2(10) PRIMARY KEY,
    ENAME VARCHAR2(20),
    EMAIL VARCHAR2(20),
    CONTACT NUMBER (10),
    DOB DATE,
    DATE_JOIN DATE,
    STREET VARCHAR2(30),
    AREA VARCHAR2(30),
    PINCODE NUMBER(6)
);
```

DESC EMPLOYEE;

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	EID	Varchar2	10	-	-	1	-	-	-
	ENAME	Varchar2	20	-	-	-	✓	-	-
	EMAIL	Varchar2	20	-	-	-	✓	-	-
	CONTACT	Number	-	10	0	-	✓	-	-
	DOB	Date	7	-	-	-	✓	-	-
	DATE_JOIN	Date	7	-	-	-	✓	-	-
	STREET	Varchar2	30	-	-	-	✓	-	-
	AREA	Varchar2	30	-	-	-	✓	-	-
	PINCODE	Number	-	6	0	-	✓	-	-
1 - 9									

## 2. Login Table

```
CREATE TABLE LOGIN(
    LID VARCHAR2(10) PRIMARY KEY,
    EID VARCHAR2(10) REFERENCES EMPLOYEE(EID),
    PASSWORD VARCHAR2(25)
);
DESC LOGIN;
```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
LOGIN	LID	Varchar2	10	-	-	1	-	-	-
	EID	Varchar2	10	-	-	-	✓	-	-
	PASSWORD	Varchar2	25	-	-	-	✓	-	-
1 - 3									

## 3. Role Table

```
CREATE TABLE ROLE(
    RID VARCHAR2(10) PRIMARY KEY,
    EID VARCHAR2(10) REFERENCES EMPLOYEE(EID),
    RNAME VARCHAR2(25)
);
DESC ROLE;
```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ROLE	RID	Varchar2	10	-	-	1	-	-	-
	EID	Varchar2	10	-	-	-	✓	-	-
	RNAME	Varchar2	25	-	-	-	✓	-	-
1 - 3									

## 4. Customer

```
CREATE TABLE CUSTOMER(
    CID VARCHAR2(10) PRIMARY KEY,
    EID VARCHAR2(10) REFERENCES EMPLOYEE(EID),
    CNAME VARCHAR2(20),
```

EMAIL VARCHAR2(20),  
 CONTACT NUMBER(10),  
 STREET VARCHAR2(30),  
 AREA VARCHAR2(30),  
 PINCODE NUMBER(6));

DESC CUSTOMER;

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMER	CID	Varchar2	10	-	-	1	-	-	-
	EID	Varchar2	10	-	-	-	✓	-	-
	CNAME	Varchar2	20	-	-	-	✓	-	-
	EMAIL	Varchar2	20	-	-	-	✓	-	-
	CONTACT	Number	-	10	0	-	✓	-	-
	STREET	Varchar2	30	-	-	-	✓	-	-
	AREA	Varchar2	30	-	-	-	✓	-	-
	PINCODE	Number	-	6	0	-	✓	-	-

## 5. Car Table

CREATE TABLE CAR(  
 CARID VARCHAR2(10) PRIMARY KEY,  
 CID VARCHAR2(10) REFERENCES CUSTOMER(CID),  
 CARNUMBER VARCHAR2(20),  
 CARMODEL VARCHAR2(20),  
 FAULT VARCHAR2(25),  
 DATE\_ENTRY DATE,  
 DATE\_EXIT DATE);

DESC CAR;



Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CAR	CARID	Varchar2	10	-	-	1	-	-	-
	CID	Varchar2	10	-	-	-	✓	-	-
	CARNUMBER	Varchar2	20	-	-	-	✓	-	-
	CARMODEL	Varchar2	20	-	-	-	✓	-	-
	FAULT	Varchar2	25	-	-	-	✓	-	-
	DATE_ENTRY	Date	7	-	-	-	✓	-	-
	DATE_EXIT	Date	7	-	-	-	✓	-	-

## 6. Payment Table

CREATE TABLE PAYMENT(

PID VARCHAR2(10) PRIMARY KEY,

CID VARCHAR2(10) REFERENCES CUSTOMER(CID),

PDATE DATE,

PMODE DATE,

TOTALCOST NUMBER(10),

PAYCOST NUMBER(10),

STATUS NUMBER(5)

);

DESC PAYMENT;

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PAYMENT	PID	Varchar2	10	-	-	1	-	-	-
	CID	Varchar2	10	-	-	-	✓	-	-
	PDATE	Date	7	-	-	-	✓	-	-
	PMODE	Date	7	-	-	-	✓	-	-
	TOTALCOST	Number	-	10	0	-	✓	-	-
	PAYCOST	Number	-	10	0	-	✓	-	-
	STATUS	Number	-	5	0	-	✓	-	-

## 7. Booking Table

CREATE TABLE BOOKING(

BOOKING\_ID VARCHAR2(10) PRIMARY KEY,

CARID VARCHAR2(10) REFERENCES CAR(CARID),

BOOKING\_DATE DATE,

```

BOOKING_TIME TIMESTAMP,
BOOKTYPE VARCHAR2(50)
);

```

DESC BOOKING;

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BOOKING	BOOKING_ID	Varchar2	10	-	-	1	-	-	-
	CARID	Varchar2	10	-	-	-	✓	-	-
	BOOKING_DATE	Date	7	-	-	-	✓	-	-
	BOOKING_TIME	Timestamp(6)	11	-	6	-	✓	-	-
	BOOKTYPE	Varchar2	50	-	-	-	✓	-	-

## 8. Station Table

```

CREATE TABLE STATION(
    SID VARCHAR2(20) PRIMARY KEY,
    EID VARCHAR2(10) REFERENCES EMPLOYEE(EID),
    SNAME VARCHAR2(20),
    STREET VARCHAR2(30),
    AREA VARCHAR2(30),
    PINCODE NUMBER(6)
);

```

DESC STATION;

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STATION	SID	Varchar2	20	-	-	1	-	-	-
	EID	Varchar2	10	-	-	-	✓	-	-
	SNAME	Varchar2	20	-	-	-	✓	-	-
	STREET	Varchar2	30	-	-	-	✓	-	-
	AREA	Varchar2	30	-	-	-	✓	-	-
	PINCODE	Number	-	6	0	-	✓	-	-

## 9. Feedback Table

```

CREATE TABLE FEEDBACK(
    CID VARCHAR2(10) REFERENCES CUSTOMER(CID),

```

```

CNAME VARCHAR2(20),
CONTACT NUMBER(10),
EMAIL VARCHAR2(30),
SUBJECT VARCHAR2(30),
MESSAGE VARCHAR2(150),
RECEIVED_DATE TIMESTAMP
);

DESC FEEDBACK;

```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
FEEDBACK	CID	Varchar2	10	-	-	-	✓	-	-
	CNAME	Varchar2	20	-	-	-	✓	-	-
	CONTACT	Number	-	10	0	-	✓	-	-
	EMAIL	Varchar2	30	-	-	-	✓	-	-
	SUBJECT	Varchar2	30	-	-	-	✓	-	-
	MESSAGE	Varchar2	150	-	-	-	✓	-	-
	RECEIVED_DATE	Timestamp(6)	11	-	6	-	✓	-	-

## 5.2 Insert Data Value

### 1. Employee Table

```

INSERT ALL
INTO
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC
ODE)
VALUES('E101','CHETAN SOLANKI','CHETANSOLANKI@GMAIL',9428888426,'26-
JAN-97','26-FEB-17','PARAS SOCIITY','SURAT',360004)

INTO
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC
ODE)
VALUES('E102','MILAN PATEL','MILANPT@GMAIL',9542888842,'27-FEB-95','26-FEB-
17','SHIVAM PARK','NADIAD',387001)

INTO
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC
ODE)

```

```
VALUES('E103','RAM MEHTA','RAMBHAI@YAHOO',9727716624,'21-JAN-96','15-FEB-19','SHRUSHTI SOCIITY','NADIAD',387001)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E104','DEVANG SHARMA','DEVANGSH@OUTLO',9524598789,'15-AUG-98','07-FEB-17','KRISHNA SOCIITY','VAPI',365544)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E105','SAHIL ROKAD','ROKADSA@GMAIL',9988989897,'07-SEP-85','20-FEB-17','GAYTRI SOCIITY','SURAT',361162)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E106','KUNJ PATEL','PATELKUNJ@YAHOO',9978456312,'12-JUL-95','26-FEB-18','BALAJI PARK','AMRELI',364578)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E107','JAY SOLANKI','JAYSOLANKI@GMAIL',9612457836,'09-MAY-99','20-JUN-15','OM NAGAR','RAJKOT',360005)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E108','KRUPESH JAGANI','KJ@OUTLO',9925631487,'08-JAN-89','07-JUL-18','150 FEET ROAD','RAJKOT',360001)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E109','SACHIT PATEL','SACHIPT@GMAIL',9725631498,'19-FEB-90','01-MAR-19','AMIN MARG','BARODA',360002)
```

```
INTO
```

```
EMPLOYEE(EID,ENAME,EMAIL,CONTACT,DOB,DATE_JOIN,STREET,AREA,PINC  
ODE)
```

```
VALUES('E110','PRANAV SHARMA','PRANAVDA@YAHOO',9921657898,'31-JAN-96','01-FEB-16','OM NAGAR','RAJKOT',360005)
SELECT * FROM DUAL;
```

```
SELECT * FROM EMPLOYEE;
```

EID	ENAME	EMAIL	CONTACT	DOB	DATE_JOIN	STREET	AREA	PINCODE
E101	CHETAN SOLANKI	CHETANSOLANKI@GMAIL	9428888428	26-JAN-97	26-FEB-17	PARAS SOCIITY	SURAT	380004
E102	MILAN PATEL	MILANPT@GMAIL	9542888842	27-FEB-95	26-FEB-17	SHIVAM PARK	NADIAD	387001
E103	RAM MEHTA	RAMBHAI@YAHOO	9727718624	21-JAN-98	18-FEB-19	SHRUSHTI SOCIITY	NADIAD	387001
E104	DEVANG SHARMA	DEVANGSH@OUTLO	9524598789	15-AUG-98	07-FEB-17	KRISHNA SOCIITY	VAPI	365544
E105	SAHIL ROKAD	ROKADSA@GMAIL	9988998897	07-SEP-85	20-FEB-17	GAYTRI SOCIITY	SURAT	381182
E106	KUNJ PATEL	PATELKUNJ@YAHOO	9978458312	12-JUL-95	28-FEB-18	BALAJI PARK	AMRELI	364578
E107	JAY SOLANKI	JAYSOLANKI@GMAIL	9812457838	09-MAY-99	20-JUN-15	OM NAGAR	RAJKOT	380005
E108	KRUPESH JAGANI	KJ@OUTLO	9925831487	08-JAN-89	07-JUL-18	150 FEET ROAD	RAJKOT	380001
E109	SACHIT PATEL	SACHTPT@GMAIL	9725831498	19-FEB-90	01-MAR-19	AMIN MARG	BARODA	380002
E110	PRANAV SHARMA	PRANAVDA@YAHOO	9921657898	31-JAN-96	01-FEB-16	OM NAGAR	RAJKOT	380005

## 2. Login Table

```
INSERT ALL
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L101','E101','OPENTHEACCOUNT')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L102','E102','PASSWORD')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L103','E103','HEHEHHE')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L104','E104','NAIKEVO')
```

```
INTO LOGIN(LID,EID,PASSWORD)
```

```
VALUES('L105','E105','KAMSU6E')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L106','E106','HACKED')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L107','E107','OPEN123456')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L108','E108','OKKAV')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L109','E109','SPIDERMAN')
```

```
INTO LOGIN(LID,EID,PASSWORD)
VALUES('L110','E110','KABIRR')
SELECT * FROM DUAL;
```

LID	EID	PASSWORD
L101	E101	OPENTHEACCOUNT
L102	E102	PASSWORD
L103	E103	HEHEHHE
L104	E104	NAIKEVO
L105	E105	KAMSU8E
L106	E106	HACKED
L107	E107	OPEN123456
L108	E108	OKKAV
L109	E109	SPIDERMAN
L110	E110	KABIRR

```
SELECT *FROM LOGIN;
```

### 3. ROLE TABLE

```
INSERT ALL
INTO ROLE(RID,EID,RNAME)
VALUES('R101','E101','WORKER')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R102','E103','MANAGER')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R103','E102','ACCOUNTANT')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R104','E110','DELIEVERY BOY')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R105','E109','WORKER')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R106','E107','MANAGER')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R107','E108','WORKER')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R108','E106','ACCOUNTANT')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R109','E105','DELIEVERY BOY')
```

```
INTO ROLE(RID,EID,RNAME)
VALUES('R110','E104','WORKER')
SELECT *FROM DUAL;
```

```
SELECT *FROM ROLE;
```

RID	EID	RNAME
R101	E101	WORKER
R102	E103	MANAGER
R103	E102	ACCOUNTANT
R104	E110	DELIEVERY BOY
R105	E109	WORKER
R106	E107	MANAGER
R107	E108	WORKER
R108	E106	ACCOUNTANT
R109	E105	DELIEVERY BOY
R110	E104	WORKER

#### 4. Customer Table

```
INSERT ALL
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C101','E109','HARSHAD
PATEL','HARSHAD@GMAIL','9624789456','TULSIDHAM SOCIITY','UNA',365489)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C102','E105','YUGI PATEL','YUGANT@GMAIL',9856234174,'SHIVAM
SOCIITY','VAPI',361162)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C103','E103','SOHAM
KHATRANI','SKPATEL@GMAIL',9952654525,'VALLABH SOCIITY','SURAT',387001)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C104','E101','DARSHAN
BORAS','BORSADAR@OUTLO',9814362574,'SHIVAM PARK','RAJKOT',365612)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C105','E109','RUTVIK
VARSANI','RUTVIKPVT@YAHOO',9612324565,'TULSIDHAM
SOCIITY','UNA',365489)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C106','E105','DARPAN PATEL','DAPU@GMAIL',9988776655,'BTOWN
HALL','BARODA',361162)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C107','E101','MOHIT SHARMA','MOHITT@GMAIL',9855663322,'SHIVAM
PARK','RAJKOT',360005)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
VALUES('C108','E103','GAURAV SHAH','SHAHJI@YAHOO',9913553893,'SHRIJI
PARK','UNA',361256)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
```



```
VALUES('C109','E110','HARSH GOHEL','HARSHGL@OUTLO',9727716624,'BALAJI HALL','RAJKOT',351235)
```

```
INTO CUSTOMER(CID,EID,CNAME,EMAIL,CONTACT,STREET,AREA,PINCODE)
```

```
VALUES('C110','E102','DARSHAN POKER','POKER@GMAIL',9825025101,'150 FEET RING ROAD','ABAD',350080)
```

```
SELECT * FROM DUAL;
```

```
SELECT *FROM CUSTOMER;
```

CID	EID	CNAME	EMAIL	CONTACT	STREET	AREA	PINCODE
C101	E109	HARSHAD PATEL	HARSHAD@GMAIL	9824789456	TULSIDHAM SOCIITY	UNA	365489
C102	E105	YUGI PATEL	YUGANT@GMAIL	9856234174	SHIVAM SOCIITY	VAPI	361162
C103	E103	SOHAM KHATrani	SKPATEL@GMAIL	9952654525	VALLABH SOCIITY	SURAT	387001
C104	E101	DARSHAN BORAS	BORSADAR@OUTLO	9814362574	SHIVAM PARK	RAJKOT	365612
C105	E109	RUTVIK VARSANI	RUTVIK PVT@YAHOO	9812324565	TULSIDHAM SOCIITY	UNA	365489
C106	E105	DARPAN PATEL	DAPU@GMAIL	9988776655	BTOWN HALL	BARODA	361162
C107	E101	MOHIT SHARMA	MOHITT@GMAIL	9855663322	SHIVAM PARK	RAJKOT	360005
C108	E103	GAURAV SHAH	SHAHJI@YAHOO	9913553893	SHRIJI PARK	UNA	361256
C109	E110	HARSH GOHEL	HARSHGL@OUTLO	9727716624	BALAJI HALL	RAJKOT	351235
C110	E102	DARSHAN POKER	POKER@GMAIL	9825025101	150 FEET RING ROAD	ABAD	350080

## 5. Car Table

```
INSERT ALL
```

```
INTO
```

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR101','C105','GJ01AB7878','SUV','WHEEL ALINGMENT','12-JAN-19','13-JAN-19')
```

```
INTO
```

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR102','C105','GJ02KJ5555','MARUTI','BREAK FAULT','13-JAN-19','15-JAN-19')
```

```
INTO
```

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR103','C103','GJ03AB1334','HUNDAI','GEAR BOX','22-FEB-19','23-FEB-19')
```

```
INTO
```

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR104','C102','GJ04PK1222','FIAT','CLUTCH','02-JUL-19','05-JUL-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR105','C101','GJ17SD9999','SUV','EGR','17-MAY-19','17-MAY-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR106','C110','GJ01RT7878','SKODA','WHEEL ALINGMENT','19-AUG-19','20-AUG-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR107','C107','GJ11BC9999','SKODA','GEAR BOX','15-JAN-19','20-JAN-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR108','C107','GJ03AB5555','FORD','ROUTINE SERVICE','29-MAR-19','30-MAR-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR109','C108','GJ03KJ2222','NISSAN','ROUTINE SERVICE','09-SEP-19','09-SEP-19')
```

INTO

```
CAR(CARID,CID,CARNUMBER,CARMODEL,FAULT,DATE_ENTRY,DATE_EXIT)
VALUES('CR110','C109','GJ05AS1111','FORD','WHEEL ALINGMENT','30-NOV-19','30-NOV-19')
```

```
SELECT * FROM DUAL;
```

```
SELECT *FROM CAR;
```

CARID	CID	CARNUMBER	CARMODEL	FAULT	DATE_ENTRY	DATE_EXIT
CR101	C105	GJ01AB7878	SUV	WHEEL ALINGMENT	12-JAN-19	13-JAN-19
CR102	C105	GJ02KJ5555	MARUTI	BREAK FAULT	13-JAN-19	15-JAN-19
CR103	C103	GJ03AB1334	HUNDAI	GEAR BOX	22-FEB-19	23-FEB-19
CR104	C102	GJ04PK1222	FIAT	CLUTCH	02-JUL-19	05-JUL-19
CR105	C101	GJ17SD9999	SUV	EGR	17-MAY-19	17-MAY-19
CR106	C110	GJ01RT7878	SKODA	WHEEL ALINGMENT	19-AUG-19	20-AUG-19
CR107	C107	GJ11BC9999	SKODA	GEAR BOX	15-JAN-19	20-JAN-19
CR108	C107	GJ03AB5555	FORD	ROUTINE SERVICE	29-MAR-19	30-MAR-19
CR109	C108	GJ03KJ2222	NISSAN	ROUTINE SERVICE	09-SEP-19	09-SEP-19
CR110	C109	GJ05AS1111	FORD	WHEEL ALINGMENT	30-NOV-19	30-NOV-19

## 6. Payment

```
INSERT ALL
```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P101','C101','17-MAY-19','CASH',100000,50000,0)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P102','C105','13-JAN-19','CHAQUE',85000,85000,1)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P103','C105','15-JAN-19','DD',99999,75000,0)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P104','C103','23-FEB-19','ONLINE',69000,69000,1)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P105','C102','06-JUL-19','CASH',75000,5000,0)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P106','C110','20-AUG-19','CASH',55000,50000,0)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P107','C107','21-JAN-19','CHAQUE',79000,79000,1)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P108','C107','30-MAR-19','DD',5000,5000,1)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P109','C108','09-SEP-19','ONLINE',29000,10000,0)

```

```

INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST,STATUS)
VALUES('P110','C109','30-NOV-19','CASH',84999,40000,0)
SELECT * FROM DUAL;

```

```

SELECT *FROM PAYMENT;

```

PID	CID	PDATE	PMODE	TOTALCOST	PAYCOST	STATUS
P101	C101	17-MAY-19	CASH	100000	50000	0
P102	C105	13-JAN-19	CHAQUE	85000	85000	1
P103	C105	15-JAN-19	DD	99999	75000	0
P104	C103	23-FEB-19	ONLINE	69000	69000	1
P105	C102	06-JUL-19	CASH	75000	5000	0
P106	C110	20-AUG-19	CASH	55000	50000	0
P107	C107	21-JAN-19	CHAQUE	79000	79000	1
P108	C107	30-MAR-19	DD	5000	5000	1
P109	C108	09-SEP-19	ONLINE	29000	10000	0
P110	C109	30-NOV-19	CASH	84999	40000	0

## 7.Booking Table

```

INSERT ALL
INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B101','CR104','01-JUL-19','DROP AT OUR SERVICE CENTRE')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B102','CR109','01-SEP-19','PICKUP SERVICE BY OUR TEAM')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B103','CR108','18-MAR-19','PICKUP SERVICE BY OUR TEAM')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B104','CR107','10-JAN-19','DROP AT OUR SERVICE CENTRE')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B105','CR106','05-AUG-19','PICKUP SERVICE BY OUR TEAM')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B106','CR105','10-MAY-19','DROP AT OUR SERVICE CENTRE')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B107','CR103','19-JAN-19','DROP AT OUR SERVICE CENTRE')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B108','CR102','11-JAN-19','PICKUP SERVICE BY OUR TEAM')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B109','CR101','11-JAN-19','DROP AT OUR SERVICE CENTRE')

INTO BOOKING(BOOKING_ID,CARID,BOOKING_DATE,BOOKTYPE)
VALUES('B110','CR110','20-OCT-19','PICKUP SERVICE BY OUR TEAM')
SELECT * FROM DUAL;

```

```
SELECT *FROM BOOKING;
```

BOOKING_ID	CARID	BOOKING_DATE	BOOKING_TIME	BOOKTYPE
B101	CR104	01-JUL-19	-	DROP AT OUR SERVICE CENTRE
B102	CR109	01-SEP-19	-	PICKUP SERVICE BY OUR TEAM
B103	CR108	18-MAR-19	-	PICKUP SERVICE BY OUR TEAM
B104	CR107	10-JAN-19	-	DROP AT OUR SERVICE CENTRE
B105	CR106	05-AUG-19	-	PICKUP SERVICE BY OUR TEAM
B106	CR105	10-MAY-19	-	DROP AT OUR SERVICE CENTRE
B107	CR103	19-JAN-19	-	DROP AT OUR SERVICE CENTRE
B108	CR102	11-JAN-19	-	PICKUP SERVICE BY OUR TEAM
B109	CR101	11-JAN-19	-	DROP AT OUR SERVICE CENTRE
B110	CR110	20-OCT-19	-	PICKUP SERVICE BY OUR TEAM

## 8. Station Table

```
INSERT ALL
INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S101','E109','COMPLETE CAR','TULSIDHAM SOCIITY','UNA',365489)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S102','E108','PERFECT SERVICE','AMBIKA PARK','NADIAD',361162)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S103','E107','CAR DECOR STATION','GAYATRI NAGAR','NADIAD',387001)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S104','E101','4 WHEEL AUTO REPAIR','BALAJI PARK','RAJKOT',360005)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S105','E102','VROOM AUTO REPAIR','AMIN MARG','RAJKOT',360004)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S106','E103','QUICK SERVICE','MAHAPOOJA DHAM','SURAT',358978)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S107','E104','FAST SERVICES','KKV CHOWK','BARODA',387332)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S108','E105','THE CAR CLINIC','PETLAD ','ANAD',365489)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S109','E106','CLASSIC MOTOR REPAIR','OM NAGAR
SOCIITY','VVNAGAR',361198)

INTO STATION(SID,EID,SNAME,STREET,AREA,PINCODE)
VALUES('S110','E110','BEST SERVICE STATION','SHIVAM PARK','BARODA',354789)

SELECT *FROM DUAL;
SELECT *FROM STATION;
```

SID	EID	SNAME	STREET	AREA	PINCODE
S101	E109	COMPLETE CAR	TULSIDHAM SOCIITY	UNA	365489
S102	E108	PERFECT SERVICE	AMBIKA PARK	NADIAD	361162
S103	E107	CAR DECOR STATION	GAYATRI NAGAR	NADIAD	367001
S104	E101	4 WHEEL AUTO REPAIR	BALAJI PARK	RAJKOT	360005
S105	E102	VROOM AUTO REPAIR	AMIN MARG	RAJKOT	360004
S106	E103	QUICK SERVICE	MAHAPOOJA DHAM	SURAT	358978
S107	E104	FAST SERVICES	KKV CHOWK	BARODA	367332
S108	E105	THE CAR CLINIC	PETLAD	ANAD	365489
S109	E106	CLASSIC MOTOR REPAIR	OM NAGAR SOCIITY	VVNAGAR	361198
S110	E110	BEST SERVICE STATION	SHIVAM PARK	BARODA	354789

## 9. Feedback Table

INSERT ALL

INTO

```
FEEDBACK(CID,CNAME,CONTACT,EMAIL,SUBJECT,MESSAGE,RECEIVED_DATE
)
VALUES('C101','HARSHAD PATEL',9725456321,'HARSHAD@GMAIL','FOR
SERVICES','EXCELLENT','31-DEC-19')
```

INTO

```
FEEDBACK(CID,CNAME,CONTACT,EMAIL,SUBJECT,MESSAGE,RECEIVED_DATE
)
VALUES('C102','YUGI PATEL',9856234174,'YUGANT@GMAIL','ABOUT STATON','5
STAR','21-JAN-19')
```

INTO

```
FEEDBACK(CID,CNAME,CONTACT,EMAIL,SUBJECT,MESSAGE,RECEIVED_DATE
)
VALUES('C103','SOHAM KHATRANI',9952654525,'SKPATEL@GMAIL','FOR
SERVICES','GOOD','25-MAY-19')
```

INTO

```
FEEDBACK(CID,CNAME,CONTACT,EMAIL,SUBJECT,MESSAGE,RECEIVED_DATE
)
VALUES('C104','DARSHAN BORAS',9814362574,'BORSADAR@OUTLO','ABOUT
STATION','BEST','11-NOV-19')
```

INTO

```
FEEDBACK(CID,CNAME,CONTACT,EMAIL,SUBJECT,MESSAGE,RECEIVED_DATE
)
```



```
VALUES('C105','RUTVIK VARSANI',9612324565,'RUTVIKPVT@YAHOO','FOR
SERVICES','NOT BAD','05-AUG-19')
SELECT *FROM DUAL;
```

```
SELECT *FROM FEEDBACK;
```

CID	CNAME	CONTACT	EMAIL	SUBJECT	MESSAGE	RECEIVED_DATE
C101	HARSHAD PATEL	9725456321	HARSHAD@GMAIL	FOR SERVICES	EXCELLENT	31-DEC-19 12.00.00.000000 AM
C102	YUGI PATEL	9856234174	YUGANT@GMAIL	ABOUT STATON	5 STAR	21-JAN-19 12.00.00.000000 AM
C103	SOHAM KHATrani	9952654525	SKPATEL@GMAIL	FOR SERVICES	GOOD	25-MAY-19 12.00.00.000000 AM
C104	DARSHAN BORAS	9814362574	BORSADAR@OUTLO	ABOUT STATION	BEST	11-NOV-19 12.00.00.000000 AM
C105	RUTVIK VARSANI	9612324565	RUTVIKPVT@YAHOO	FOR SERVICES	NOT BAD	05-AUG-19 12.00.00.000000 AM

### 5.3 Queries(Based on functions group by, having ,joins ,sub query etc.)

1. Determine the minimum and the maximum cost of service from payment.

```
SQL : SELECT MAX(TOTALCOST) AS MAXIMUM_COST,MIN(TOTALCOST) AS
MINIMUM_COST FROM PAYMENT;
```

MAXIMUM_COST	MINIMUM_COST
100000	5000

2. Find The Address Of Customer Which Has Cost Is Minimum.

```
SQL : SELECT C.CID,C.STREET,C.AREA,C.PINCODE FROM CUSTOMER
C,PAYMENT P WHERE C.CID=P.CID AND P.TOTALCOST=(SELECT
MIN(P.TOTALCOST) FROM PAYMENT P);
```

CID	STREET	AREA	PINCODE
C107	SHIVAM PARK	RAJKOT	380005

### 3. Display Customer Info With Payment Id Which Has Given On Date Before 25 Of Any Month.

**SQL :** SELECT

P.PID,C.CID,C.EID,C.CNAME,C.EMAIL,C.CONTACT,C.AREA,C.STREET,C.PINCODE  
FROM CUSTOMER C,PAYMENT P WHERE P.CID=C.CID AND EXTRACT(DAY  
FROM P.PDATE)<25;

PID	CID	EID	CNAME	EMAIL	CONTACT	AREA	STREET	PINCODE
P101	C101	E109	HARSHAD PATEL	HARSHAD@GMAIL	9824789456	UNA	TULSIDHAM SOCIITY	365489
P105	C102	E105	YUGI PATEL	YUGANT@GMAIL	9856234174	VAPI	SHIVAM SOCIITY	361162
P104	C103	E103	SOHAM KHATrani	SKPATEL@GMAIL	9952654525	SURAT	VALLABH SOCIITY	387001
P103	C105	E109	RUTVIK VARSANI	RUTVIK PVT@YAHOO	9612324565	UNA	TULSIDHAM SOCIITY	365489
P102	C105	E109	RUTVIK VARSANI	RUTVIK PVT@YAHOO	9612324565	UNA	TULSIDHAM SOCIITY	365489
P107	C107	E101	MOHIT SHARMA	MOHITT@GMAIL	9856663322	RAJKOT	SHIVAM PARK	360005
P109	C108	E103	GAURAV SHAH	SHAHJI@YAHOO	9913553893	UNA	SHRIJI PARK	361256
P106	C110	E102	DARSHAN POKER	POKER@GMAIL	9825025101	ABAD	150 FEET RING ROAD	350080

### 4. Find The Car Details Who Is Associated With Employee'PRANAV SHARMA'.

**SQL :** SELECT C.CARID,C.CID,C.CARNUMBER,C.CARMODEL,C.FAULT FROM CAR C INNER  
JOIN CUSTOMER O ON C.CID=O.CID INNER JOIN EMPLOYEE E ON E.EID=O.EID WHERE  
E.ENAME='PRANAV SHARMA';

CARID	CID	CARNUMBER	CARMODEL	FAULT
CR110	C109	GJ05AS1111	FORD	WHEEL ALINGMENT

### 5. Find All The Employee Name and Customer Name Who Have Payment>7000.

**SQL :** SELECT C.CNAME,E.ENAME FROM CUSTOMER C INNER JOIN EMPLOYEE  
E ON E.EID=C.EID INNER JOIN PAYMENT P ON P.CID=C.CID WHERE  
P.TOTALCOST>70000;

CNAME	ENAME
MOHIT SHARMA	CHETAN SOLANKI
YUGI PATEL	SAHIL ROKAD
RUTVIK VARSANI	SACHIT PATEL
RUTVIK VARSANI	SACHIT PATEL
HARSHAD PATEL	SACHIT PATEL
HARSH GOHEL	PRANAV SHARMA



**6. Find The All Detail of Booking where Booking Date is Earlier.**

SQL : SELECT \*FROM BOOKING WHERE BOOKING\_DATE=(SELECT MIN (BOOKING\_DATE )FROM BOOKING);

BOOKING_ID	CARID	BOOKING_DATE	BOOKING_TIME	BOOKTYPE
B104	CR107	10-JAN-19	-	DROP AT OUR SERVICE CENTRE

**7. Display Employee name whose Role Is Worker.**

SQL: SELECT E.ENAME FROM EMPLOYEE E WHERE E.EID IN(SELECT R.EID FROM ROLE R WHERE RNAME IN('WORKER'));

ENAME
CHETAN SOLANKI
SACHIT PATEL
KRUPESH JAGANI
DEVANG SHARMA

**8. Display Car Id,Car Model ,Car Number Which Customer Id Is Highest.**

SQL : SELECT CARMODEL,CARID ,CARNUMBER FROM CAR WHERE CID=(SELECT MAX(CID) FROM CAR);

CARMODEL	CARID	CARNUMBER
SKODA	CR106	GJ01RT7878

**9. Display Payment Id, Total Cost , Pay Cost Status whose cost is greater than the average cost.**

SQL : SELECT P.PID,P.TOTALCOST,P.PAYCOST,P.STATUS FROM PAYMENT P WHERE(((P.TOTALCOST)>(SELECT AVG(TOTALCOST) FROM PAYMENT))) ORDER BY P.TOTALCOST DESC;

PID	TOTALCOST	PAYCOST	STATUS
P101	100000	50000	0
P103	99999	75000	0
P102	85000	85000	1
P110	84999	40000	0
P107	79000	79000	1
P105	75000	5000	0
P104	69000	69000	1

**10.Display name of all the customer's whose payment is done in one time.**

SQL : SELECT C.CNAME , COUNT(P.TOTALCOST) FROM (PAYMENT P INNER JOIN CUSTOMER C ON C.CID=P.CID)

GROUP BY CNAME HAVING COUNT(P.TOTALCOST)=1;

CNAME	COUNT(P.TOTALCOST)
HARSHAD PATEL	1
HARSH GOHEL	1
YUGI PATEL	1
SOHAM KHATrani	1
GAURAV SHAH	1
DARSHAN POKER	1

**11.Display name of all the customer's whose second letter is U.**

SQL : SELECT \* FROM CUSTOMER WHERE CNAME LIKE '\_U%';

CID	EID	CNAME	EMAIL	CONTACT	STREET	AREA	PINCODE
C102	E105	YUGI PATEL	YUGANT@GMAIL	9856234174	SHIVAM SOCIITY	VAPI	361162
C105	E109	RUTVIK VARSANI	RUTVIK PVT@YAHOO	9812324565	TULSIDHAM SOCIITY	UNA	365489

**5.4 PL/SQL BLOCKS****5.4.1 PROCEDURES**

**1.CREATE A PROCEDURE TO CALCULATE THE DIFFERENCE BETWEEN THE LAST SERVICE AND TODAY'S DATE IF THE TOTAL DIFFERENCE ARE GRATER THAN 200 THEN SERVICE IS NEEDED ELSE NOT NEEDED.**

CREATE OR REPLACE PROCEDURE SERVICETIME (CARR\_ID CAR.CARID%TYPE)  
IS  
DAYDIFF NUMBER;

BEGIN

SELECT(SYSDATE-DATE\_EXIT) INTO DAYDIFF FROM CAR WHERE  
CARID=CARR\_ID;

IF DAYDIFF>200 THEN

```

DBMS_OUTPUT.PUT_LINE('SERVICE IS NEEDED');
ELSE
DBMS_OUTPUT.PUT_LINE('YOUR CAR IS UPTO DATE');
END IF;
END;

```

```

DECLARE
A CAR.CARID%TYPE;
BEGIN
A:=:CARID;
SERVICETIME(A);
END;

```

:CARID

SERVICE IS NEEDED

statement processed.

## 2. CREATE A PROCEDURE THAT FINDS THE AVERAGE PAYCOST OF ALL CUSTOMER AND COMPARE PAYCOST OF ONE CUSTOMER WITH THE AVERAGE PAYCOST OF REST OF ALL THE CUSTOMER.

```

CREATE OR REPLACE PROCEDURE AVGPAY (
  p_PID    IN PAYMENT.PID%TYPE
)
IS
  v_CID    VARCHAR2(10);
  v_PMODE  VARCHAR2(9);
  v_PDATE  DATE;
  v_TOTALCOST    NUMBER(8,2);
  v_PAYCOST      NUMBER(8);
  v_AVGPAY      NUMBER(8,2);
BEGIN
  SELECT CID, PMODE, PDATE, TOTALCOST, PAYCOST
  INTO v_CID, v_PMODE, v_PDATE, v_TOTALCOST, v_PAYCOST
  FROM PAYMENT WHERE PID = p_PID;
  DBMS_OUTPUT.PUT_LINE('PAYMENT ID # : ' || p_PID);
  DBMS_OUTPUT.PUT_LINE('CUSTOMER ID    : ' || v_CID);
  DBMS_OUTPUT.PUT_LINE('PAY MODE      : ' || v_PMODE);
  DBMS_OUTPUT.PUT_LINE('PAY Date   : ' || v_PDATE);
  DBMS_OUTPUT.PUT_LINE('TOTALCOST   : ' || v_TOTALCOST);
  DBMS_OUTPUT.PUT_LINE('PAYCOST     : ' || v_PAYCOST);

```

```

SELECT AVG(PAYCOST) INTO v_AVGPAY
FROM PAYMENT;
IF v_TOTALCOST > v_AVGPAY THEN
    DBMS_OUTPUT.PUT_LINE('CUSTOMER"s PAYMENT is more than the OTHERS '
        || 'average of ' || v_AVGPAY);
ELSE
    DBMS_OUTPUT.PUT_LINE('Employee"s PAYMENT does not exceed the OTHERS '
        || 'average of ' || v_AVGPAY);
END IF;
END;

```

```

declare
NA PAYMENT.PID%TYPE;
BEGIN
NA:=:ENTERTHEPAYID;
AVGPAY(NA);
END;

```

:ENTERTHEPAYID

```

PAYMENT ID # : P102
CUSTOMER ID   : C105
PAY MODE      : CHAQUE
PAY Date     : 13-JAN-19
TOTALCOST     : 85000
PAYCOST       : 85000
CUSTOMER"s PAYMENT is more than the OTHERS average of 46800

statement processed.

```

### 3.CREATE A PROCEDURE TO REGISTER A NEW CUSTOMER.

```

CREATE OR REPLACE PROCEDURE NEWCUST
(
    CIID IN CUSTOMER.CID%TYPE,
    EIID IN CUSTOMER.EID%TYPE,
    CNNAME IN CUSTOMER.CNAME%TYPE,
    EEMAIL IN CUSTOMER.EMAIL%TYPE,
    CCON IN CUSTOMER.CONTACT%TYPE,
    STR IN CUSTOMER.STREET%TYPE,
    AR IN CUSTOMER.AREA%TYPE,
    PIN IN CUSTOMER.PINCODE%TYPE
)
IS
BEGIN

```

```
INSERT INTO  
CUSTOMER("CID","EID","CNAME","EMAIL","CONTACT","STREET","AREA","PINC  
ODE")  
VALUES(CIID,EIID,CNNAME,EEMAIL,CCON,STR,AR,PIN);  
END;
```

```
DECLARE  
A CUSTOMER.CID%TYPE;  
B CUSTOMER.EID%TYPE;  
C CUSTOMER.CNAME%TYPE;  
D CUSTOMER.EMAIL%TYPE;  
E CUSTOMER.CONTACT%TYPE;  
F CUSTOMER.STREET%TYPE;  
G CUSTOMER.AREA%TYPE;  
H CUSTOMER.PINCODE%TYPE;
```

```
BEGIN  
A:=:CID;  
B:=:EID;  
C:=:CNAME;  
D:=:EMAIL;  
E:=:CONTACT;  
F:=:STREET;  
G:=:AREA;  
H:=:PINCODE;
```

```
NEWCUST(A,B,C,D,E,F,G,H);  
END;
```

:CID	C119
:EID	E101
:CNAME	JAY SOLANKI
:EMAIL	JAY@G.COM
:CONTACT	9426465190
:STREET	TULSIDHAM
:AREA	UNA
:PINCODE	362560

CID	EID	CNAME	EMAIL	CONTACT	STREET	AREA	PINCODE
C101	E109	HARSHAD PATEL	HARSHAD@GMAIL	9824789458	TULSIDHAM SOCIETY	UNA	365489
C102	E105	YUGI PATEL	YUGANT@GMAIL	9856234174	SHIVAM SOCIETY	VAPI	361162
C103	E103	SOHAM KHATrani	SKPATEL@GMAIL	9952654525	VALLABH SOCIETY	SURAT	387001
C104	E101	DARSHAN BORAS	BORSADAR@OUTLO	9814362574	SHIVAM PARK	RAJKOT	365612
C105	E109	RUTVIK VARSANI	RUTVIKPVT@YAHOO	9812324565	TULSIDHAM SOCIETY	UNA	365489
C106	E105	DARPAN PATEL	DAPU@GMAIL	9988776655	BTOWN HALL	BARODA	361162
C107	E101	MOHIT SHARMA	MOHITT@GMAIL	9855663322	SHIVAM PARK	RAJKOT	360005
C108	E103	GAURAV SHAH	SHAHJI@YAHOO	9913553893	SHRIJI PARK	UNA	361256
C109	E110	HARSH GOHEL	HARSHGL@OUTLO	9727716624	BALAJI HALL	RAJKOT	351235
C110	E102	DARSHAN POKER	POKER@GMAIL	9825025101	150 FEET RING ROAD	ABAD	350080
C119	E101	JAY SOLANKI	JAY@G.COM	9426465190	TULSIDHAM	UNA	362560

11 rows returned in 0.00 seconds

[CSV Export](#)

#### 4.CREATE A PROCEDURE TO GIVE FEEDBACK BY CUSTOMER.

CREATE OR REPLACE PROCEDURE GIVE\_FEEDBACK

```
(
C IN FEEDBACK.CID%TYPE,
SR IN FEEDBACK.CNAME%TYPE,
SG IN FEEDBACK.CONTACT%TYPE,
SH IN FEEDBACK.EMAIL%TYPE,
ST IN FEEDBACK.SUBJECT%TYPE,
SI IN FEEDBACK.MESSAGE%TYPE
)
```

IS

BEGIN

INSERT INTO

FEEDBACK("CID","CNAME","CONTACT","EMAIL","SUBJECT","MESSAGE")

VALUES(C,SR,SG,SH,ST,SI);

END;

DECLARE

A FEEDBACK.CID%TYPE;

B FEEDBACK.CNAME%TYPE;

C FEEDBACK.CONTACT%TYPE;

D FEEDBACK.EMAIL%TYPE;

E FEEDBACK.SUBJECT%TYPE;

F FEEDBACK.MESSAGE%TYPE;

BEGIN

A:=CID;

B:=CNAME;

C:=CONTACT;

```

D:=:EMAIL;
E:=:SUBJECT;
F:=:MESSAGE;
GIVE_FEEDBACK(A,B,C,D,E,F);
END;

```

:CID	C101
:CNAME	JAY
:CONTACT	9426465190
:EMAIL	JAY@GMAIL.COM
:SUBJECT	REPAIR
:MESSAGE	XYZ

CID	CNAME	CONTACT	EMAIL	SUBJECT	MESSAGE	RECEIVED_DATE
C101	HARSHAD PATEL	9725455321	HARSHAD@GMAIL	FOR SERVICES	EXCELLENT	31-DEC-19 12.00.00.000000 AM
C102	YUGI PATEL	9858234174	YUGANT@GMAIL	ABOUT STATON	5 STAR	21-JAN-19 12.00.00.000000 AM
C103	SOHAM KHATrani	9952654525	SKPATEL@GMAIL	FOR SERVICES	GOOD	25-MAY-19 12.00.00.000000 AM
C104	DARSHAN BORAS	9814362574	BORSADAR@OUTLO	ABOUT STATION	BEST	11-NOV-19 12.00.00.000000 AM
C105	RUTVIK VARSANI	9812324565	RUTVIKPV@YAHOO	FOR SERVICES	NOT BAD	05-AUG-19 12.00.00.000000 AM
C101	JAY	9426465190	JAY@GMAIL.COM	REPAIR	XYZ	-

6 rows returned in 0.02 seconds

[CSV Export](#)

### 5.4.2 FUNCTIONS

**1.CREATE A FUNCTION WHICH RETURN TOTAL OF TOTAL AMOUNT OF ALL THOSE CUSTOMER WHO HAS GIVEN THE PAYMENT WITH THE CASH MODE.**

CREATE OR REPLACE FUNCTION TOT\_TOTALPAYMENT  
RETURN NUMBER AS TOTAL NUMBER:=0;

```

BEGIN
SELECT SUM(TOTALCOST) INTO TOTAL FROM PAYMENT WHERE
Pmode='CASH';
RETURN TOTAL;
END;

```

SELECT TOT\_TOTALPAYMENT FROM DUAL;

TOT_TOTALPAYMENT
314999

1 rows returned in 0.00 seconds

[CSV Export](#)

## 2.CREATE A FUNCTION TO FIND OUT THE TOTALCOST OF CUSTOMER BY PAYMENTID.

```
CREATE OR REPLACE FUNCTION BILL_INFO(P IN PAYMENT.PID%TYPE)
RETURN NUMBER IS
NAMOUNT PAYMENT.TOTALCOST%TYPE;
PAYMENT_ID PAYMENT.PID%TYPE;
BEGIN
PAYMENT_ID:=P;
SELECT TOTALCOST INTO NAMOUNT FROM PAYMENT P WHERE
P.PID=PAYMENT_ID;
DBMS_OUTPUT.PUT_LINE('NET PAYABLE AMOUNT IS: ');
RETURN NAMOUNT;
END;
```

```
DECLARE
P PAYMENT.PID%TYPE;
A PAYMENT.TOTALCOST%TYPE;
BEGIN
P:=:PAYMENT_ID;
A:=BILL_INFO(P);
DBMS_OUTPUT.PUT_LINE(A);
END;
```

:PAYMENT\_ID

NET PAYABLE AMOUNT IS:  
99999

Statement processed.

## 3.CREATE A FUNCTION TO OUT MODE OF PAYMENT AND PAYMENT DETAILS BY USING CUSTOMER ID.

```
CREATE OR REPLACE FUNCTION PAYMENT_INFO(C IN CUSTOMER.CID%TYPE)
RETURN NUMBER IS
PAYMENTID PAYMENT.PID%TYPE;
TAMOUNT PAYMENT.TOTALCOST%TYPE;
PAMODE PAYMENT.PMODE%TYPE;
BEGIN
SELECT PID INTO PAYMENTID FROM PAYMENT WHERE CID=C;
SELECT TOTALCOST INTO TAMOUNT FROM PAYMENT P WHERE
P.PID=PAYMENTID;
```



```

SELECT PMODE INTO PAMODE FROM PAYMENT P WHERE P.PID=PAYMENTID;
DBMS_OUTPUT.PUT_LINE('MODE OF PAYMENT: '||PAMODE);
RETURN TAMOUNT;
END;

```

```

DECLARE
C CUSTOMER.CID%TYPE;
A PAYMENT.TOTALCOST%TYPE;
BEGIN
C:=CUSTOMER_ID;
A:=PAYMENT_INFO(C);
DBMS_OUTPUT.PUT_LINE('TOTAL AMOUNT TO PAY: '||A);
END;

```

:CUSTOMER\_ID

```

MODE OF PAYMENT: CASH
TOTAL AMOUNT TO PAY: 100000

Statement processed.

0.03 seconds

```

## 5.5 EXCEPTION

### INBUILT EXCEPTION

**1.CREATE A EXCEPTION IF TWO ROWS ARE SELECTED TO GATHER THEN IT WILL THROW THE FOLLOWING EXCEPTION..**

```

DECLARE
D_OB DATE;
BEGIN
SELECT DOB INTO D_OB FROM EMPLOYEE WHERE DATE_JOIN='26-FEB-17';
DBMS_OUTPUT.PUT_LINE('BIRTH DATE IS '|| D_OB);
EXCEPTION
WHEN TOO_MANY_ROWS THEN
DBMS_OUTPUT.PUT_LINE('YOUR SELECTOR STATEMENT RETRIVED MULTIPLE
ROWS.SO USE CURSOR ');
END;

```

```
YOUR SELECTOR STATEMENT RETRIVED MULTIPLE ROWS.SO USE CURSOR  
statement processed.  
0.04 seconds
```

## EXCEPTION(USER DEFINED)

### 2.CREATE A EXCEPTION THAT WILL CHECK THE BOOKING DATE AND OF THE DATE IS GONE THAT CREATE FOLLOWING DATE IS GONE EXCEPTION.

```
DECLARE  
B BOOKING.BOOKING_ID%TYPE;  
C BOOKING.BOOKING_DATE%TYPE;  
DATEGONE EXCEPTION;  
  
BEGIN  
B:=BOOKING_ID;  
SELECT BOOKING_DATE INTO C FROM BOOKING WHERE BOOKING_ID=B;  
IF C<SYSDATE THEN  
RAISE DATEGONE;  
ELSE  
DBMS_OUTPUT.PUT_LINE('YOUR DATE IS' || C);  
END IF;  
EXCEPTION  
WHEN DATEGONE THEN  
DBMS_OUTPUT.PUT_LINE('CAR SERVICE DATE IS GONE');  
END;
```

:BOOKING\_ID B111

```
YOUR DATE IS01-JUL-20  
statement processed.  
0.00 seconds
```

## 5.6 TRIGGERS

### 1.CREATE A TRIGGER TO DISPLAY DETAILS OF STATION NAME AFTER THE NAME OF STATION HAS BEEN UPDATED

```
CREATE OR REPLACE TRIGGER ST_trig  
AFTER UPDATE OF SNAME
```

```
ON STATION
FOR EACH ROW
WHEN (OLD.SNAME != NEW.SNAME)
BEGIN
DBMS_OUTPUT.PUT_LINE('STATION Name '
||:OLD.SNAME
||' has change to '
||:NEW.SNAME);
END;
```

```
UPDATE STATION
SET SNAME = 'EXPRESS SERVICE'
WHERE SNAME = 'COMPLETE CAR';
```

```
STATION Name COMPLETE CAR has change to EXPRESS SERVICE

1 row(s) updated.

0.03 seconds
```

## **2.CREATE A TRIGGER TO DISPLAY DETAILS OF PAYMENT AND ALSO THE DIFFERENCE BETWEEN TOTALCOST AND PAYCOST OF PARTICULAR CUSTOMER(I.E ONE TYPE OF BILL IS AUTOMATICALLY GENERATED).**

```
CREATE OR REPLACE TRIGGER "DUE_PAY"
AFTER INSERT OR UPDATE ON PAYMENT
FOR EACH ROW
```

```
DECLARE
CD VARCHAR2(20);
PD VARCHAR2(20);
PDA DATE;
TOT NUMBER(10);
PAY NUMBER(10);
DIFF NUMBER(10);
CN VARCHAR2(20);
A VARCHAR2(20);
```

```
BEGIN
A:=:NEW.PID;
CD:=:NEW.CID;
PD:=:NEW.PMODE;
PDA:=:NEW.PDATE;
```

```
TOT:=NEW.TOTALCOST;  
PAY:=NEW.PAYCOST;  
DIFF:=TOT-PAY;
```

```
SELECT CNAME INTO CN FROM CUSTOMER WHERE CID=CD;
```

```
DBMS_OUTPUT.PUT_LINE('CUSTOMER ID:= '||CD);  
DBMS_OUTPUT.PUT_LINE('CUSTOMER NAME:= '||CN);  
DBMS_OUTPUT.PUT_LINE('PAYMENT MODE:= '||PD);  
DBMS_OUTPUT.PUT_LINE('PAYMENT DATE:= '||PDA);  
DBMS_OUTPUT.PUT_LINE('TOTAL COST:= '||TOT);  
DBMS_OUTPUT.PUT_LINE('PAYMENT COST:= '||PAY);  
DBMS_OUTPUT.PUT_LINE('DUE:= '||DIFF);  
END;
```

```
INSERT  
INTO PAYMENT(PID,CID,PDATE,PMODE,TOTALCOST,PAYCOST)  
VALUES('P118','C101','17-MAY-19','CASH',100000,50000);
```

```
CUSTOMER ID:= C101  
CUSTOMER NAME:= HARSHAD PATEL  
PAYMENT MODE:= CASH  
PAYMENT DATE:= 17-MAY-19  
TOTAL COST:= 100000  
PAYMENT COST:= 50000  
DUE:= 50000
```

```
1 row(s) inserted.
```

```
0.02 seconds
```

## 5.7 CURSOR

### 1.CREATE A CURSOR FOR THE CUSTOMER WHO HAS CHANGED THEIR TYPE OF BOOKTYPE.

```
DECLARE  
ID VARCHAR2(5);  
NEWBOOKINGTYPE VARCHAR2(50);  
BEGIN  
ID:=ENTER_CAR_ID;  
NEWBOOKINGTYPE:=ENTER_NEW_BOOKING_TYPE;  
UPDATE BOOKING SET BOOKTYPE=NEWBOOKINGTYPE WHERE CARID=ID;
```

```

IF SQL%FOUND THEN
DBMS_OUTPUT.PUT_LINE('BOOKTYPE IS UPDATED TO '||NEWBOOKINGTYPE || '
FOR CARID ' || ID);
END IF;
IF SQL%NOTFOUND THEN
DBMS_OUTPUT.PUT_LINE('CUSTOMER NOT FOUND');
END IF;
END;

```

:ENTER_CAR_ID	CR101
:ENTER_NEW_BOOKING_TYPE	DROPTYPE

BOOKTYPE IS UPDATED TO DROPTYPE FOR CARID CR101

Statement processed.

## 2. CREATE A CURSOR TO MAKE A DISCOUNT FOR THE CUSTOMER BASED ON DIFFERENCE BETWEEN TOTALCOST AND PAYMENT HE HAS DONE.

```

DECLARE
CURSOR BONUS IS
SELECT PID,
CID,
PDATE,
PMODE,
TOTALCOST,
PAYCOST

FROM PAYMENT;
DIFF_COST NUMBER(10);
DISCOUNT NUMBER(8,2);
A PAYMENT.PID%TYPE;
B PAYMENT.PID%TYPE;
C PAYMENT.PID%TYPE;
D PAYMENT.PID%TYPE;
E PAYMENT.PID%TYPE;
F PAYMENT.PID%TYPE;
BEGIN
OPEN BONUS;
dbms_output.Put_line('PID'|| ' ' ||'CID' ||' ' || ' PDATE' ||' ' || 'PAYMODE' ||'
'||TOTALCOST'|| ' ||'PAYCOST' ||' ' ||' DISCOUNT');
LOOP
FETCH BONUS INTO A,B,C,D,E,F;

```

```

exit WHEN BONUS%NOTFOUND;
DIFF_COST:=E-F;
IF DIFF_COST = 0 THEN
    DISCOUNT := E * 50 / 100;
ELSIF DIFF_COST BETWEEN 1 AND 10000 THEN
    DISCOUNT := E * 25 / 100;
ELSIF DIFF_COST BETWEEN 10001 AND 20000 THEN
    DISCOUNT := E * 20 / 100;
ELSIF DIFF_COST BETWEEN 20001 AND 30000 THEN
    DISCOUNT := E * 20 / 100;
ELSE
    DISCOUNT:='0';
END IF;
dbms_output.Put_line(A || ' ' || B || ' ' || C || ' ' || D || ' ' || E || ' ' || F || ' ' || DISCOUNT);
END LOOP;
CLOSE BONUS;
END;

```

PID	CID	PDATE	PAYMODE	TOTALCOST	PAYCOST	DISCOUNT
P101	C101	17-MAY-19	CASH	100000	50000	0
P102	C105	13-JAN-19	CHAQUE	85000	85000	42500
P103	C105	15-JAN-19	DD	99999	75000	19999.8
P104	C103	23-FEB-19	ONLINE	69000	69000	34500
P105	C102	06-JUL-19	CASH	75000	5000	0
P106	C110	20-AUG-19	CASH	55000	50000	13750
P107	C107	21-JAN-19	CHAQUE	79000	79000	39500
P108	C107	30-MAR-19	DD	5000	5000	2500
P109	C108	09-SEP-19	ONLINE	29000	10000	5800
P110	C109	30-NOV-19	CASH	84999	40000	0

Statement processed.

## **6. Future Enhancements of the System**

- The System can be expanded by adding more number of Service Station in state/country.
- The tracking facility can be added to the system through which the customer can track the service of their car through GPS when the car will be delivered.
- The detail of Delivery can be more enhanced.
- The automatic generation of status of payment can be done.
- As well as the automatic generation of CID,EID, can be implemented.

## 7. Bibliography

For the successful working of my project I have referred books as well as websites. most I Searched For requirement possession on the websites. Many of the logics in my project are used from the book which I referred & the concepts there in book which indeed provided a platform To achieve such reliable system database.

### Reference books:

- Data Base System Concepts  
- Henry F.Korth & A.Silberschatz. 2nd Ed. McGraw-Hill 1991

### References Website:

- <https://w3shools.com/>