

AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH (AIUB)



FACULTY OF SCIENCE AND TECHNOLOGY

Introduction to Database

Summer 2022-2023

PROJECT TITLE: Tourism Management System.

Group Members:

Name	ID	Contribution
Abul Bashar Saurov	22-48823-3	Table Creation
Kamrul Hasan Tonmoy	22-48825-3	Data Insertion
Ahasan Habib	22-48877-3	Query
Md Kamruzzamn Jubo	22-47692-2	Normalization

Date of submission:
August 9, 2023

Contents

1	INTRODUCTION	3
2	SCENARIO	3
3	ER DIAGRAM	4
4	NORMALIZATION	5
4.1	Employee – works for – Travel Agency	5
4.2	Travel Agency – services – Customer	7
4.3	Customer – takes – Flight	8
4.4	Customer – uses – Local_Transportation	9
4.5	Customer – makes _r eservation–Hotel	10
4.6	Customer – gets - Accommodation	11
4.7	Hotel – offers – Accommodation	12
4.8	Hotel – manages – Food _B ooking	13
4.9	Customer – makes – Payment	14
5	FINALIZATION	15
6	OPTIMAIZATION	17
7	TABLE CREATION	18
7.1	Agency_ID:	18
7.2	Agency_Info :	18
7.3	Agency_Con:	19
7.4	EMP_Info:	19
7.5	EMP_Skill:	20
7.6	Cus_Info:	20
7.7	Cus_Loc:	21
7.8	Cus_Con:	21
7.9	Airline:	22
7.10	Airline_Info:	22
7.11	Transportation:	23
7.12	Hotel_Info:	23
7.13	Accommodation:	24
7.14	Food_Booking:	25
7.15	Hotel_Ratings:	25
7.16	Payment:	26
8	DATA INSERTION:	27
8.1	Agency_Info	27
8.2	Agency_Loc	27
8.3	Agency_con	27
8.4	EMP_Info	28
8.5	EMP_Skill	28

8.6	CUS_Info	29
8.7	CUS_Loc	30
8.8	CUS_Con	30
8.9	Airline	31
8.10	Airline_Info	31
8.11	Transportation	32
8.12	Hotel_Info	32
8.13	Accommodation	33
8.14	Food_Booking	33
8.15	Hotel_Ratings	34
8.16	Payment	34
9	QUERY WRITING	36
9.1	Single Row Query	36
9.2	Multiple Row Query	37
9.3	SUB QUERY	38
9.4	JOINING QUERY	39
9.5	VIEW	40
10	CONCLUSION	42

1 INTRODUCTION

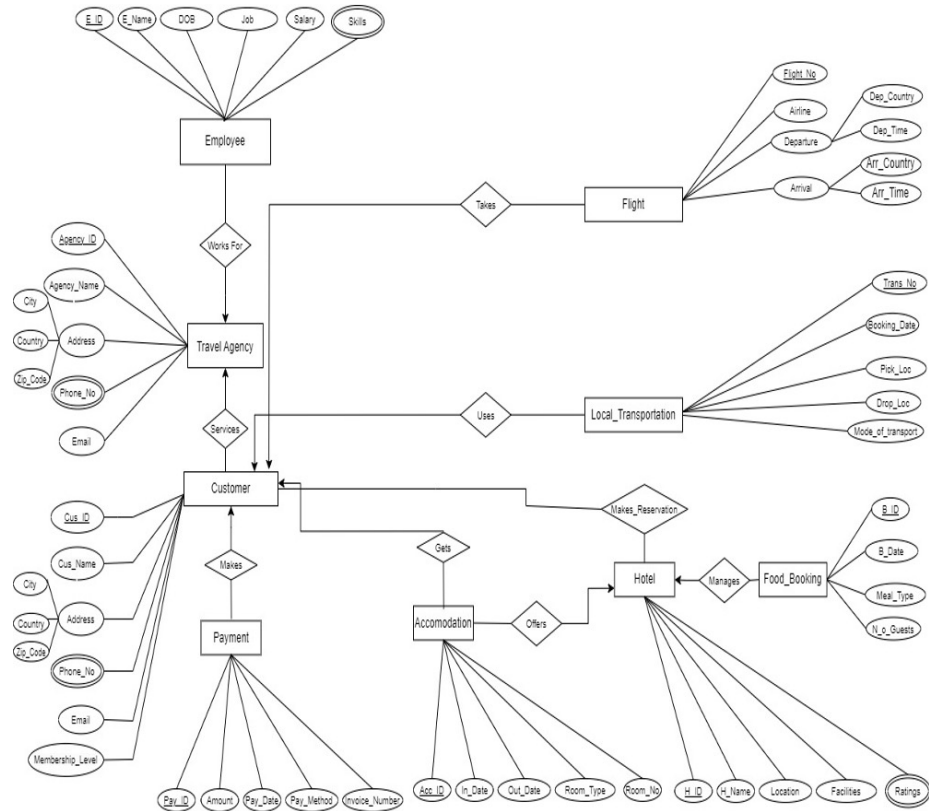
In today's fast-paced world, managing travel arrangements has become increasingly complex. Our project is based on a tourism management system which aims to establish a comprehensive and efficient solution for managing travel arrangements. By creating a centralized database, this project seeks to streamline the process of storing and retrieving critical travel information such as flight bookings, hotel reservations, food, transportation, and more. This system benefits both travel agencies and individual users by providing improved efficiency and a seamless travel experience.

2 SCENARIO

In a tourism management system, a travel agency employs various types of employees to handle tasks. A travel agency contains agency ID, name, address, phone number, and email address. An employee has an employee ID, name, date of birth, job type, salary, and skills. Customers register with customer ID, name, address, phone number, email, and membership level.

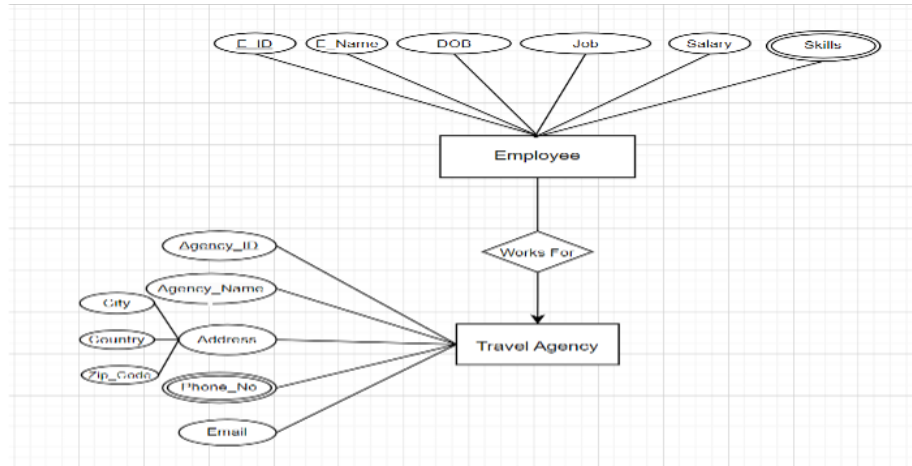
Various addresses are composed of city, country, and zip code. Customers make payments for facilities, take flights, use local transportation, and make reservations at hotels. Hotels offer accommodations, and customers get accommodations. Payment processes are tracked using payment ID, amount, payment date, payment method, and invoice number.

3 ER DIAGRAM



4 NORMALIZATION

4.1 Employee – works for – Travel Agency



Relation: Many to One

UNF:

- Agency ID
- Agency_Name
- City
- Country
- Zip_code
- Phone_No
- Email
- E_ID
- E_Name
- DOB (Date of Birth)
- Job Title
- Salary
- Skills

1NF:

- Agency ID

- Agency_Name
- City
- Country
- Zip_code
- Email
- E_ID
- E_Name
- DOB (Date of Birth)
- Job Title
- Salary

2NF:

1. Agency ID, Phone_No
2. E_ID, Skills
3. Agency ID, Agency_Name, City, Country, Zip_code, Email
4. E_ID, E_Name, DOB, Job Title, Salary, Agency ID

3NF:

1. Agency ID, City, Country, Zip_code
2. Agency ID, Phone_No
3. E_ID, Skills
4. Agency ID, Agency_Name, Email
5. E_ID, E_Name, DOB, Job Title, Salary, Agency ID

4.2 Travel Agency – services – Customer



Relation: One to Many

UNF: Agency_ID, Agency_Name, City, Country, Zip_Code, Phone_No, Email, Cus_ID, Cus_Name, City, Country, Zip-code, Email, Phone_No, Membership_Level

1NF:

Multivalued attributes: Phone_No, Phone.No. Agency_ID, Agency_Name, City, Country, Zip-code, Email, Cus_ID, Cus_Name, City, Country, Zip-code, Email, Membership_Level

2NF:

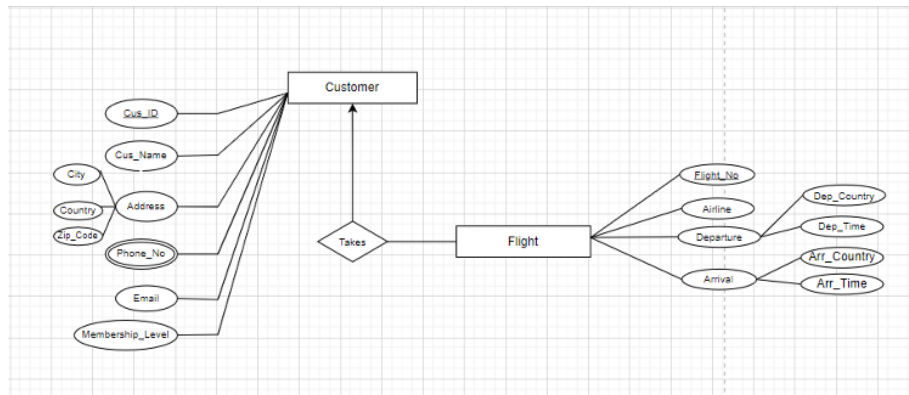
1. Agency_ID, Phone.No
2. Cus_ID, Phone.No
3. Agency_ID, Agency_Name, City, Country, Zip-code, Email
4. Cus_ID, Cus_Name, City, Country, Zip_Code, Email, Membership_Level, Agency_ID

3NF:

1. Agency_ID, City, Country, Zip_Code
2. Cus_ID, City, Country, Zip_Code

3. Agency_ID,Phone_No
4. Cus_ID,Phone_No
5. Agency_ID,Agency_Name,Email
6. Cus_ID,Cus_Name,Email,Membership_Level,Agency_ID

4.3 Customer – takes – Flight



Relation: One to Many

UNF: Cus_ID, Cus_Name, City, Country, Zip_code, Email, phone_no, Membership_Level, Flight_No, Airline, Dep_Country, Arr_Country, Dep_Time, Arr_Time

1NF: Multivalued attributes: Phone_No Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level, Flight_No, Airline, Dep_Country, Arr_Country, Dep_Time, Arr_Time

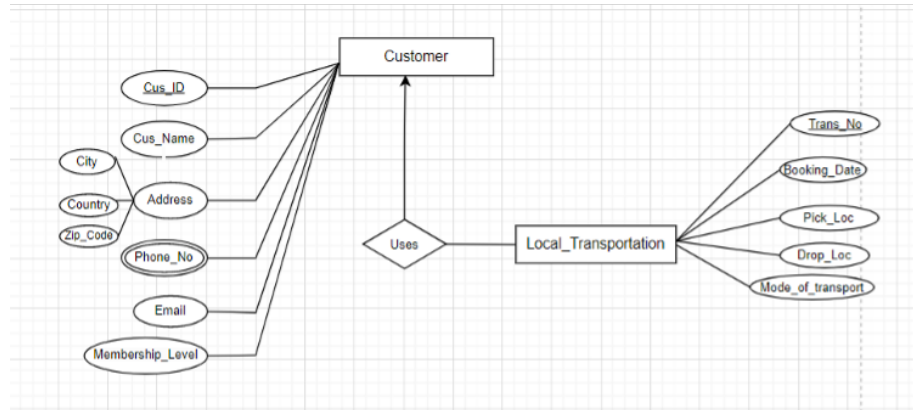
2NF:

1. Cus_ID, Phone_No
2. Flight_No, Airline, Dep_Country, Arr_Country, Dep_Time, Arr_Time, Cus_ID
3. Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level

3NF:

1. Cus_ID, Phone_No
2. Flight_No, Dep_Country, Arr_Country, Dep_Time, Arr_Time
3. Cus_ID, City, Country, Zip_code
4. Cus_ID, Cus_Name, Email, Membership_Level
5. Flight_No, Airline, Cus_ID

4.4 Customer – uses – Local Transportation



Relation: One to Many

UNF:

Cus_ID, Cus_Name, City, Country, Zip_code, Email, Phone_No, Membership_Level, Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode_of_Transport

1NF:

Multivalued attributes: Phone_No Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level, Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode_of_Transport

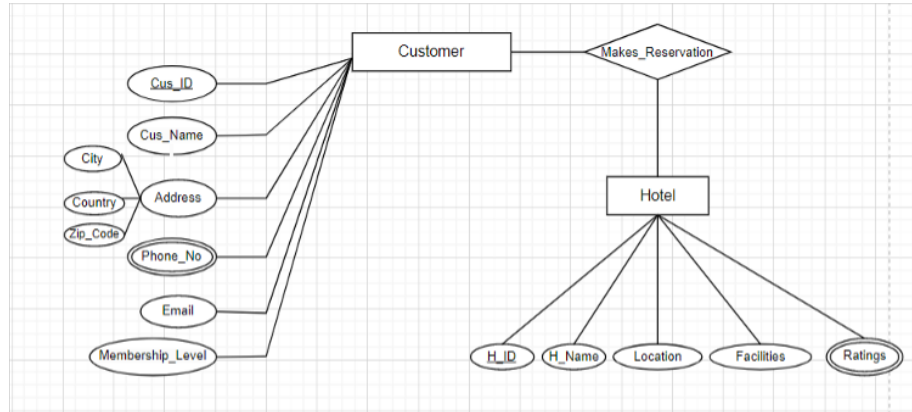
2NF:

1. Cus_ID, Phone_No
2. Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode_of_Transport, Cus_ID
3. Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level

3NF:

1. Cus_ID, City, Country, Zip_code
2. Cus_ID, Phone_No
3. Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode of Transport, Cus_ID
4. Cus_ID, Cus_Name, Email, Membership_Level

4.5 Customer – makes_reservation–Hotel



Relation: Many to Many

UNF:

Cus_ID, Cus_Name, City, Country, Zip_code, Email, Phone_No, Membership_Level, H_ID, H_Name, Location, Facilities, Ratings

1NF:

Multivalued attributes: Phone_No, Ratings Cus_ID, Cus_Name, City, Country, Zip_code, Email, Phone_No, Membership_Level, H_ID, H_Name, Location, Facilities

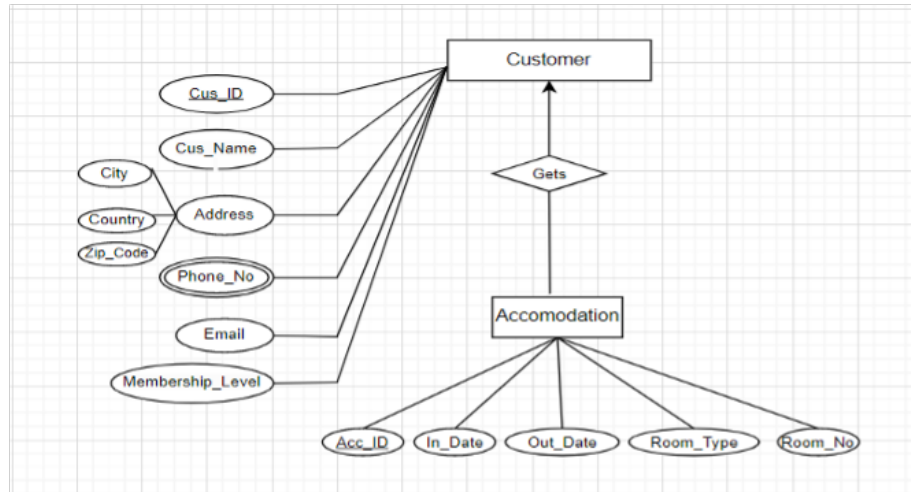
2NF:

1. Cus_ID, Phone_No
2. H_ID, Ratings
3. Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level
4. H_ID, H_Name, Location, Facilities, Cus_ID

3NF:

1. Cus_ID, City, Country, Zip_code
2. Cus_ID, Phone_No
3. H_ID, Ratings
4. Cus_ID, Cus_Name, Email, Membership_Level
5. H_ID, H_Name, Location, Facilities, Cus_ID

4.6 Customer – gets - Accommodation



Relation: One to Many

UNF:

Cus_ID, Cus_Name, City, Country, Zip_code, Email, phone_no, Membership_Level, Acc_ID, In_Date, Out_Date, Room_Type, Room_No

1NF:

Multivalued attributes: Phone_No Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level, Acc_ID, In_Date, Out_Date, Room_Type, Room_No

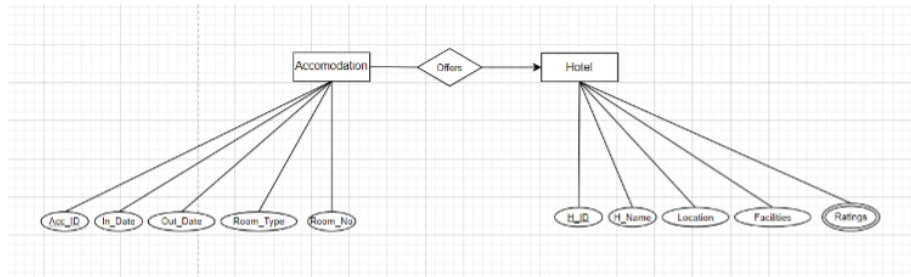
2NF:

1. Cus_ID, Phone_No
2. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, Cus_ID
3. Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level

3NF:

1. Cus_ID, City, Country, Zip_code
2. Cus_ID, Phone_No
3. Cus_ID, Cus_Name, Email, Membership_Level
4. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, Cus_ID

4.7 Hotel – offers – Accommodation



Relation: One to Many

UNF:

Acc_ID, In_Date, Out_Date, Room_Type, Room_No, H_ID, H_name, location, Facilities, Ratings

1NF:

Multivalued attributes: Ratings Acc_ID, In_Date, Out_Date, Room_Type, Room_No, H_ID, H_name, location, Facilities

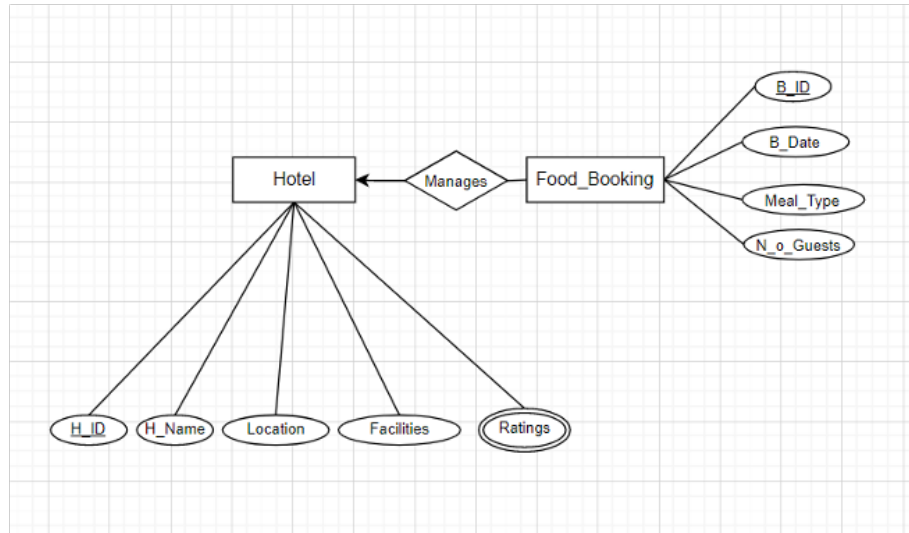
2NF:

1. H_ID, Ratings
2. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, H_ID
3. H_ID, H_name, location, Facilities

3NF:

1. H_ID, Ratings
2. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, H_ID
3. H_ID, H_name, location, Facilities

4.8 Hotel – manages – Food_{Booking}



One to Many

UNF:

H_ID, H_name, location, Facilities, Ratings, B_ID, B_Date, Meal_Type, N_o_Guests

1NF:

Multivalued attributes: Ratings H_ID, H_name, location, Facilities, B_ID, B_Date, Meal_Type, N_o_Guests

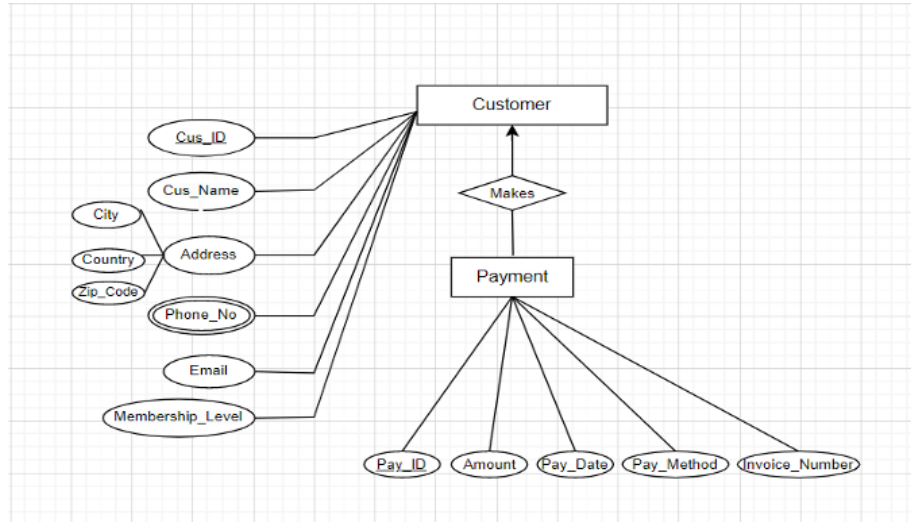
2NF:

1. H_ID, Ratings
2. H_ID, H_name, location, Facilities
3. B_ID, B_Date, Meal_Type, N_o_Guests, H_ID

3NF:

1. H_ID, Ratings
2. H_ID, H_name, location, Facilities
3. B_ID, B_Date, Meal_Type, N_o_Guests, H_ID

4.9 Customer – makes – Payment



Relation: One to Many

UNF:

Cus_ID, Cus_Name, City, Country, Zip_code, Email, phone_no, Membership_Level, Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number, Phone_No

1NF:

Multivalued attributes: Phone_No Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level, Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number Phone_No

2NF:

1. Cus_ID, Phone_No
2. Cus_ID, Cus_Name, City, Country, Zip_code, Email, Membership_Level
3. Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number, Cus_ID

3NF:

1. Cus_ID, City, Country, Zip_code
2. Cus_ID, Phone_No
3. Cus_ID, Cus_Name, Email, Membership_Level
4. Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number, Cus_ID

5 FINALIZATION

1. Agency_ID,City,Country,Zip_code
2. Agency_ID,Phone No
3. E_ID, Skills
4. Agency_ID,Agency_Name, Zip_code,Email
5. E_ID, E_Name, DOB, Job_Title, Salary, **Agency_ID**
6. Agency_Id, City, Country,Zip_code
7. Cus_ID, City, Country,Zip_code
8. Agency_ID, Phone_No
9. Cus_ID,Phone_No
10. Agency_Id, Agency Name, Email
11. Cus_ID, Cus_Name,Email, Membership_Level,**Agency_ID**
12. Cus_ID, City, Country, Zip_code
13. Cus_ID, Phone_No
14. Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode of Transport, **Cus_Id**
15. Cus_ID, Cus_Name, Email, Membership_Level
16. Cus_ID,City, Country,Zip_code
17. Cus_ID,Phone_No
18. H_ID, Ratings
19. Cus_ID,Cus_Name,Email,Membership_Level
20. H_ID, H_Name, Locations, Facilities, **Cus_ID**
21. Cus_ID, City, Country, Zip_code
22. Cus_ID,Phone_No
23. Cus_ID, Cus_Name, Email, Membership_Level
24. Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number,Cus_ID
25. H_ID, Ratings
26. H_ID, H_Name, Location, Facilities
27. B_ID, B_Date, Meal_Type, N_o_Guests,**H_ID**

- 28. Cus_ID, City, Country, Zip_code
- 29. Cus_ID, Phone_No
- 30. Cus_ID, Cus_Name, Email, Membership_Level
- 31. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, **Cus_ID**
- 32. H_ID, Ratings
- 33. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, **H_ID**
- 34. H_ID, H_Name, Locations, Facilities
- 35. Cus_ID, Phone_No
- 36. Flight_No, Dep_Country, Arr_Country, Dep_Time
- 37. Cus_ID, City, Country, Zip_code
- 38. Cus_ID, Email, Membership_Level
- 39. Flight_No, Airines, **Cus_ID**

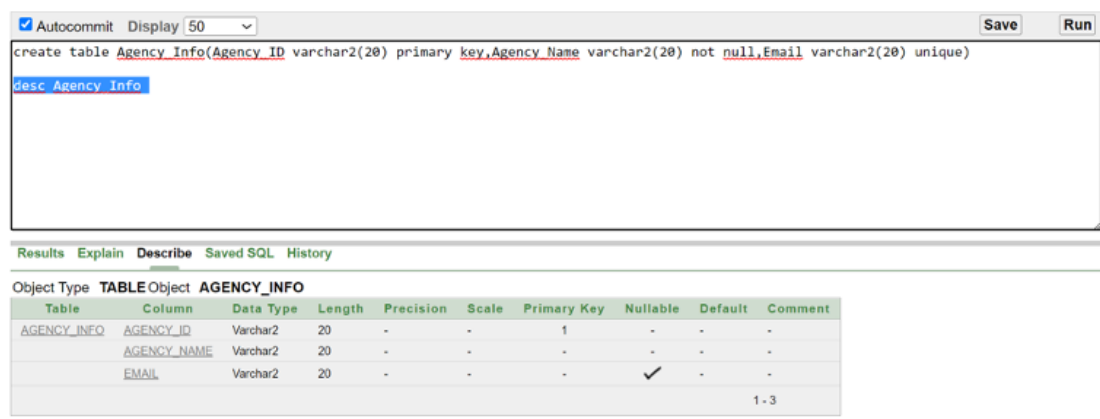
6 OPTIMAIZATION

1. Agency_ID, City, Zip_code -(Agency_Loc)
2. Agency_ID, Phone_No -(Agency_Con)
3. Agency_ID, Agency_Name, Email -(Agency_Info)
4. E_ID, Skills -(EMP_Skill)
5. E_ID, E_Name, DOB, Job_Title, Salary, **Agency_ID** -(EMP_Info)
6. Cus_ID City, Country, Zip_code – (CUS_Loc)
7. Cus_ID, Phone_No -(Cus_con)
8. Cus_ID, Cus_Name, Email, Membership_Level, **Agency_ID** – (CUS_Info)
9. Flight_No, Dep_Country, Arr_Country, Dep_Time, Arr_Time - (Airline_Info)
10. Flight_No, Airline, **Cus_ID** -(Airline)
11. Trans_No, Booking_Date, Pick_Loc, Drop_Loc, Mode_of_Transport, **Cus_ID** -(Transportation)
12. H_ID, Name, Location, Facilities, **Cus_ID** -(Hotel_Info)
13. H_ID, Ratings -(Hotel_Ratings)
14. Acc_ID, In_Date, Out_Date, Room_Type, Room_No, **Cus_ID**, **H_ID** -(Accommodation)
15. B_ID, Meal_Type, N_o_Guests, **H_ID** -(Food_Booking)
16. Pay_ID, Amount, Pay_Date, Pay_Method, Invoice_Number, **Cus_ID** - (Payment)

7 TABLE CREATION

7.1 Agency_ID:

CREATE TABLE Agency_Info(Agency_ID varchar2(20) primary key,Agency_Name varchar2(20) not null, Email varchar2(20) unique) desc Agency_Info



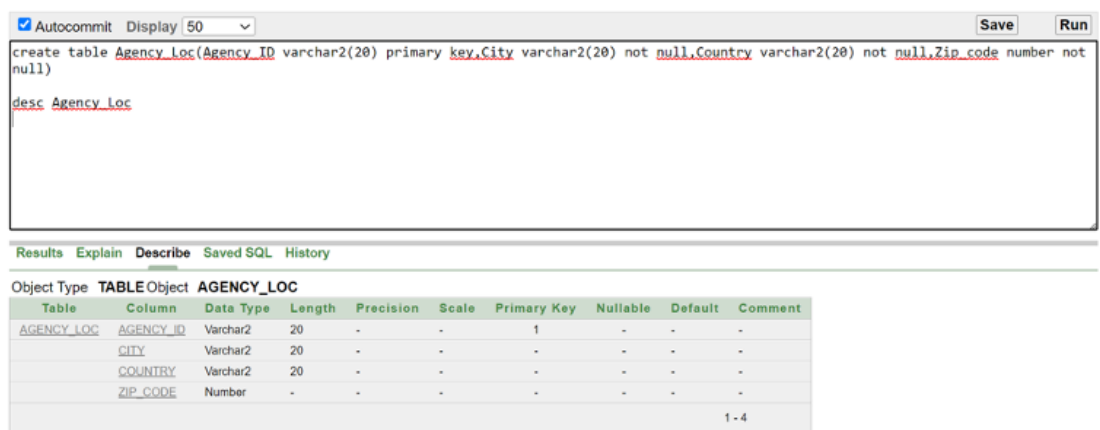
The screenshot shows a SQL IDE interface. At the top, there are buttons for 'Autocommit' (checked), 'Display' (set to 50), 'Save', and 'Run'. Below these, the SQL command 'create table Agency_Info(Agency_ID varchar2(20) primary key,Agency_Name varchar2(20) not null,Email varchar2(20) unique)' is entered, followed by 'desc Agency_Info;'. The 'Describe' tab is selected, displaying the table structure for 'AGENCY_INFO'.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AGENCY_INFO	AGENCY_ID	Varchar2	20	-	-	1	-	-	-
	AGENCY_NAME	Varchar2	20	-	-	-	-	-	-
	EMAIL	Varchar2	20	-	-	-	✓	-	-

1 - 3

7.2 Agency_Info :

CREATE TABLE Agency_Loc(Agency_ID varchar2(20) primary key,City varchar2(20) not null,Country varchar2(20) not null,Zip_code number not null) desc Agency_Loc



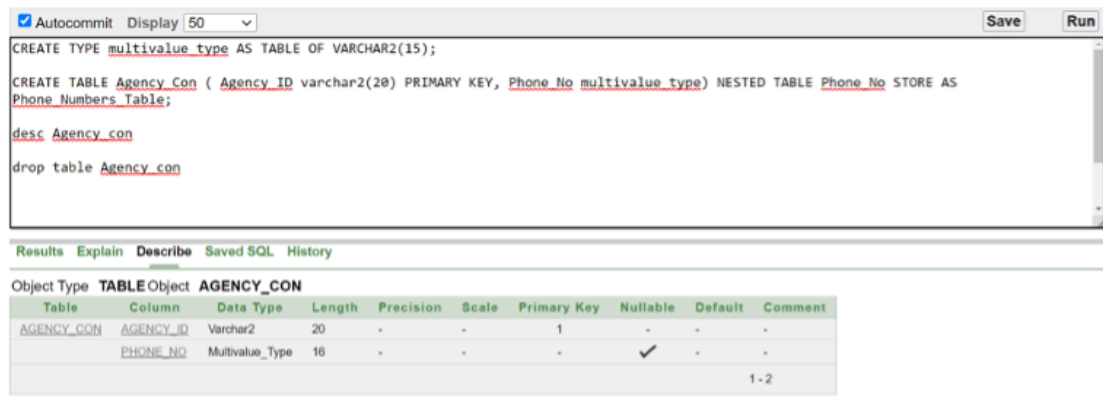
The screenshot shows a SQL IDE interface. At the top, there are buttons for 'Autocommit' (checked), 'Display' (set to 50), 'Save', and 'Run'. Below these, the SQL command 'create table Agency_Loc(Agency_ID varchar2(20) primary key,City varchar2(20) not null,Country varchar2(20) not null,Zip_code number not null)' is entered, followed by 'desc Agency_Loc;'. The 'Describe' tab is selected, displaying the table structure for 'AGENCY_LOC'.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AGENCY_LOC	AGENCY_ID	Varchar2	20	-	-	1	-	-	-
	CITY	Varchar2	20	-	-	-	-	-	-
	COUNTRY	Varchar2	20	-	-	-	-	-	-
	ZIP_CODE	Number	-	-	-	-	-	-	-

1 - 4

7.3 Agency_Con:

```
CREATE TYPE multivalue_type AS TABLE OF VARCHAR2(15); CREATE
TABLE Agency_Con ( Agency_ID varchar2(20) PRIMARY KEY, Phone_No
multivalue_type) NESTED TABLE Phone_No STORE AS Phone_Numbers_Table;
desc Agency_con
drop table Agency_con
```



The screenshot shows the SQL Developer interface. The top toolbar includes 'Autocommit' (checked), 'Display' (set to 50), 'Save', and 'Run' buttons. The SQL editor contains the following code:

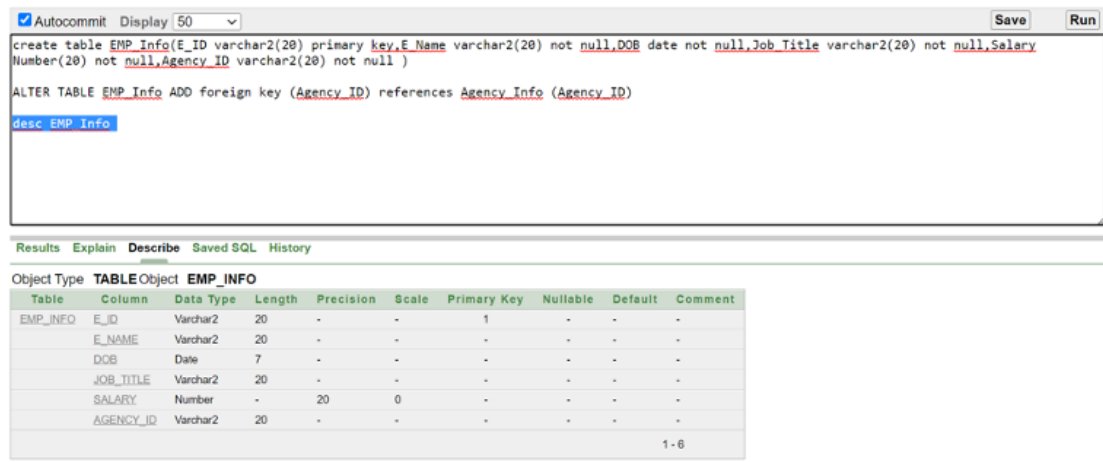
```
CREATE TYPE multivalue_type AS TABLE OF VARCHAR2(15);
CREATE TABLE Agency_Con ( Agency_ID varchar2(20) PRIMARY KEY, Phone_No multivalue_type) NESTED TABLE Phone_No STORE AS Phone_Numbers_Table;
desc Agency_con
drop table Agency_con
```

Below the editor, the 'Results' tab is active, displaying the table structure for 'AGENCY_CON'.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AGENCY_CON	AGENCY_ID	Varchar2	20	-	-	1	-	-	-
	PHONE_NO	Multivalue_Type	16	-	-	-	✓	-	-
1-2									

7.4 EMP_Info:

```
CREATE TABLE EMP_Info(E_ID varchar2(20) primary key,E_Name varchar2(20)
not null,DOB date not null,Job_Title varchar2(20) not null,Salary Number(20)
not null,Agency_ID varchar2(20) not null )
ALTER TABLE EMP_Info ADD foreign key (Agency_ID) references Agency_Info
(Agency_ID)
desc EMP_Info
```



The screenshot shows the SQL Developer interface. The top toolbar includes 'Autocommit' (checked), 'Display' (set to 50), 'Save', and 'Run' buttons. The SQL editor contains the following code:

```
create table EMP_Info(E_ID varchar2(20) primary key,E_Name varchar2(20) not null,DOB date not null,Job_Title varchar2(20) not null,Salary
Number(20) not null,Agency_ID varchar2(20) not null )
ALTER TABLE EMP_Info ADD foreign key (Agency_ID) references Agency_Info (Agency_ID)
desc EMP_Info
```

Below the editor, the 'Results' tab is active, displaying the table structure for 'EMP_INFO'.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP_INFO	E_ID	Varchar2	20	-	-	1	-	-	-
	E_NAME	Varchar2	20	-	-	-	-	-	-
	DOB	Date	7	-	-	-	-	-	-
	JOB_TITLE	Varchar2	20	-	-	-	-	-	-
	SALARY	Number	-	20	0	-	-	-	-
	AGENCY_ID	Varchar2	20	-	-	-	-	-	-
1-6									

7.5 EMP_Skill:

```
CREATE TYPE multivalue_type AS TABLE OF VARCHAR2(15);
CREATE TABLE EMP_Skill ( E_ID varchar2(20) PRIMARY KEY, skills
multivalue_type) NESTED TABLE skills STORE AS Phone_Numbers_Table4;
desc EMP_Skill
```

<input checked="" type="checkbox"/> Autocommit Display 50 Save Run									
<pre>CREATE TYPE multivalue_type AS TABLE OF VARCHAR2(15); CREATE TABLE EMP_Skill (E_ID varchar2(20) PRIMARY KEY, skills multivalue_type) NESTED TABLE skills STORE AS Phone_Numbers_Table4; desc EMP_Skill</pre>									
<div>Results Explain Describe Saved SQL History</div>									
Object Type TABLE Object EMP_SKILL									
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP_SKILL	E_ID	Varchar2	20	-	-	1	-	-	-
	SKILLS	Multivalue_Type	16	-	-	-	✓	-	-
1 - 2									

7.6 Cus_Info:

```
CREATE TABLE CUS_Info(CUS_ID varchar2(20) PRIMARY KEY, Cus_Name
varchar2(100) NOT NULL, Email varchar2(100) UNIQUE, Membership_Level
varchar2(20), Agency_ID varchar2(20));
ALTER TABLE CUS_Info add constraint agency FOREIGN KEY (Agency_ID)
references Agency_Info (Agency_ID) ;
DESC CUS_Info;
```

<input checked="" type="checkbox"/> Autocommit Display 50 Save Run									
<pre>CREATE TABLE CUS_Info(CUS_ID varchar2(20) PRIMARY KEY, Cus_Name varchar2(100) NOT NULL, Email varchar2(100) UNIQUE, Membership_Level varchar2(20), Agency_ID varchar2(20)); ALTER TABLE CUS_Info add constraint agency FOREIGN KEY (Agency_ID) references Agency_Info (Agency_ID) ; DESC CUS_Info;</pre>									
<div>Results Explain Describe Saved SQL History</div>									
Object Type TABLE Object CUS_INFO									
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUS_INFO	CUS_ID	Varchar2	20	-	-	1	-	-	-
	CUS_NAME	Varchar2	100	-	-	-	-	-	-
	EMAIL	Varchar2	100	-	-	-	✓	-	-
	MEMBERSHIP_LEVEL	Varchar2	20	-	-	-	✓	-	-
	AGENCY_ID	Varchar2	20	-	-	-	✓	-	-
1 - 5									

7.7 Cus_Loc:

```
CREATE TABLE CUS_Loc(CUS_ID varchar2(20) PRIMARY KEY, country
varchar2(20) NOT NULL, city varchar2(20) NOT NULL, Zip_Code varchar2(20));
DESC CUS_Loc
```

The screenshot shows a SQL Developer window with the following SQL commands entered:

```
CREATE TABLE CUS_Loc(CUS_ID varchar2(20) PRIMARY KEY, country varchar2(20) NOT NULL, city varchar2(20) NOT NULL, Zip_Code varchar2(20));
DESC CUS_Loc;
```

The results pane displays the table structure for **CUS_LOC**:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUS_LOC	CUS_ID	Varchar2	20	-	-	1	-	-	-
	COUNTRY	Varchar2	20	-	-	-	-	-	-
	CITY	Varchar2	20	-	-	-	-	-	-
	ZIP_CODE	Varchar2	20	-	-	-	✓	-	-

7.8 Cus_Con:

```
DESC cus_con;
INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '10', multivalue_type('018*****',
'019*****')); INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES (
'20', multivalue_type('01811*****', '01822*****')); INSERT INTO cus_con(CUS_ID,
PHONE_NO) VALUES ( '30', multivalue_type('018123*****', '015678*****',
'016*****'));
SELECT CUS_ID, COLUMN_VALUE AS PHONE_NO FROM cus_con ,
TABLE(cus_con.PHONE_NO);
```

The screenshot shows a SQL Developer window with the following SQL commands entered:

```
CREATE TYPE multivalue_type AS TABLE OF VARCHAR2(15);
CREATE TABLE cus_con ( cus_id varchar(20) PRIMARY KEY, Phone_No multivalue_type) NESTED TABLE Phone_No STORE AS Phone_Numbers_Table1;
desc cus_con
```

The results pane displays the table structure for **CUS_CON**:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUS_CON	CUS_ID	Varchar2	20	-	-	1	-	-	-
	PHONE_NO	Multivalue_Type	16	-	-	-	✓	-	-

7.9 Airline:

desc Airline

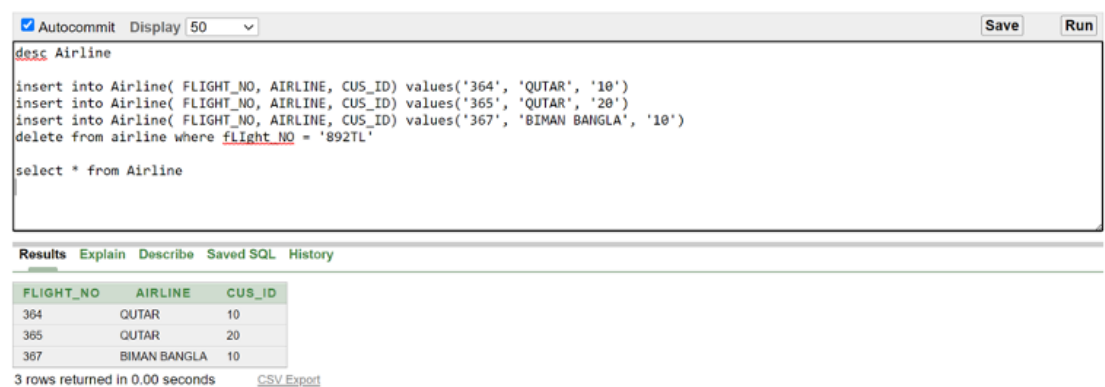
```
INSERT INTO Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('364', 'QUTAR', '10')
```

```
INSERT INTO Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('365', 'QUTAR', '20')
```

```
INSERT INTO Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('367', 'BIMAN BANGLA', '10')
```

```
delete from airline where flight_NO = '892TL'
```

```
select * from Airline
```



The screenshot shows a SQL IDE interface. At the top, there's a toolbar with 'Autocommit' checked, 'Display' set to 50, and 'Save' and 'Run' buttons. Below the toolbar is a text area containing SQL commands. The commands are: 'desc Airline', 'insert into Airline(FLIGHT_NO, AIRLINE, CUS_ID) values('364', 'QUTAR', '10')', 'insert into Airline(FLIGHT_NO, AIRLINE, CUS_ID) values('365', 'QUTAR', '20')', 'insert into Airline(FLIGHT_NO, AIRLINE, CUS_ID) values('367', 'BIMAN BANGLA', '10')', 'delete from airline where flight_NO = '892TL'', and 'select * from Airline'. Below the text area is a 'Results' tab, which is active. It shows a table with three columns: 'FLIGHT_NO', 'AIRLINE', and 'CUS_ID'. The table contains three rows of data: (364, QUTAR, 10), (365, QUTAR, 20), and (367, BIMAN BANGLA, 10). Below the table, it says '3 rows returned in 0.00 seconds' and there is a 'CSV Export' link.

```
desc Airline
```

```
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('364', 'QUTAR', '10')
```

```
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('365', 'QUTAR', '20')
```

```
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('367', 'BIMAN BANGLA', '10')
```

```
delete from airline where flight_NO = '892TL'
```

```
select * from Airline
```

FLIGHT_NO	AIRLINE	CUS_ID
364	QUTAR	10
365	QUTAR	20
367	BIMAN BANGLA	10

3 rows returned in 0.00 seconds [CSV Export](#)

7.10 Airline_Info:

desc Airline_Info

```
INSERT INTO Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('892TL', 'BANGLADESH', 'USA', '26-NOV-22', '28-NOV-22')
```

```
INSERT INTO Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('367FC', 'USA', 'UK', '03-MAY-22', '04-MAY-22')
```

```
select * from Airline_Info
```

<input checked="" type="checkbox"/> Autocommit Display 50 Save Run					
<pre> desc Airline_Info insert into Airline_Info(FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('892TL','BANGLADESH','USA','26-NOV-22','28-NOV-22') insert into Airline_Info(FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('367FC','USA','UK','03-MAY-22','04-MAY-22') select * from Airline_Info </pre>					
Results Explain Describe Saved SQL History					
FLIGHT_NO	DEP_COUNTRY	ARR_COUNTRY	DEP_TIME	ARR_TIME	
892TL	BANGLADESH	USA	26-NOV-22	28-NOV-22	
367FC	USA	UK	03-MAY-22	04-MAY-22	
2 rows returned in 0.00 seconds CSV Export					

7.11 Transportation:

DESC transportation

```
INSERT INTO transportation( TRANS_NO, BOOKING_DATE, PICK_LOC,
DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('T245GF','21-JUL-23',
'UTTARA','KURIL','AC_CAR','10')
```

```
INSERT INTO transportation( TRANS_NO, BOOKING_DATE, PICK_LOC,
DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('F259OG','13-MAY-23',
'AZIMPUR','CUMILLA','AC_TRAIN','20')
```

```
INSERT INTO transportation( TRANS_NO, BOOKING_DATE, PICK_LOC,
DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('O2T59F','01-APR-233',
'RAJSHAI','DHAKA','AC_BUS','30')
```

```
SELECT * FROM transportation
```

<input checked="" type="checkbox"/> Autocommit Display 50 Save Run					
<pre> DESC transportation insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('T245GF','21-JUL-23', 'UTTARA','KURIL','AC_CAR','10') insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('F259OG','13-MAY-23', 'AZIMPUR','CUMILLA','AC_TRAIN','20') insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('O2T59F','01-APR-233', 'RAJSHAI','DHAKA','AC_BUS','30') SELECT * FROM transportation </pre>					
Results Explain Describe Saved SQL History					
TRANS_NO	BOOKING_DATE	PICK_LOC	DROP_LOC	MODE_OF_TRANSPORT	CUS_ID
T245GF	21-JUL-23	UTTARA	KURIL	AC_CAR	10
F259OG	13-MAY-23	AZIMPUR	CUMILLA	AC_TRAIN	20
O2T59F	01-APR-33	RAJSHAI	DHAKA	AC_BUS	30
3 rows returned in 0.00 seconds CSV Export					

7.12 Hotel_Info:

DESC Hotel_Info

```
INSERT INTO Hotel_Info (H.ID, H.NAME, LOCATION, FACILITES, CUS.ID)
values('63K9','GHI','DHAKA','SWIMMING','10')
```



```

INSERT INTO Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID)
values('26R0','BCD', 'CUMILLA', 'PARKING','20')
INSERT INTO Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID)
values('55SK','JKL', 'MYMENSINGH', 'CONFERENCE','30')
SELECT * FROM Hotel_Info

```

The screenshot shows a database management interface with a SQL editor and a results pane. The SQL editor contains the following commands:

```

DESC Hotel_Info
insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('63K9','GHI', 'DHAKA', 'SWIMMING','10')
insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('26R0','BCD', 'CUMILLA', 'PARKING','20')
insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('55SK','JKL', 'MYMENSINGH', 'CONFERENCE','30')
SELECT * FROM Hotel_Info

```

The results pane shows the output of the SQL commands. It includes a table with 5 columns: H_ID, H_NAME, LOCATION, FACILITES, and CUS_ID. The table contains 3 rows of data. Below the table, it states "3 rows returned in 0.02 seconds" and provides a link to "CSV Export".

H_ID	H_NAME	LOCATION	FACILITES	CUS_ID
63K9	GHI	DHAKA	SWIMMING	10
26R0	BCD	CUMILLA	PARKING	20
55SK	JKL	MYMENSINGH	CONFERENCE	30

7.13 Accommodation:

```

insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE,
ROOM_NO, CUS_ID, H_ID) values('23-43','21-JUL-23', '24-JUL-23', 'SINGLE
BED',501,'10','63K9')
insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE,
ROOM_NO, CUS_ID, H_ID) values('23-62','13-MAY-23', '16-MAY-23', 'DOU-
BLE BED',509,'20','26R0')
insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE,
ROOM_NO, CUS_ID, H_ID) values('23-65','01-APR-23', '10-APR-23', 'SINGLE
BED',203,'30','55SK')
SELECT * FROM Accommodation

```

Autocommit Display 50 Save Run

```

desc Accommodation;

insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, CUS_ID, H_ID) values('23-43','21-JUL-23', '24-JUL-23', 'SINGLE
BED',501,'10','63K9')
insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, CUS_ID, H_ID) values('23-62','13-MAY-23', '16-MAY-23', 'DOUBLE
BED',509,'20','26R0')
insert into Accommodation( ACC_ID, IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, CUS_ID, H_ID) values('23-65','01-APR-23', '10-APR-23', 'SINGLE
BED',203,'30','55SK')

SELECT * FROM Accommodation

```

Results Explain Describe Saved SQL History

ACC_ID	IN_DATE	OUT_DATE	ROOM_TYPE	ROOM_NO	CUS_ID	H_ID
23-43	21-JUL-23	24-JUL-23	SINGLE BED	501	10	63K9
23-65	01-APR-23	10-APR-23	SINGLE BED	203	30	55SK
23-62	13-MAY-23	16-MAY-23	DOUBLE BED	509	20	26R0

3 rows returned in 0.00 seconds [CSV Export](#)

7.14 Food_Booking:

desc Food_Booking

```
insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N.O.GUESTS,
H_ID) values('1001','22-JUL-23', 'LUNCH', '5','63K9')
```

```
insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N.O.GUESTS,
H_ID) values('2030','12-SEP-23', 'DINNER', '3','26R0')
```

```
insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N.O.GUESTS,
H_ID) values('3002','03-NOV-23', 'BREAKFAST', '6','55SK')
```

```
SELECT * FROM Food_Booking
```

Autocommit Display 50 Save Run

```

desc Food_Booking

insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('1001','22-JUL-23', 'LUNCH', '5','63K9')

insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('2030','12-SEP-23', 'DINNER', '3','26R0')

insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('3002','03-NOV-23', 'BREAKFAST', '6','55SK')

SELECT * FROM Food_Booking

```

Results Explain Describe Saved SQL History

B_ID	B_DATE	MEAL_TYPE	N_O_GUESTS	H_ID
1001	22-JUL-23	LUNCH	5	63K9
2030	12-SEP-23	DINNER	3	26R0
3002	03-NOV-23	BREAKFAST	6	55SK

3 rows returned in 0.00 seconds [CSV Export](#)

7.15 Hotel_Ratings:

desc Hotel_Ratings

```
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('63K9', mul-
tivalue_type('GOOD', 'EXCELLENT')); INSERT INTO Hotel_Ratings(H_ID,
```

```

RATINGS) VALUES ('55SK', multivalue_type('VERY GOOD', 'NOT BAD'));
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('26R0', multivalue_type('EXCELLENT'));
SELECT H_ID, COLUMN_VALUE AS RATINGS FROM Hotel_Ratings,
TABLE(Hotel_Ratings.RATINGS);

```

Autocommit Display 50 Save Run

desc Hotel_Ratings

```

INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('63K9', multivalue_type('GOOD', 'EXCELLENT'));
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('55SK', multivalue_type('VERY GOOD', 'NOT BAD'));
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('26R0', multivalue_type('EXCELLENT'));

SELECT H_ID, COLUMN_VALUE AS RATINGS FROM Hotel_Ratings, TABLE(Hotel_Ratings.RATINGS);

```

Results Explain Describe Saved SQL History

H_ID	RATINGS
63K9	GOOD
63K9	EXCELLENT
26R0	EXCELLENT
55SK	VERY GOOD
55SK	NOT BAD

5 rows returned in 0.01 seconds CSV Export

7.16 Payment:

desc Payment

```

insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, IN-
VOICE.NUMBER, CUS_ID) values('507B', 2000,'10-JAN-23', 'BKASH', 984,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, IN-
VOICE.NUMBER, CUS_ID) values('465C', 500,'11-JUL-23', 'NAGAD', 985,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, IN-
VOICE.NUMBER, CUS_ID) values('203A', 5000,'23-DEC-23', 'CASH', 986,'10')

```

Autocommit Display 50 Save Run

desc Payment

```

insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('507B', 2000,'10-JAN-23', 'BKASH', 984,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('465C', 500,'11-JUL-23', 'NAGAD', 985,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('203A', 5000,'23-DEC-23', 'CASH', 986,'10')

select * from Payment

```

Results Explain Describe Saved SQL History

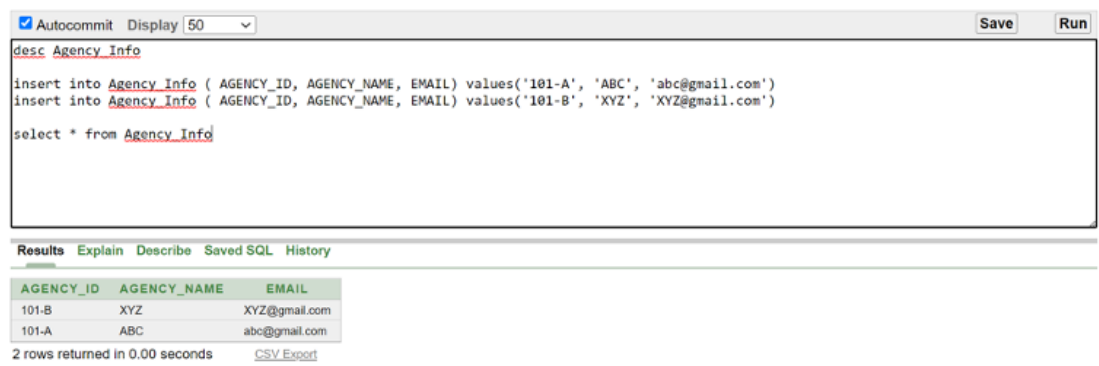
PAY_ID	AMOUNT	PAY_DATE	PAY_METHOD	INVOICE_NUMBER	CUS_ID
507B	2000	10-JAN-23	BKASH	984	10
465C	500	11-JUL-23	NAGAD	985	10
203A	5000	23-DEC-23	CASH	986	10

3 rows returned in 0.00 seconds CSV Export

8 DATA INSERTION:

8.1 Agency_Info

```
desc Agency_Info insert into Agency_Info ( AGENCY_ID, AGENCY_NAME,
EMAIL) values('101-A', 'ABC', 'abc@gmail.com') insert into Agency_Info (
AGENCY_ID, AGENCY_NAME, EMAIL) values('101-B', 'XYZ', 'XYZ@gmail.com')
select * from Agency_Info
```

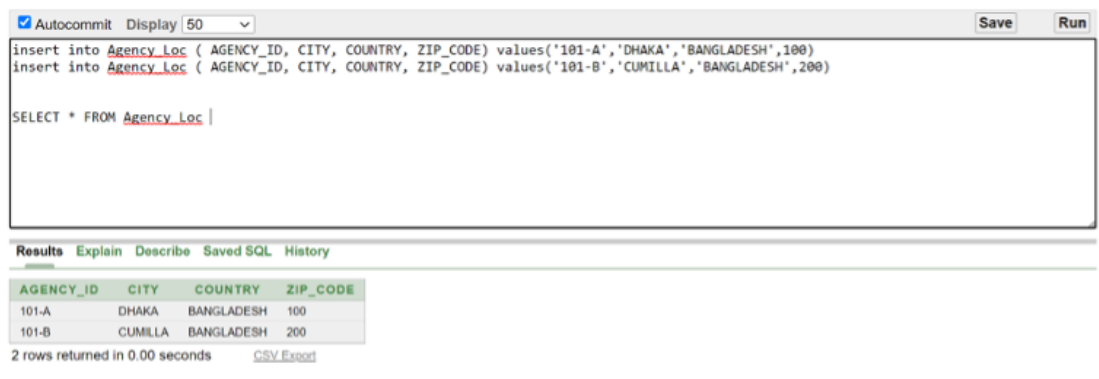


AGENCY_ID	AGENCY_NAME	EMAIL
101-B	XYZ	XYZ@gmail.com
101-A	ABC	abc@gmail.com

2 rows returned in 0.00 seconds [CSV Export](#)

8.2 Agency_Loc

```
desc Agency_Loc insert into Agency_Loc ( AGENCY_ID, CITY, COUNTRY,
ZIP_CODE) values('101-A', 'DHAKA', 'BANGLADESH',100) insert into Agency_Loc
( AGENCY_ID, CITY, COUNTRY, ZIP_CODE) values('101-B', 'CUMILLA',
'BANGLADESH',200) SELECT * FROM Agency_Loc
```



AGENCY_ID	CITY	COUNTRY	ZIP_CODE
101-A	DHAKA	BANGLADESH	100
101-B	CUMILLA	BANGLADESH	200

2 rows returned in 0.00 seconds [CSV Export](#)

8.3 Agency_con

```
desc Agency_con INSERT INTO Agency_Con (Agency_ID, Phone_No) VALUES
('101-A', multivalue_type('017*****', '016*****')); INSERT INTO Agency_Con
```



```

INTO EMP_Skill (E_ID, SKILLS) VALUES ( '2', multivalue_type('ORGANIZATION',
'PROBLEM SOLVING')); INSERT INTO EMP_Skill (E_ID, SKILLS) VAL-
UES ( '3', multivalue_type('LANGUAGE SKILLS')); SELECT E_ID, COL-
UMN_VALUE AS SKILLS FROM EMP_Skill , TABLE(EMP_Skill.SKILLS );

```

Autocommit Display 50 Save Run

desc EMP_Skills

```

INSERT INTO EMP_Skill (E_ID, SKILLS) VALUES ( '1', multivalue_type('COMMUNICATION', 'TIME MANAGEMENT'));
INSERT INTO EMP_Skill (E_ID, SKILLS) VALUES ( '2', multivalue_type('ORGANIZATION', 'PROBLEM SOLVING'));
INSERT INTO EMP_Skill (E_ID, SKILLS) VALUES ( '3', multivalue_type('LANGUAGE SKILLS'));

SELECT E_ID, COLUMN_VALUE AS SKILLS FROM EMP_Skill , TABLE(EMP_Skill.SKILLS );

```

Results Explain Describe Saved SQL History

E_ID	SKILLS
1	COMMUNICATION
1	TIME MANAGEMENT
2	ORGANIZATION
2	PROBLEM SOLVING
3	LANGUAGE SKILLS

5 rows returned in 0.02 seconds [CSV Export](#)

8.6 CUS_Info

```

DESC CUS_Info; insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, MEM-
BERSHIP_LEVEL, AGENCY_ID) values('10', 'ANJUM', 'anjum@gmail.com',
'PREMIUM','101-A') insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, MEM-
BERSHIP_LEVEL, AGENCY_ID) values('20', 'RAISUL', 'raisul@gmail.com',
'PREMIUM','101-A') insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, AGENCY_ID)
values('30', 'SIAM', 'siam@gmail.com','101-A') select * from CUS_Info

```

Autocommit Display 50 Save Run

DESC CUS_Info;

```

insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, MEMBERSHIP_LEVEL, AGENCY_ID) values('10', 'ANJUM', 'anjum@gmail.com', 'PREMIUM','101-A')
insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, MEMBERSHIP_LEVEL, AGENCY_ID) values('20', 'RAISUL', 'raisul@gmail.com', 'PREMIUM','101-A')
insert into CUS_Info( CUS_ID, CUS_NAME, EMAIL, AGENCY_ID) values('30', 'SIAM', 'siam@gmail.com','101-A')

select * from CUS_Info

```

Results Explain Describe Saved SQL History

CUS_ID	CUS_NAME	EMAIL	MEMBERSHIP_LEVEL	AGENCY_ID
10	ANJUM	anjum@gmail.com	PREMIUM	101-A
20	RAISUL	raisul@gmail.com	PREMIUM	101-A
30	SIAM	siam@gmail.com	-	101-A

3 rows returned in 0.00 seconds [CSV Export](#)

8.7 CUS_Loc

```
DESC CUS_Loc; insert into CUS_Loc( CUS_ID, COUNTRY, CITY, ZIP_CODE)
values('10','BANGLADESH', 'MYMENSINGH', '2200') insert into CUS_Loc(
CUS_ID, COUNTRY, CITY, ZIP_CODE) values('20','BANGLADESH', 'CUMILLA',
'3500') insert into CUS_Loc( CUS_ID, COUNTRY, CITY, ZIP_CODE) val-
ues('30','BANGLADESH', 'LAKSHAM', '3570') select * from CUS_Loc
```

```
DESC CUS_Loc;
insert into CUS_Loc( CUS_ID, COUNTRY, CITY, ZIP_CODE) values('10','BANGLADESH', 'MYMENSINGH', '2200')
insert into CUS_Loc( CUS_ID, COUNTRY, CITY, ZIP_CODE) values('20','BANGLADESH', 'CUMILLA', '3500')
insert into CUS_Loc( CUS_ID, COUNTRY, CITY, ZIP_CODE) values('30','BANGLADESH', 'LAKSHAM', '3570')
select * from CUS_Loc
```

CUS_ID	COUNTRY	CITY	ZIP_CODE
10	BANGLADESH	MYMENSINGH	2200
20	BANGLADESH	CUMILLA	3500
30	BANGLADESH	LAKSHAM	3570

3 rows returned in 0.00 seconds

8.8 CUS_Con

```
DESC cus_con; INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '10',
multivalue_type('018*****', '019*****')); INSERT INTO cus_con(CUS_ID,
PHONE_NO) VALUES ( '20', multivalue_type('01811*****', '01822*****'));
INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '30', multivalue_type('018123*****',
'015678*****', '016*****')); SELECT CUS_ID, COLUMN_VALUE AS PHONE_NO
FROM cus_con , TABLE(cus_con.PHONE_NO);
```

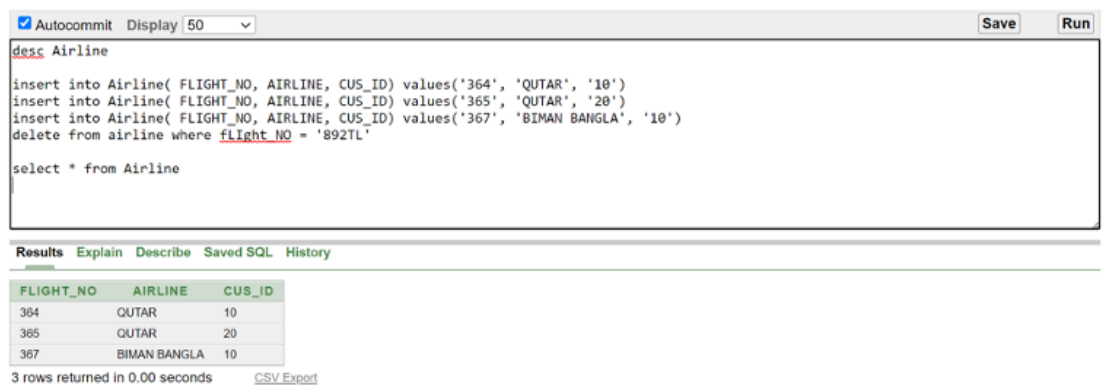
```
DESC cus_con;
INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '10', multivalue_type('018*****', '019*****'));
INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '20', multivalue_type('01811*****', '01822*****'));
INSERT INTO cus_con(CUS_ID, PHONE_NO) VALUES ( '30', multivalue_type('018123*****', '015678*****', '016*****'));
SELECT CUS_ID, COLUMN_VALUE AS PHONE_NO FROM cus_con , TABLE(cus_con.PHONE_NO);
```

CUS_ID	PHONE_NO
10	018*****
10	019*****
20	01811*****
20	01822*****
30	018123*****
30	015678*****
30	016*****

7 rows returned in 0.01 seconds

8.9 Airline

```
desc Airline insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('364',  
'QUTAR', '10') insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('365',  
'QUTAR', '20') insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('367',  
'BIMAN BANGLA', '10') delete from airline where fLIght_NO = '892TL' select  
* from Airline
```



Autocommit Display 50 Save Run

```
desc Airline  
  
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('364', 'QUTAR', '10')  
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('365', 'QUTAR', '20')  
insert into Airline( FLIGHT_NO, AIRLINE, CUS_ID) values('367', 'BIMAN BANGLA', '10')  
delete from airline where fLIght_NO = '892TL'  
  
select * from Airline
```

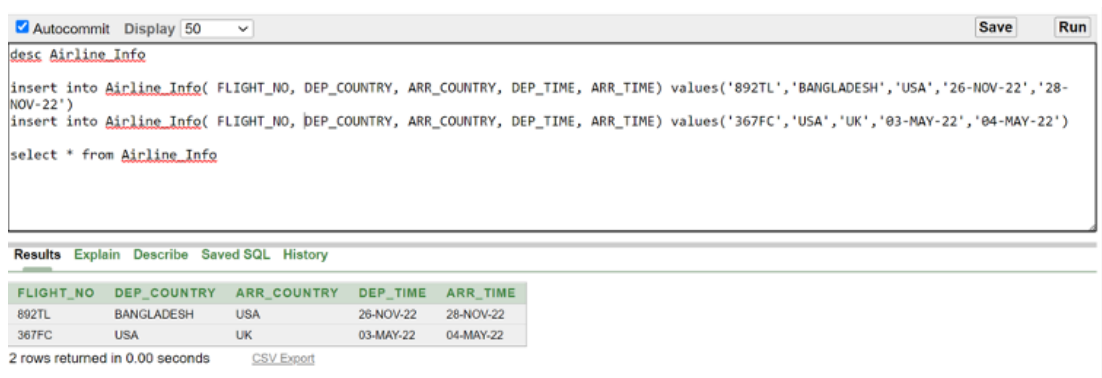
Results Explain Describe Saved SQL History

FLIGHT_NO	AIRLINE	CUS_ID
364	QUTAR	10
365	QUTAR	20
367	BIMAN BANGLA	10

3 rows returned in 0.00 seconds [CSV Export](#)

8.10 Airline_Info

```
desc Airline_Info insert into Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY,  
DEP_TIME, ARR_TIME) values('892TL','BANGLADESH','USA','26-NOV-22','28-  
NOV-22') insert into Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY,  
DEP_TIME, ARR_TIME) values('367FC','USA','UK','03-MAY-22','04-MAY-22')  
select * from Airline_Info
```



Autocommit Display 50 Save Run

```
desc Airline_Info  
  
insert into Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('892TL','BANGLADESH','USA','26-NOV-22','28-  
NOV-22')  
insert into Airline_Info( FLIGHT_NO, DEP_COUNTRY, ARR_COUNTRY, DEP_TIME, ARR_TIME) values('367FC','USA','UK','03-MAY-22','04-MAY-22')  
  
select * from Airline_Info
```

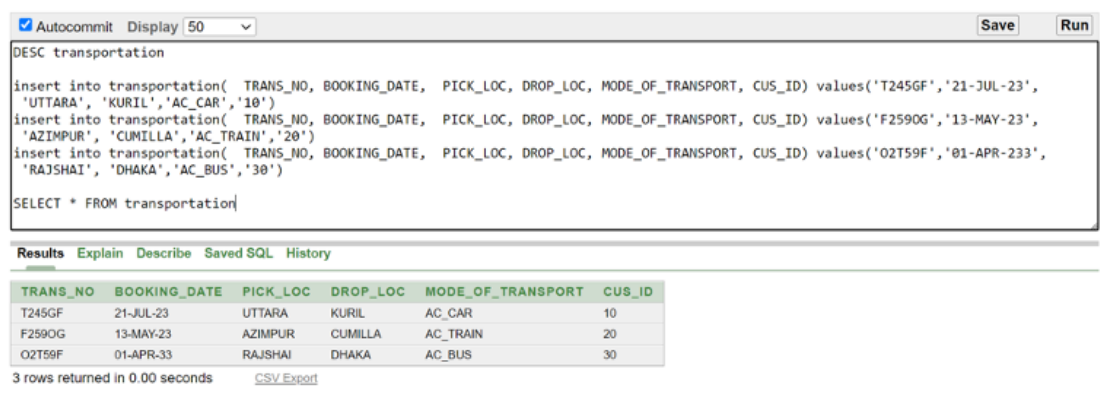
Results Explain Describe Saved SQL History

FLIGHT_NO	DEP_COUNTRY	ARR_COUNTRY	DEP_TIME	ARR_TIME
892TL	BANGLADESH	USA	26-NOV-22	28-NOV-22
367FC	USA	UK	03-MAY-22	04-MAY-22

2 rows returned in 0.00 seconds [CSV Export](#)

8.11 Transportation

DESC transportation insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('T245GF','21-JUL-23', 'UTTARA', 'KURIL', 'AC_CAR', '10') insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('F259OG','13-MAY-23', 'AZIMPUR', 'CUMILLA', 'AC_TRAIN', '20') insert into transportation(TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('O2T59F','01-APR-23', 'RAJSHAI', 'DHAKA', 'AC_BUS', '30') SELECT * FROM transportation



The screenshot shows a database management interface with a SQL editor at the top and a results pane below. The SQL editor contains the following commands:

```
DESC transportation
insert into transportation( TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('T245GF','21-JUL-23', 'UTTARA', 'KURIL', 'AC_CAR', '10')
insert into transportation( TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('F259OG','13-MAY-23', 'AZIMPUR', 'CUMILLA', 'AC_TRAIN', '20')
insert into transportation( TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, CUS_ID) values('O2T59F','01-APR-23', 'RAJSHAI', 'DHAKA', 'AC_BUS', '30')
SELECT * FROM transportation
```

The results pane displays the output of the SELECT statement as a table with 6 columns: TRANS_NO, BOOKING_DATE, PICK_LOC, DROP_LOC, MODE_OF_TRANSPORT, and CUS_ID. It contains 3 rows of data. Below the table, it states "3 rows returned in 0.00 seconds" and provides a "CSV Export" link.

TRANS_NO	BOOKING_DATE	PICK_LOC	DROP_LOC	MODE_OF_TRANSPORT	CUS_ID
T245GF	21-JUL-23	UTTARA	KURIL	AC_CAR	10
F259OG	13-MAY-23	AZIMPUR	CUMILLA	AC_TRAIN	20
O2T59F	01-APR-23	RAJSHAI	DHAKA	AC_BUS	30

8.12 Hotel_Info

DESC Hotel_Info insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('63K9','GHI', 'DHAKA', 'SWIMMING', '10') insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('26R0','BCD', 'CUMILLA', 'PARKING', '20') insert into Hotel_Info (H_ID, H_NAME, LOCATION, FACILITES, CUS_ID) values('55SK','JKL', 'MYMENSINGH', 'CON-FERENCE', '30') SELECT * FROM Hotel_Info

N_O_GUESTS, H_ID) values('3002','03-NOV-23', 'BREAKFAST', '6','55SK')
 SELECT * FROM Food_Booking

Autocommit Display 50 Save Run

desc Food_Booking

```

insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('1001','22-JUL-23', 'LUNCH', '5','63K9')
insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('2030','12-SEP-23', 'DINNER', '3','26R0')
insert into Food_Booking( B_ID, B_DATE, MEAL_TYPE, N_O_GUESTS, H_ID) values('3002','03-NOV-23', 'BREAKFAST', '6','55SK')
SELECT * FROM Food_Booking
  
```

Results Explain Describe Saved SQL History

B_ID	B_DATE	MEAL_TYPE	N_O_GUESTS	H_ID
1001	22-JUL-23	LUNCH	5	63K9
2030	12-SEP-23	DINNER	3	26R0
3002	03-NOV-23	BREAKFAST	6	55SK

3 rows returned in 0.00 seconds [CSV Export](#)

8.15 Hotel_Ratings

desc Hotel_Ratings INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('63K9', multivalue_type('GOOD', 'EXCELLENT')); INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('55SK', multivalue_type('VERY GOOD', 'NOT BAD')); INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('26R0', multivalue_type('EXCELLENT')); SELECT H_ID, COLUMN_VALUE AS RATINGS FROM Hotel_Ratings, TABLE(Hotel_Ratings.RATINGS);

Autocommit Display 50 Save Run

desc Hotel_Ratings

```

INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('63K9', multivalue_type('GOOD', 'EXCELLENT'));
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('55SK', multivalue_type('VERY GOOD', 'NOT BAD'));
INSERT INTO Hotel_Ratings(H_ID, RATINGS) VALUES ('26R0', multivalue_type('EXCELLENT'));

SELECT H_ID, COLUMN_VALUE AS RATINGS FROM Hotel_Ratings, TABLE(Hotel_Ratings.RATINGS);
  
```

Results Explain Describe Saved SQL History

H_ID	RATINGS
63K9	GOOD
63K9	EXCELLENT
26R0	EXCELLENT
55SK	VERY GOOD
55SK	NOT BAD

5 rows returned in 0.01 seconds [CSV Export](#)

8.16 Payment

desc Payment insert into Payment(PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('507B', 2000,'10-JAN-23', 'BKASH', 984,'10')

```

insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, IN-
VOICE_NUMBER, CUS_ID) values('465C', 500,'11-JUL-23', 'NAGAD', 985,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, IN-
VOICE_NUMBER, CUS_ID) values('203A', 5000,'23-DEC-23', 'CASH', 986,'10')
select * from Payment

```

☒ Autocommit
 Display 50
 Save
Run

```

desc Payment

insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('507B', 2000,'10-JAN-23', 'BKASH', 984,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('465C', 500,'11-JUL-23', 'NAGAD', 985,'10')
insert into Payment( PAY_ID, AMOUNT, PAY_DATE, PAY_METHOD, INVOICE_NUMBER, CUS_ID) values('203A', 5000,'23-DEC-23', 'CASH', 986,'10')

select * from Payment

```

Results
[Explain](#)
[Describe](#)
[Saved SQL](#)
[History](#)

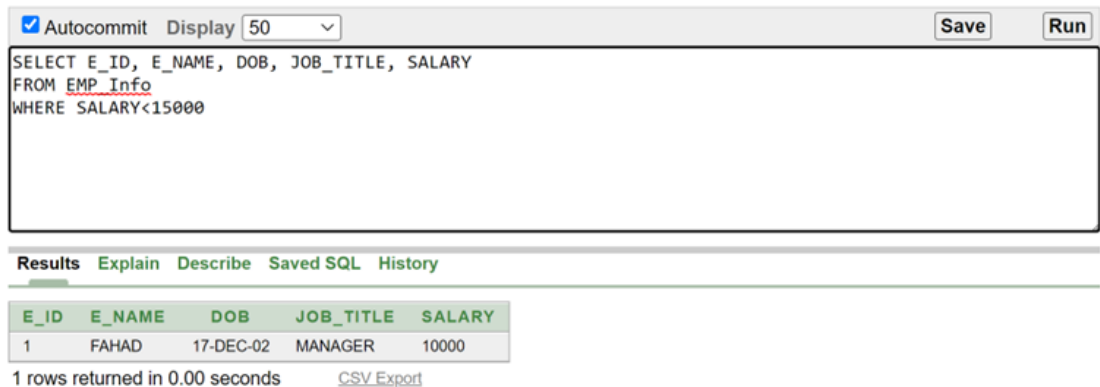
PAY_ID	AMOUNT	PAY_DATE	PAY_METHOD	INVOICE_NUMBER	CUS_ID
507B	2000	10-JAN-23	BKASH	984	10
465C	500	11-JUL-23	NAGAD	985	10
203A	5000	23-DEC-23	CASH	986	10

3 rows returned in 0.00 seconds
[CSV Export](#)

9 QUERY WRITING

9.1 Single Row Query

Q_01: Show employee ID, name, date of birth, job title and salary who earn less than 15000. Query: `SELECT E_ID, E_NAME, DOB, JOB_TITLE, SALARY FROM EMP_Info WHERE SALARY<15000`

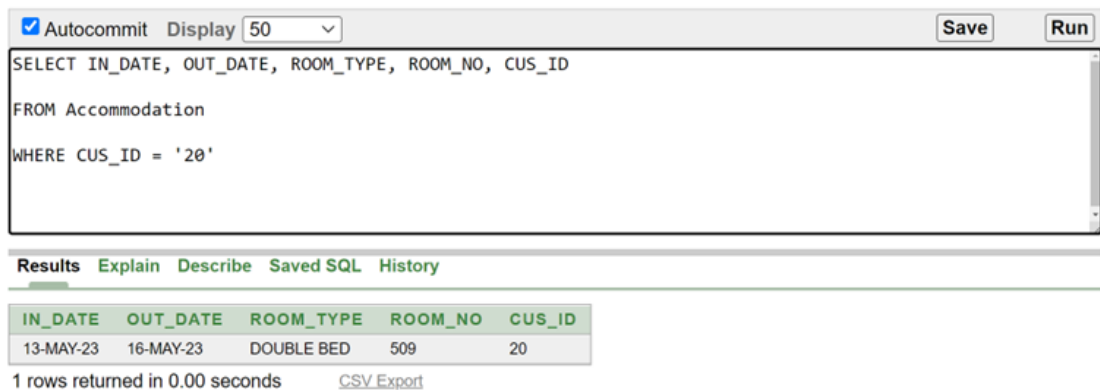


The screenshot shows a SQL query execution window. At the top, there are checkboxes for 'Autocommit' (checked) and a 'Display' dropdown set to '50'. There are 'Save' and 'Run' buttons. The query text is: `SELECT E_ID, E_NAME, DOB, JOB_TITLE, SALARY FROM EMP_Info WHERE SALARY<15000`. Below the query, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, showing a table with 5 columns: E_ID, E_NAME, DOB, JOB_TITLE, and SALARY. The table contains one row: 1, FAHAD, 17-DEC-02, MANAGER, 10000. Below the table, it says '1 rows returned in 0.00 seconds' and there is a 'CSV Export' link.

E_ID	E_NAME	DOB	JOB_TITLE	SALARY
1	FAHAD	17-DEC-02	MANAGER	10000

1 rows returned in 0.00 seconds [CSV Export](#)

Q_02: Show check-in, check-out time, room type and room number of the customer who's ID is 20. Query: `SELECT IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, CUS_ID FROM Accommodation WHERE CUS_ID = '20'`



The screenshot shows a SQL query execution window. At the top, there are checkboxes for 'Autocommit' (checked) and a 'Display' dropdown set to '50'. There are 'Save' and 'Run' buttons. The query text is: `SELECT IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, CUS_ID FROM Accommodation WHERE CUS_ID = '20'`. Below the query, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, showing a table with 5 columns: IN_DATE, OUT_DATE, ROOM_TYPE, ROOM_NO, and CUS_ID. The table contains one row: 13-MAY-23, 16-MAY-23, DOUBLE BED, 509, 20. Below the table, it says '1 rows returned in 0.00 seconds' and there is a 'CSV Export' link.

IN_DATE	OUT_DATE	ROOM_TYPE	ROOM_NO	CUS_ID
13-MAY-23	16-MAY-23	DOUBLE BED	509	20

1 rows returned in 0.00 seconds [CSV Export](#)

Q_03: Show customer phone numbers who's ID is 10. Query: `SELECT CUS_ID, COLUMN_VALUE AS Phone_No FROM CUS_CON, TABLE(CUS_CON.Phone_No) WHERE CUS_ID = '10';`

Autocommit Display 50 Save Run

```
SELECT CUS_ID, COLUMN_VALUE AS Phone_No
FROM CUS_CON, TABLE(CUS_CON.Phone_No)
WHERE CUS_ID = '10';
```

Results Explain Describe Saved SQL History

CUS_ID	PHONE_NO
10	018*****
10	019*****

2 rows returned in 0.00 seconds [CSV Export](#)

9.2 Multiple Row Query

Q₀₄: Show invoice number and payment date where payment is maximum.

Query:

```
SELECT AMOUNT, INVOICE_NUMBER, PAY_DATE FROM
PAYMENT WHERE AMOUNT = (SELECT MAX(AMOUNT) FROM
PAYMENT);
```

Autocommit Display 50 Save Run

```
SELECT AMOUNT, INVOICE_NUMBER, PAY_DATE
FROM PAYMENT
WHERE AMOUNT = (SELECT MAX(AMOUNT) FROM PAYMENT);
```

Results Explain Describe Saved SQL History

AMOUNT	INVOICE_NUMBER	PAY_DATE
5000	986	23-DEC-23

1 rows returned in 0.00 seconds [CSV Export](#)

Q₀₅: Show the average salary based on agency company.

Query:

```
SELECT ROUND(AVG(SALARY),2) AS "AVERAGE SALARY",
AGENCY_ID FROM EMP_Info GROUP BY AGENCY_ID
```

Autocommit Display 50 Save Run

```
SELECT ROUND(AVG(SALARY),2) AS "AVERAGE SALARY", AGENCY_ID
FROM EMP_Info
GROUP BY AGENCY_ID
```

Results Explain Describe Saved SQL History

AVERAGE SALARY	AGENCY_ID
26666.67	101-A

1 rows returned in 0.00 seconds [CSV Export](#)

9.3 SUB QUERY

Q_06: Show all the customer's information from ABC agency. If any customer does not have a membership, display their membership status as no membership.

Query:

```
SELECT CUS_ID, CUS_NAME, EMAIL, NVL(MEMBERSHIP_LEVEL,'NO
MEMBERSHIP'), AGENCY_ID FROM CUS_INFO WHERE AGENCY_ID
= ( SELECT AGENCY_ID FROM AGENCY_INFO WHERE AGENCY_NAME
= 'ABC')
```

Autocommit Display 50 Save Run

```
SELECT CUS_ID, CUS_NAME, EMAIL, NVL(MEMBERSHIP_LEVEL,'NO MEMBERSHIP'), AGENCY_ID
FROM CUS_INFO
WHERE AGENCY_ID = ( SELECT AGENCY_ID
FROM AGENCY_INFO
WHERE AGENCY_NAME = 'ABC')
```

Results Explain Describe Saved SQL History

CUS_ID	CUS_NAME	EMAIL	NVL(MEMBERSHIP_LEVEL,'NOMEMBERSHIP')	AGENCY_ID
10	ANJUM	anjum@gmail.com	PREMIUM	101-A
20	RAISUL	raisul@gmail.com	PREMIUM	101-A
30	SIAM	siam@gmail.com	NO MEMBERSHIP	101-A

3 rows returned in 0.00 seconds [CSV Export](#)

Q_07: Show airline info for Raisul.

Query:

```
SELECT * FROM AIRLINE WHERE CUS_ID = (SELECT CUS_ID
FROM CUS_INFO WHERE CUS_NAME = 'RAISUL')
```

Autocommit Display 50 Save Run

```
SELECT *
FROM AIRLINE
WHERE CUS_ID = (SELECT CUS_ID
                FROM CUS_INFO
                WHERE CUS_NAME = 'RAISUL')
```

Results Explain Describe Saved SQL History

FLIGHT_NO	AIRLINE	CUS_ID
365	QUTAR	20

1 rows returned in 0.01 seconds [CSV Export](#)

9.4 JOINING QUERY

Q_08: Show transportation details for all customer.

Query:

```
SELECT C.CUS_NAME, T.* FROM CUS_Info C, transportation
T WHERE C.CUS_ID = T.CUS_ID
```

Autocommit Display 50 Save Run

```
SELECT C.CUS_NAME, T.*
FROM CUS_Info C, transportation T
WHERE C.CUS_ID = T.CUS_ID
```

Results Explain Describe Saved SQL History

CUS_NAME	TRANS_NO	BOOKING_DATE	PICK_LOC	DROP_LOC	MODE_OF_TRANSPORT	CUS_ID
ANJUM	T245GF	21-JUL-23	UTTARA	KURIL	AC_CAR	10
RAISUL	F259OG	13-MAY-23	AZIMPUR	CUMILLA	AC_TRAIN	20
SIAM	O2T59F	01-APR-33	RAJSHAI	DHAKA	AC_BUS	30

3 rows returned in 0.00 seconds [CSV Export](#)

Q_09: Show all the agency and their information which exists on this database.

Query:

```
select info.*, loc.* from Agency_Info info, Agency_loc loc where
info.agency_id = loc.agency_id
```


Autocommit Display 50 Save Run

```
select info.*, loc.*
from Agency_Info info, Agency_loc loc
where info.agency_id = loc.agency_id
```

Results Explain Describe Saved SQL History

AGENCY_ID	AGENCY_NAME	EMAIL	AGENCY_ID	CITY	COUNTRY	ZIP_CODE
101-B	XYZ	XYZ@gmail.com	101-B	CUMILLA	BANGLADESH	200
101-A	ABC	abc@gmail.com	101-A	DHAKA	BANGLADESH	100

2 rows returned in 0.00 seconds [CSV Export](#)

9.5 VIEW

Q_10: Create a view called transport based on the transportation no., pick-up location and drop location for customer ID 10.

Query:

create view transport as select TRANS_NO,PICK_LOC, DROP_LOC
from transportation where cus_id = '10' DESC TRANSPORT

Autocommit Display 50 Save Run

```
create view transport
as select TRANS_NO,PICK_LOC, DROP_LOC
from transportation
where cus_id = '10'
DESC TRANSPORT
```

Results Explain Describe Saved SQL History

Object Type VIEW Object TRANSPORT

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TRANSPORT	TRANS_NO	Varchar2	20	-	-	-	-	-	-
	PICK_LOC	Varchar2	20	-	-	-	-	-	-
	DROP_LOC	Varchar2	20	-	-	-	-	-	-

1 - 3

Q_11: Create a view named EMP_{SAL} that contains minimum salary and maximum salary according to agency name.

Query:

create view EMP_SAL (NAME, MINSAL, MAXSAL) as select
a.agency_name, min(e.salary), max(e.salary) from emp_info e, agency_info
a where e.agency_id = a.agency_id group by a.agency_name DESC
EMP_SAL

☒ Autocommit
 Display 50
 Save
Run

```

create view EMP_SAL ( NAME, MINSAL, MAXSAL)
as select a.agency_name, min(e.salary), max(e.salary)
from emp_info e, agency_info a
where e.agency_id = a.agency_id
group by a.agency_name

DESC EMP_SAL
    
```

Results
Explain
Describe
Saved SQL
History

Object Type

VIEW

Object

EMP_SAL

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP_SAL	NAME	Varchar2	20	-	-	-	-	-	-
	MINSAL	Number	-	-	-	-	✓	-	-
	MAXSAL	Number	-	-	-	-	✓	-	-
1-3									

10 CONCLUSION

Our project on tourism management system is built so that it can be used as a database schema of any travel management system. We tried to include all types of entities and store their information in the possible easiest way. Through this project, we have put our theoretical knowledge into a practical project that was learnt from this Introduction to Database course. The knowledge we gathered throughout this course and this project hopefully will help to build any projects related to databases we will have in the coming future. We are also very excited to work on this project and willing to add more features as well.