UE20CS301: DATABASE MANAGEMENT SYSTEM

MINI PROJECT

SRN: PES1UG20CS808	NAME: BHAVANA N G	SEC: C

EVENT MANAGEMENT COMPANY DATABASE

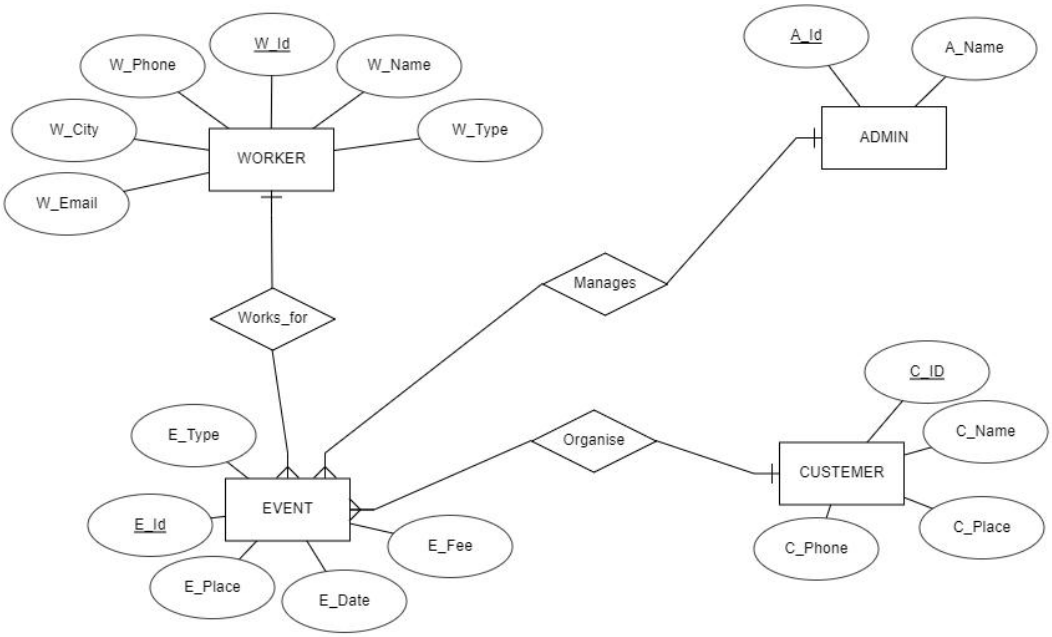
CATALOG

1) SCOPE OF THE PROJECT

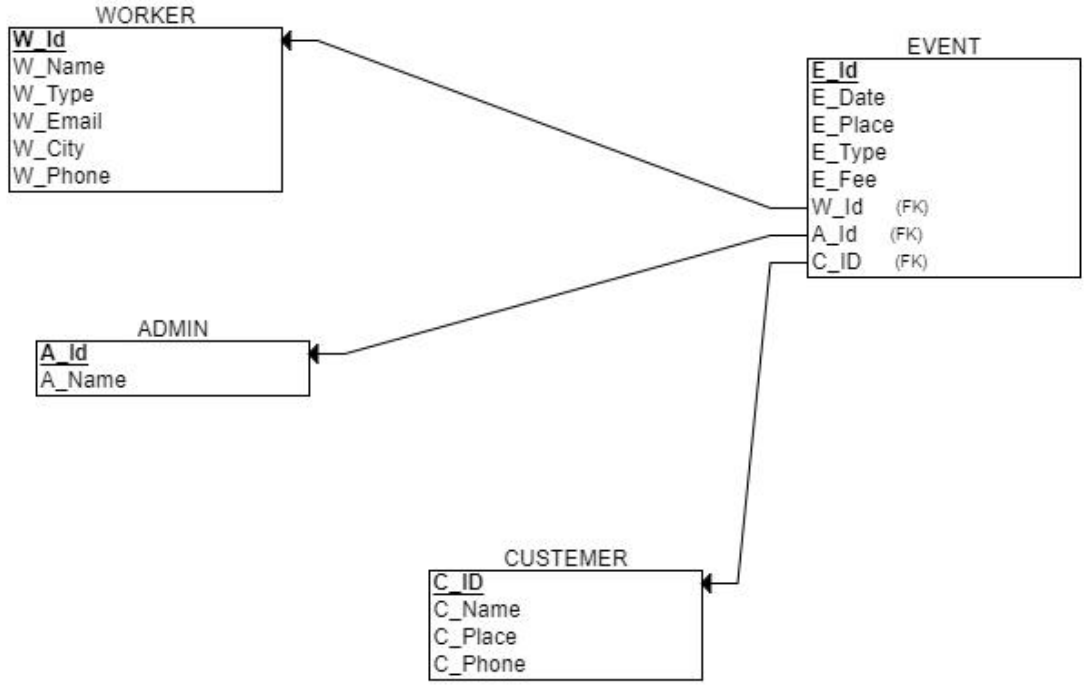
THIS IS AN EVENT MANAGEMENT COMPANY DATABASE WHERE WE MAINTAIN INFORMATION OF THE ALL EVENTS,CUSTOMERS,WORKERS AND WORKERS.

EVENT TABLE CONTAINS EVENT ID, EVENT LOCATION,ADMIN ID WHOSE HANDLING THAT EVENT,WORKER ID WHOSE WORKING ON THAT EVENT.

2) ER DIAGRAM:



3) RELATIONAL SCHEMA DIAGRAM



MYSQL :

4) CREATING DATABASE

#CREATE DATABASE EMS;

```
MariaDB [(none)]> create database ems;
Query OK, 1 row affected (0.003 sec)
```

#USE EMS;

```
MariaDB [(none)]> use ems;
Database changed
```

5) CREATING TABLES

A)WORKER

#CREATE TABLE WORKER
(
  W_ID INT NOT NULL,
  W_NAME VARCHAR(20),
  W_TYPE VARCHAR(20),
  W_EMAIL VARCHAR(20),
  W_CITY VARCHAR(20),
  W_PHONE VARCHAR(10) NOT NULL,
  PRIMARY KEY (W_ID)

);

```
MariaDB [EMS]> CREATE TABLE WORKER
    -> (
    ->   W_Id INT NOT NULL,
    ->   W_Name VARCHAR(20),
    ->   W_Type VARCHAR(20),
    ->   W_Email VARCHAR(20),
    ->   W_City VARCHAR(20),
    ->   W_Phone VARCHAR(10) NOT NULL,
    ->   PRIMARY KEY (W_Id)
    -> );
Query OK, 0 rows affected (0.035 sec)
```

#DESC WORKER;

```
MariaDB [EMS]> desc WORKER;
+---------+------------+------+-----+---------+-------+
| Field   | Type       | Null | Key | Default | Extra |
+---------+------------+------+-----+---------+-------+
| W_Id    | int(11)    | NO   | PRI | NULL    |       |
| W_Name  | varchar(20)| YES  |     | NULL    |       |
| W_Type  | varchar(20)| YES  |     | NULL    |       |
| W_Email | varchar(20)| YES  |     | NULL    |       |
| W_City  | varchar(20)| YES  |     | NULL    |       |
| W_Phone | varchar(10)| NO   |     | NULL    |       |
+---------+------------+------+-----+---------+-------+
6 rows in set (0.019 sec)
```

B)ADMIN

#CREATE TABLE ADMIN
(
A_ID INT NOT NULL,
A_NAME VARCHAR(20),
PRIMARY KEY (A_ID)
);

```
MariaDB [ems]> CREATE TABLE ADMIN
    -> (
    ->   A_Id INT NOT NULL,
    ->   A_Name VARCHAR(20),
    ->   PRIMARY KEY (A_Id)
    -> );
Query OK, 0 rows affected (0.044 sec)
```

#DESC ADMIN;

```
MariaDB [ems]> desc ADMIN;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| A_Id   | int(11)     | NO   | PRI | NULL    |       |
| A_Name | varchar(20) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (0.021 sec)
```

C)CUSTOMER

#CREATE TABLE CUSTOMER

```
(
  C_ID INT NOT NULL,
  C_NAME VARCHAR(20),
  C_PLACE VARCHAR(20),
  C_PHONE VARCHAR(10) NOT NULL,
  PRIMARY KEY (C_ID)
);
```

```
MariaDB [EMS]> CREATE TABLE CUSTOMER
    -> (
    ->   C_ID INT NOT NULL,
    ->   C_Name VARCHAR(20),
    ->   C_Place VARCHAR(20),
    ->   C_Phone VARCHAR(10) NOT NULL,
    ->   PRIMARY KEY (C_ID)
    -> );
Query OK, 0 rows affected (0.039 sec)
```

#DESC CUSTOMER;

```
MariaDB [EMS]> desc customer;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| C_ID    | int(11)     | NO   | PRI | NULL    |       |
| C_Name  | varchar(20) | YES  |     | NULL    |       |
| C_Place | varchar(20) | YES  |     | NULL    |       |
| C_Phone | varchar(10) | NO   |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
4 rows in set (0.019 sec)
```

D)EVENT

```
# CREATE TABLE EVENT
(
  E_ID INT NOT NULL,
  E_DATE DATE,
  E_PLACE VARCHAR(20),
  E_TYPE VARCHAR(20),
  E_FEE INT NOT NULL,
  W_ID INT NOT NULL,
  A_ID INT NOT NULL,
  C_ID INT NOT NULL,
  PRIMARY KEY (E_ID),
  FOREIGN KEY (W_ID) REFERENCES WORKER(W_ID),
  FOREIGN KEY (A_ID) REFERENCES ADMIN(A_ID),
  FOREIGN KEY (C_ID) REFERENCES CUSTOMER(C_ID)
);
```

```
MariaDB [ems]> CREATE TABLE EVENT
    -> (
    ->   E_Id INT NOT NULL,
    ->   E_Date DATE,
    ->   E_Place VARCHAR(20),
    ->   E_Type VARCHAR(20),
    ->   E_Fee INT NOT NULL,
    ->   W_Id INT NOT NULL,
    ->   A_Id INT NOT NULL,
    ->   C_ID INT NOT NULL,
    ->   PRIMARY KEY (E_Id),
    ->   FOREIGN KEY (W_Id) REFERENCES WORKER(W_Id),
    ->   FOREIGN KEY (A_Id) REFERENCES ADMIN(A_Id),
    ->   FOREIGN KEY (C_ID) REFERENCES CUSTOMER(C_ID)
    -> );
Query OK, 0 rows affected (0.042 sec)
```

#DESC EVENT;

```
MariaDB [ems]> desc EVENT;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| E_Id    | int(11)     | NO   | PRI | NULL    |       |
| E_Date  | date        | YES  |     | NULL    |       |
| E_Place | varchar(20) | YES  |     | NULL    |       |
| E_Type  | varchar(20) | YES  |     | NULL    |       |
| E_Fee   | int(11)     | NO   |     | NULL    |       |
| W_Id    | int(11)     | NO   | MUL | NULL    |       |
| A_Id    | int(11)     | NO   | MUL | NULL    |       |
| C_ID    | int(11)     | NO   | MUL | NULL    |       |
+---------+-------------+------+-----+---------+-------+
8 rows in set (0.022 sec)
```

6) POPULATING DATA

A) INSERTING VALUES ONE AFTER ANOTHER

#INSERT INTO ADMIN VALUES(001,"ANU");

#INSERT INTO ADMIN VALUES(002,"ANJALI");

#SELECT * FROM ADMIN;

```
MariaDB [ems]> insert into ADMIN values(001,"ANU");
Query OK, 1 row affected (0.009 sec)

MariaDB [ems]> insert into ADMIN values(002,"ANJALI");
Query OK, 1 row affected (0.003 sec)

MariaDB [ems]> select * from ADMIN;
+------+--------+
| A_Id | A_Name |
+------+--------+
|    1 | ANU    |
|    2 | ANJALI |
+------+--------+
2 rows in set (0.003 sec)
```

B) INSERTING VALUES TO TABLES AT ONCE USING THE COMMAND BELOW

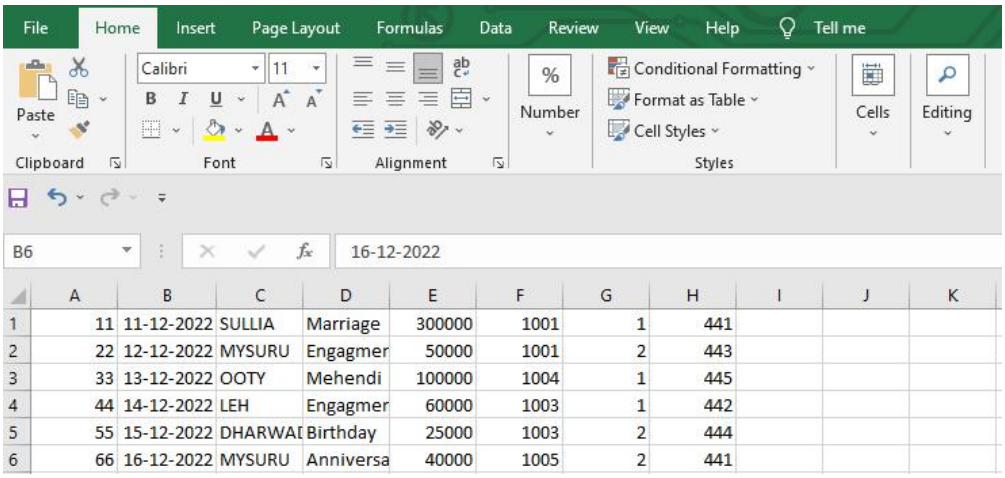
#INSERT INTO CUSTOMER(C_ID,C_NAME,C_PLACE,C_PHONE) VALUES(441,"PRATHIBHA","MYSURU",”7001931823”),(445,"SAMPREETH","BENGALURU",”7117289117”),(442,"ARPITH","UDUPI",”6633662728”),(444,"PREM","KOLAR",”6112881826”),(443,"SNEHA","SULLIA",”7332211100”);

```
MariaDB [ems]> insert into CUSTOMER(C_ID,C_Name,C_Place,C_Phone) values(441,"PRATHIBHA","MYSURU","7001931823"),(445,"SAM
PREETH","BENGALURU","7117289117"),(442,"ARPITH","UDUPI","6633662728"),(444,"PREM","KOLAR","6112881826"),(443,"SNEHA","SU
LLIA","7332211100");
Query OK, 5 rows affected (0.004 sec)
Records: 5  Duplicates: 0  Warnings: 0

MariaDB [ems]> select * from customer;
+------+-----------+-----------+------------+
| C_ID | C_Name    | C_Place   | C_Phone    |
+------+-----------+-----------+------------+
|  441 | PRATHIBHA | MYSURU    | 7001931823 |
|  442 | ARPITH    | UDUPI     | 6633662728 |
|  443 | SNEHA     | SULLIA    | 7332211100 |
|  444 | PREM      | KOLAR     | 6112881826 |
|  445 | SAMPREETH | BENGALURU | 7117289117 |
+------+-----------+-----------+------------+
5 rows in set (0.001 sec)
```


```
MariaDB [ems]> insert into CUSTOMER(C_ID,C_Name,C_Place,C_Phone) values(441,"PRATHIBHA","MYSURU","7001931823"),(445,"SAM
PREETH","BENGALURU","7117289117"),(442,"ARPITH","UDUPI","6633662728"),(444,"PREM","KOLAR","6112881826"),(443,"SNEHA","SU
LLIA","7332211100");
Query OK, 5 rows affected (0.004 sec)
Records: 5  Duplicates: 0  Warnings: 0

MariaDB [ems]> select * from customer;
+------+-----------+-----------+------------+
| C_ID | C_Name    | C_Place   | C_Phone    |
+------+-----------+-----------+------------+
|  441 | PRATHIBHA | MYSURU    | 7001931823 |
|  442 | ARPITH    | UDUPI     | 6633662728 |
|  443 | SNEHA     | SULLIA    | 7332211100 |
|  444 | PREM      | KOLAR     | 6112881826 |
|  445 | SAMPREETH | BENGALURU | 7117289117 |
+------+-----------+-----------+------------+
5 rows in set (0.001 sec)
```


C)INSERTING TABLES USING CSV IMPORTING

>EVENT TABLE



	A	B	C	D	E	F	G	H	I	J	K
1	11	11-12-2022	SULLIA	Marriage	300000	1001	1	441			
2	22	12-12-2022	MYSURU	Engagmer	50000	1001	2	443			
3	33	13-12-2022	OOTY	Mehendi	100000	1004	1	445			
4	44	14-12-2022	LEH	Engagmer	60000	1003	1	442			
5	55	15-12-2022	DHARWAI	Birthday	25000	1003	2	444			
6	66	16-12-2022	MYSURU	Anniversa	40000	1005	2	441			

Import has been successfully finished, 6 queries executed. (EVENT.csv)

1 row inserted. (Query took 0.0015 seconds.)

```
INSERT INTO `event` VALUES ('11', '2022-12-11', 'SULLIA', 'Marriage', '300000', '1001', '1', '441');
```

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0013 seconds.)

```
INSERT INTO `event` VALUES ('22', '2022-12-12', 'MYSURU', 'Engagment', '50000', '1001', '2', '443');
```

[Edit inline] [Edit] [Create PHP code]

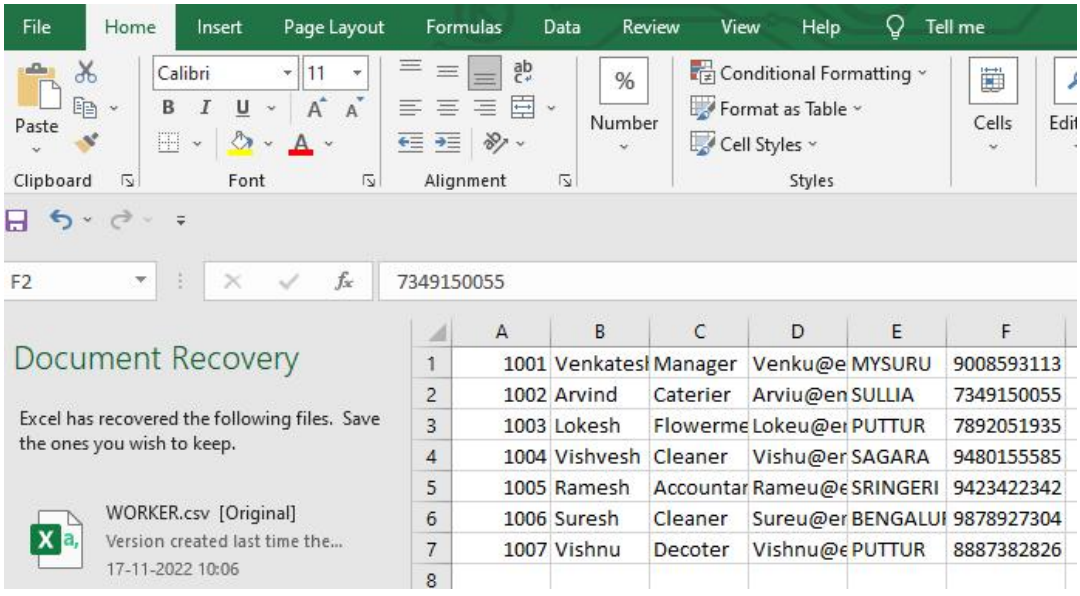
1 row inserted. (Query took 0.0018 seconds.)

```
INSERT INTO `event` VALUES ('33', '2022-12-13', 'OOTY', 'Mehendi', '100000', '1004', '1', '445');
```

[Edit inline] [Edit] [Create PHP code]

```
MariaDB [ems]> select * FROM EVENT;
+------+------------+---------+-------------+--------+------+------+------+
| E_Id | E_Date     | E_Place | E_Type      | E_Fee  | W_Id | A_Id | C_ID |
+------+------------+---------+-------------+--------+------+------+------+
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 | 1001 |    1 |  441 |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 | 1001 |    2 |  443 |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 | 1004 |    1 |  445 |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 | 1003 |    1 |  442 |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 | 1003 |    2 |  444 |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 | 1005 |    2 |  441 |
+------+------------+---------+-------------+--------+------+------+------+
6 rows in set (0.001 sec)
```

>WORKER



```
Import has been successfully finished, 7 queries executed. (WORKER.csv)

1 row inserted. (Query took 0.0016 seconds.)
INSERT INTO `worker` VALUES ('1001', 'Venkatesh', 'Manager', 'Venku@ems', 'MYSURU', '9008593113');
[ Edit inline ] [ Edit ] [ Create PHP code ]

1 row inserted. (Query took 0.0015 seconds.)
INSERT INTO `worker` VALUES ('1002', 'Arvind', 'Caterier', 'Arviu@ems', 'SULLIA', '7349150055');
[ Edit inline ] [ Edit ] [ Create PHP code ]

1 row inserted. (Query took 0.0019 seconds.)
INSERT INTO `worker` VALUES ('1003', 'Lokesh', 'Flowermen', 'Lokeu@ems', 'PUTTUR', '7892051935');
[ Edit inline ] [ Edit ] [ Create PHP code ]

1 row inserted. (Query took 0.0013 seconds.)
INSERT INTO `worker` VALUES ('1004', 'Vishvesh', 'Cleaner', 'Vishu@ems', 'SAGARA', '9480155585');
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

```
MariaDB [ems]> select * from worker;
+------+-----------+------------+------------+-----------+------------+
| W_Id | W_Name    | W_Type     | W_Email    | W_City    | W_Phone    |
+------+-----------+------------+------------+-----------+------------+
| 1001 | Venkatesh | Manager    | Venku@ems  | MYSURU    | 9008593113 |
| 1002 | Arvind    | Caterier   | Arviu@ems  | SULLIA    | 7349150055 |
| 1003 | Lokesh    | Flowermen  | Lokeu@ems  | PUTTUR    | 7892051935 |
| 1004 | Vishvesh  | Cleaner    | Vishu@ems  | SAGARA    | 9480155585 |
| 1005 | Ramesh    | Accountant | Rameu@ems  | SRINGERI  | 9423422342 |
| 1006 | Suresh    | Cleaner    | Sureu@ems  | BENGALURU | 9878927304 |
| 1007 | Vishnu    | Decoter    | Vishnu@ems | PUTTUR    | 8887382826 |
+------+-----------+------------+------------+-----------+------------+
7 rows in set (0.001 sec)
```

QUERIES

7) JOIN OPERATIONS

a) DISPLAY THE DETAILS OF WORKERS WHO WORK UNDER EVENT ID 11(INNER JOIN).

#SELECT WORKER.W_ID,WORKER.W_NAME,WORKER.W_TYPE,WORKER.W_EMAIL,WORKER.W_CITY, WORKER.W_PHONE,EVENT.E_ID FROM WORKER INNER JOIN EVENT ON WORKER.W_ID=EVENT.W_ID AND E_ID=11;

```
MariaDB [ems]> Select WORKER.W_Id,WORKER.W_Name,WORKER.W_Type,WORKER.W_Email,WORKER.W_City,WORKER.W_Phone,EVENT.E_Id fro
m WORKER INNER JOIN EVENT ON WORKER.W_Id=EVENT.W_Id AND E_ID=11;
+------+-----------+---------+----------+--------+------------+------+
| W_Id | W_Name    | W_Type  | W_Email  | W_City | W_Phone    | E_Id |
+------+-----------+---------+----------+--------+------------+------+
| 1001 | Venkatesh | Manager | Venku@ems | MYSURU | 9008593113 |   11 |
+------+-----------+---------+----------+--------+------------+------+
1 row in set (0.001 sec)
```

b) THE FOLLOWING SQL STATEMENT WILL SELECT ALL CUSTOMERS, AND ANY EVENTS THEY MIGHT HAVE(LEFT JOIN)

SELECT CUSTOMER.C_NAME,EVENT.E_ID FROM CUSTOMER LEFT JOIN EVENT ON CUSTOMER.C_ID=EVENT.C_ID;

```
MariaDB [ems]> SELECT CUSTEMER.C_NAME,EVENT.E_Id from CUSTEMER LEFT JOIN  EVENT ON CUSTEMER.C_Id=EVENT.C_Id;
ERROR 1146 (42S02): Table 'ems.custemer' doesn't exist
MariaDB [ems]> SELECT CUSTOMER.C_NAME,EVENT.E_Id from CUSTOMER LEFT JOIN  EVENT ON CUSTOMER.C_Id=EVENT.C_Id;
+-----------+------+
| C_NAME    | E_Id |
+-----------+------+
| PRATHIBHA |   11 |
| PRATHIBHA |   66 |
| ARPITH    |   44 |
| SNEHA     |   22 |
| PREM      |   55 |
| SAMPREETH |   33 |
+-----------+------+
6 rows in set (0.001 sec)
```

c) THE FOLLOWING SQL STATEMENT WILL RETURN ALL WORKER, AND ANY EVENT THEY MIGHT HAVE PLACED(RIGHT JOIN)

SELECT EVENT.E_ID, WORKER.W_NAME FROM EVENT
RIGHT JOIN WORKER ON EVENT.W_ID = WORKER.W_ID;

```
MariaDB [ems]> SELECT EVENT.E_ID, WORKER.W_Name FROM EVENT RIGHT JOIN WORKER ON EVENT.W_Id = WORKER.W_Id;
+------+-----------+
| E_ID | W_Name    |
+------+-----------+
|   11 | Venkatesh |
|   22 | Venkatesh |
| NULL | Arvind    |
|   44 | Lokesh    |
|   55 | Lokesh    |
|   33 | Vishvesh  |
|   66 | Ramesh    |
| NULL | Suresh    |
| NULL | Vishnu    |
+------+-----------+
9 rows in set (0.001 sec)
```

d) THE FOLLOWING SQL QUERY WILL RETURN ALL ADMIN AND EVENTS THEY HANDLE.

#SELECT E.E_ID,E.E_DATE,E.E_PLACE,E.E_TYPE,E.E_FEE,A.A_ID,A.A_NAME FROM EVENT AS E JOIN ADMIN AS A;

```
MariaDB [ems]> SELECT E.E_ID,E.E_Date,E.E_Place,E.E_Type,E.E_Fee,A.A_Id,A.A_Name from EVENT AS E JOIN ADMIN AS A;
+------+------------+---------+-------------+--------+------+--------+
| E_ID | E_Date     | E_Place | E_Type      | E_Fee  | A_Id | A_Name |
+------+------------+---------+-------------+--------+------+--------+
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 |    1 | ANU    |
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 |    2 | ANJALI |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 |    1 | ANU    |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 |    2 | ANJALI |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 |    1 | ANU    |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 |    2 | ANJALI |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 |    1 | ANU    |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 |    2 | ANJALI |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 |    1 | ANU    |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 |    2 | ANJALI |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 |    1 | ANU    |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 |    2 | ANJALI |
+------+------------+---------+-------------+--------+------+--------+
12 rows in set (0.001 sec)
```

e) THE FOLLOWING SQL QUERY WILL RETURN ALL THE EVENT TYPE, WORKER WORKING ON THAT EVENT AND AND WORKER PHONE NUMBER.

#SELECT E_TYPE AS EVENT,W_NAME AS WORKER,W_PHONE AS PHONE_NUMBER FROM WORKER,EVENT WHERE EVENT.W_ID = WORKER.W_ID;

```
MariaDB [ems]> SELECT E_TYPE AS EVENT,W_NAME AS WORKER,W_PHONE AS PHONE_NUMBER FROM WORKER,EVENT WHERE EVENT.W_ID = WORKER.W_ID;
+-------------+-----------+--------------+
| EVENT       | WORKER    | PHONE_NUMBER |
+-------------+-----------+--------------+
| Marriage    | Venkatesh | 9008593113   |
| Engagment   | Venkatesh | 9008593113   |
| Mehendi     | Vishvesh  | 9480155585   |
| Engagment   | Lokesh    | 7892051935   |
| Birthday    | Lokesh    | 7892051935   |
| Anniversary | Ramesh    | 9423422342   |
| Fest        | Venkatesh | 9008593113   |
+-------------+-----------+--------------+
7 rows in set (0.213 sec)
```

f) THE FOLLOWING SQL QUERY WILL RETURN ALL THE CUSTOMER NAME, EVENT TYPE, EVENT DATE, ADMIN AND WORKER WHERE EVENT TYPE IS ENGAGEMENT.

#SELECT E_TYPE AS EVENT,C_NAME AS CUSTOMER,E_DATE ,A_NAME AS ADMIN, W_NAME AS WORKER FROM WORKER,ADMIN,CUSTOMER WHERE EVENT.E_TYPE = "Engagment" AND ADMIN.A_ID=EVENT.E_ID AND EVENT.W_ID=WORKER.W_ID AND CUSTOMER.C_ID=EVENT.C_ID;

```
MariaDB [ems]> SELECT E_TYPE AS EVENT,C_NAME AS CUSTOMER,E_DATE ,A_NAME AS ADMIN, W_NAME AS WORKER FROM WORKER,ADMIN,CUSTOMER,EVENT WHERE EVENT.E_TYPE = "Engagment" AND
 ADMIN.A_ID=EVENT.A_ID AND EVENT.W_ID=WORKER.W_ID AND CUSTOMER.C_ID=EVENT.C_ID;
+-----------+----------+------------+---------+-----------+
| EVENT     | CUSTOMER | E_DATE     | ADMIN   | WORKER    |
+-----------+----------+------------+---------+-----------+
| Engagment | ARPITH   | 2022-11-28 | ANU     | Lokesh    |
| Engagment | SNEHA    | 2022-11-25 | Bhavana | Venkatesh |
+-----------+----------+------------+---------+-----------+
2 rows in set (0.001 sec)
```

8) AGGREGATE FUNCTIONS

A)TO DISPLAY TOTAL NUMBER OF WORKERS IN THE DATABASE(COUNT).

SELECT COUNT(**W_ID**) **FROM** WORKER;

```
MariaDB [ems]> SELECT * FROM WORKER;
+------+-----------+-------------+------------+-----------+------------+
| W_Id | W_Name    | W_Type      | W_Email    | W_City    | W_Phone    |
+------+-----------+-------------+------------+-----------+------------+
| 1001 | Venkatesh | Manager     | Venku@ems  | MYSURU    | 9008593113 |
| 1002 | Arvind    | Caterier    | Arviu@ems  | SULLIA    | 7349150055 |
| 1003 | Lokesh    | Flowermen   | Lokeu@ems  | PUTTUR    | 7892051935 |
| 1004 | Vishvesh  | Cleaner     | Vishu@ems  | SAGARA    | 9480155585 |
| 1005 | Ramesh    | Accountant  | Rameu@ems  | SRINGERI  | 9423422342 |
| 1006 | Suresh    | Cleaner     | Sureu@ems  | BENGALURU | 9878927304 |
| 1007 | Vishnu    | Decoter     | Vishnu@ems | PUTTUR    | 8887382826 |
+------+-----------+-------------+------------+-----------+------------+
7 rows in set (0.001 sec)
```

```
MariaDB [ems]> SELECT COUNT(W_ID) FROM WORKER;
+-------------+
| COUNT(W_ID) |
+-------------+
|           7 |
+-------------+
1 row in set (0.001 sec)
```

B)TO SUM TOTAL FEE FOR ALL EVENTS.

SELECT SUM(E_FEE) **AS** "TOTAL AMOUNT" **FROM** EVENT;

```
MariaDB [ems]> SELECT * FROM EVENT;
+------+------------+---------+-------------+--------+------+------+------+
| E_Id | E_Date     | E_Place | E_Type      | E_Fee  | W_Id | A_Id | C_ID |
+------+------------+---------+-------------+--------+------+------+------+
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 | 1001 |    1 |  441 |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 | 1001 |    2 |  443 |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 | 1004 |    1 |  445 |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 | 1003 |    1 |  442 |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 | 1003 |    2 |  444 |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 | 1005 |    2 |  441 |
+------+------------+---------+-------------+--------+------+------+------+
6 rows in set (0.001 sec)

MariaDB [ems]> SELECT SUM(E_Fee) AS "Total amount" FROM EVENT;
+--------------+
| Total amount |
+--------------+
|       575000 |
+--------------+
1 row in set (0.001 sec)
```

C)TO GET THE MINIMUM FEE IMPOSED ON ANY EVENT.

#**SELECT MIN**(E_FEE) **AS** MINIMUM_FEE_IMPOSED **FROM** EVENT;

```
MariaDB [ems]> SELECT * FROM EVENT;
+------+------------+---------+-------------+--------+------+------+------+
| E_Id | E_Date     | E_Place | E_Type      | E_Fee  | W_Id | A_Id | C_ID |
+------+------------+---------+-------------+--------+------+------+------+
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 | 1001 |    1 |  441 |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 | 1001 |    2 |  443 |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 | 1004 |    1 |  445 |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 | 1003 |    1 |  442 |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 | 1003 |    2 |  444 |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 | 1005 |    2 |  441 |
+------+------------+---------+-------------+--------+------+------+------+
6 rows in set (0.001 sec)
```

```
MariaDB [ems]> SELECT MIN(E_Fee) AS Minimum_Fee_Imposed FROM EVENT;
+---------------------+
| Minimum_Fee_Imposed |
+---------------------+
|               25000 |
+---------------------+
1 row in set (0.001 sec)
```

D) TO GET THE MAXIMUM FEE IMPOSED ON ANY EVENT.

#**SELECT MAX**(E_FEE) **AS** MAX_FEE_IMPOSED **FROM** EVENT;

```
MariaDB [ems]> SELECT * FROM EVENT;
+------+------------+---------+-------------+--------+------+------+------+
| E_Id | E_Date     | E_Place | E_Type      | E_Fee  | W_Id | A_Id | C_ID |
+------+------------+---------+-------------+--------+------+------+------+
|   11 | 2022-12-11 | SULLIA  | Marriage    | 300000 | 1001 |    1 |  441 |
|   22 | 2022-12-12 | MYSURU  | Engagment   |  50000 | 1001 |    2 |  443 |
|   33 | 2022-12-13 | OOTY    | Mehendi     | 100000 | 1004 |    1 |  445 |
|   44 | 2022-12-14 | LEH     | Engagment   |  60000 | 1003 |    1 |  442 |
|   55 | 2022-12-15 | DHARWAD | Birthday    |  25000 | 1003 |    2 |  444 |
|   66 | 2022-12-16 | MYSURU  | Anniversary |  40000 | 1005 |    2 |  441 |
+------+------------+---------+-------------+--------+------+------+------+
```

```
MariaDB [ems]> SELECT MAX(E_Fee) AS Max_Fee_Imposed FROM EVENT;
+-----------------+
| Max_Fee_Imposed |
+-----------------+
|          300000 |
+-----------------+
1 row in set (0.001 sec)
```

9) FUNCTIONS:

THIS FUNCTION RETRIEVES LIST OF DAYS REMAINING FOR THE EVENTS FROM THE CURRENT DATE.

CREATING FUNCTION :

```
DELIMITER $$
CREATE FUNCTION NO_OF_DAYS_REMAINING_FOR_E(EVENT_DATE DATE) RETURNS INT DETERMINISTIC
BEGIN
  DECLARE CUR_DATE DATE;
  SELECT CURRENT_DATE()INTO CUR_DATE;
  RETURN (EVENT_DATE)-(CUR_DATE);
END
$$
DELIMITER ;
```

```
MariaDB [ems]> DELIMITER $$
MariaDB [ems]> CREATE FUNCTION no_of_days_remaining_for_e(event_date date) RETURNS int DETERMINISTIC
    -> BEGIN
    -> DECLARE cur_date DATE;
    -> Select current_date()into cur_date;
    -> RETURN (event_date)-(cur_date);
    -> END
    -> $$
Query OK, 0 rows affected (0.015 sec)

MariaDB [ems]> DELIMITER ;
```

QUERY :

```
SELECT E_ID, E_TYPE AS EVENT, C_NAME AS CUSTOMER,
NO_OF_DAYS_REMAINING_FOR_E(E_DATE) AS 'DAYS LEFT' FROM EVENT JOIN CUSTOMER
WHERE EVENT.C_ID = CUSTOMER.C_ID;
```

```
MariaDB [ems]> SELECT E_ID, E_TYPE AS EVENT, C_NAME AS CUSTOMER, NO_OF_DAYS_REMAINING_FOR_E(E_DATE) AS 'DAYS LEFT'
    -> FROM EVENT JOIN CUSTOMER WHERE EVENT.C_ID = CUSTOMER.C_ID;
+------+-------------+-----------+-----------+
| E_ID | EVENT       | CUSTOMER  | DAYS LEFT |
+------+-------------+-----------+-----------+
|   11 | Marriage    | PRATHIBHA |        11 |
|   22 | Engagment   | SNEHA     |         6 |
|   33 | Mehendi     | SAMPREETH |         8 |
|   44 | Engagment   | ARPITH    |         2 |
|   55 | Birthday    | PREM      |         0 |
|   66 | Anniversary | PRATHIBHA |        10 |
|   78 | Fest        | PRATHIBHA |         3 |
+------+-------------+-----------+-----------+
7 rows in set (0.002 sec)
```

10) TRIGGERS:

a).CREATE A TRIGGER WHICH WILL WORK BEFORE DELETION IN EMPLOYEE TABLE AND CREATE A DUPLICATE COPY OF THE RECORD IN ANOTHER TABLE EVENT_BACKUP.

BEFORE WRITING TRIGGER, WE NEED TO CREATE TABLE EVENT_BACKUP.

```
CREATE TABLE EVENT_BACKUP
(
  E_ID INT NOT NULL,
  E_DATE DATE,
  E_PLACE VARCHAR(20),
  E_TYPE VARCHAR(20),
  E_FEE INT NOT NULL,
  W_ID INT NOT NULL,
  A_ID INT NOT NULL,
  C_ID INT NOT NULL,
  PRIMARY KEY (E_ID),
  FOREIGN KEY (W_ID) REFERENCES WORKER(W_ID),
  FOREIGN KEY (A_ID) REFERENCES ADMIN(A_ID),
  FOREIGN KEY (C_ID) REFERENCES CUSTOMER(C_ID)
);
```

CREATING TRIGGER:

```
DELIMITER $$
CREATE TRIGGER BACKUP BEFORE DELETE ON EVENT
FOR EACH ROW
BEGIN
INSERT INTO EVENT_BACKUP
VALUES
(OLD.E_ID,OLD.E_DATE,OLD.E_PLACE,OLD.E_TYPE,OLD.E_FEE,OLD.W_ID,OLD.A_ID,OLD.C_ID);
END; $$
DELIMITER;
```

```
MariaDB [ems]> delimiter $$
MariaDB [ems]> CREATE TRIGGER Backup BEFORE DELETE ON event
    -> FOR EACH ROW
    -> BEGIN
    -> INSERT INTO event_backup
    -> VALUES (OLD.e_id,OLD.e_date,OLD.e_place,OLD.e_type,OLD.e_fee,OLD.w_id,OLD.a_id,OLD.c_id);
    -> END; $$
Query OK, 0 rows affected (0.015 sec)

MariaDB [ems]> delimiter;
```

Query :

DELETE FROM EVENT WHERE E_ID=11;

```
MariaDB [ems]> DELETE FROM event where e_id=11;
    -> $$
Query OK, 1 row affected (0.013 sec)
```

AFTER EXECUTING THE QUERY:

```
MariaDB [ems]> select *from event;
    -> $$
+------+------------+---------+-------------+--------+------+------+------+
| E_ID | E_DATE     | E_PLACE | E_TYPE      | E_FEE  | W_ID | A_ID | C_ID |
+------+------------+---------+-------------+--------+------+------+------+
|   22 | 2022-11-25 | MYSURU  | ENGAGEMENT  |  50000 | 1001 |    2 |  443 |
|   33 | 2022-11-26 | OOTY    | MEHENDI     | 100000 | 1004 |    1 |  445 |
|   44 | 2022-11-27 | LEH     | ENGAGEMENT  |  60000 | 1003 |    1 |  442 |
|   55 | 2022-11-28 | DHARWAD | BIRTHDAY    |  25000 | 1003 |    2 |  444 |
|   66 | 2022-11-29 | MYSURU  | ANNIVERSARY |  40000 | 1005 |    2 |  441 |
+------+------------+---------+-------------+--------+------+------+------+
5 rows in set (0.001 sec)

MariaDB [ems]> select *from event_backup;
    -> $$
+------+------------+---------+----------+--------+------+------+------+
| E_ID | E_DATE     | E_PLACE | E_TYPE   | E_FEE  | W_ID | A_ID | C_ID |
+------+------------+---------+----------+--------+------+------+------+
|   11 | 2022-11-24 | SULLIA  | MARRIAGE | 300000 | 1001 |    1 |  441 |
+------+------------+---------+----------+--------+------+------+------+
1 row in set (0.001 sec)
```

b).THIS TRIGGER CHECKS ENTERED EVENT DATE WHETHER DATE IS AVAILABLE OR NOT (EVENT DATE SHOULD BE IN THE FUTURE NOT IN THE PAST) WHILE INSERTING INTO THE EVENT TABLE.

CREATING TRIGGER:

DELIMITER $$

CREATE TRIGGER CHECK_DATE BEFORE INSERT ON EVENT

FOR EACH ROW

```
BEGIN
        DECLARE CUR_DATE DATE;
        SELECT CURRENT_DATE()INTO CUR_DATE;
        IF NEW.E_DATE < CUR_DATE THEN
                SIGNAL SQLSTATE '45000'
                SET MESSAGE_TEXT = 'ERROR: EVENT DATE IS NOT AVAILABLE';
        END IF;
END
$$
DELIMITER;
```

```
MariaDB [ems]> delimiter $$
MariaDB [ems]> CREATE TRIGGER  Check_date  BEFORE INSERT ON event
    -> FOR EACH ROW
    -> BEGIN
    -> DECLARE cur_date DATE;
    -> Select current_date()into cur_date;
    -> IF NEW.e_date < cur_date THEN
    -> SIGNAL SQLSTATE '45000'
    -> SET MESSAGE_TEXT = 'ERROR: Event Date Is Not Available';
    -> END IF;
    -> END;
    -> $$
Query OK, 0 rows affected (0.029 sec)
```

QUERY :

INSERT INTO EVENT VALUES(124,"2022-10-20","KOLAR","ENGAGEMENT",25000,123,825,123);

```
MariaDB [ems]> Insert into event values(124,"2022-10-20","Kolar","Engagement",25000,123,825,123);
ERROR 1644 (45000): ERROR: Event Date Is Not Available
```

11) CURSORS AND PROCEDUCES :

THIS CURSOR CUREMAIL IS DECLARED IN PROCEDURE WORKEREMAILLIST THAT PULLS OUT EMAIL ADDRESS OF THE WORKERS FROM THE WORKERS TABLE.

CREATING PROCEDURE AND CURSOR:

DELIMITER $$

CREATE PROCEDURE WORKEREMAILLIST (INOUT EMAILLIST VARCHAR(4000))

```
BEGIN
        DECLARE FINISHED INTEGER DEFAULT 0;
        DECLARE WORKER_EMAIL VARCHAR(100) DEFAULT "";

        DECLARE CUREMAIL CURSOR FOR SELECT W_EMAIL FROM WORKER;
        DECLARE CONTINUE HANDLER FOR NOT FOUND SET FINISHED = 1;

        OPEN CUREMAIL;

        GETEMAIL: LOOP
                FETCH CUREMAIL INTO WORKER_EMAIL;
                IF FINISHED = 1 THEN
                        LEAVE GETEMAIL;
                END IF;

                SET EMAILLIST = CONCAT(WORKER_EMAIL,";",EMAILLIST);
        END LOOP GETEMAIL;
        CLOSE CUREMAIL;
END
$$
DELIMITER ;
```

```
MariaDB [ems]> CREATE OR REPLACE PROCEDURE WorkerEmailList (INOUT emailList varchar(4000))
    -> BEGIN
    -> DECLARE finished INTEGER DEFAULT 0;
    -> DECLARE Worker_Email varchar(100) DEFAULT "";
    -> DEClARE curEmail CURSOR FOR SELECT w_email FROM worker;
    -> DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;
    -> OPEN curEmail;
    -> getEmail: LOOP
    -> FETCH curEmail INTO Worker_Email;
    -> IF finished = 1 THEN
    -> LEAVE getEmail;
    -> END IF;
    -> SET emailList = CONCAT(Worker_Email,";",emailList);
    -> END LOOP getEmail;
    -> CLOSE curEmail;
    -> END
    -> $$
Query OK, 0 rows affected (0.011 sec)
```

QUERY :

SET @WORKERS_EMAIL = "";

CALL WORKEREMAILLIST(@WORKERS_EMAIL);

SELECT @WORKERS_EMAIL;

```
MariaDB [ems]> SET @Workers_Email = "";
Query OK, 0 rows affected (0.000 sec)

MariaDB [ems]> CALL WorkerEmailList(@Workers_Email);
Query OK, 0 rows affected (0.019 sec)

MariaDB [ems]> SELECT @Workers_Email;
+-----------------------------------------------------------------------------+
| @Workers_Email                                                              |
+-----------------------------------------------------------------------------+
| Vishnu@ems;Sureu@ems;Rameu@ems;Vishu@ems;Lokeu@ems;Arviu@ems;Venku@ems;     |
+-----------------------------------------------------------------------------+
1 row in set (0.000 sec)
```

12)VIEW :

CREATING A VIEW THAT PULLS OUT ALL THE WORKER NAME AND EVENT TYPE THEY ARE WORKING ON.

CREATE VIEW WORKEREVENT AS

SELECT W_NAME AS WORKER,E_TYPE AS EVENT

FROM WORKER,EVENT

WHERE WORKER.W_ID=EVENT.W_ID;

SELECT *FROM WORKEREVENT;

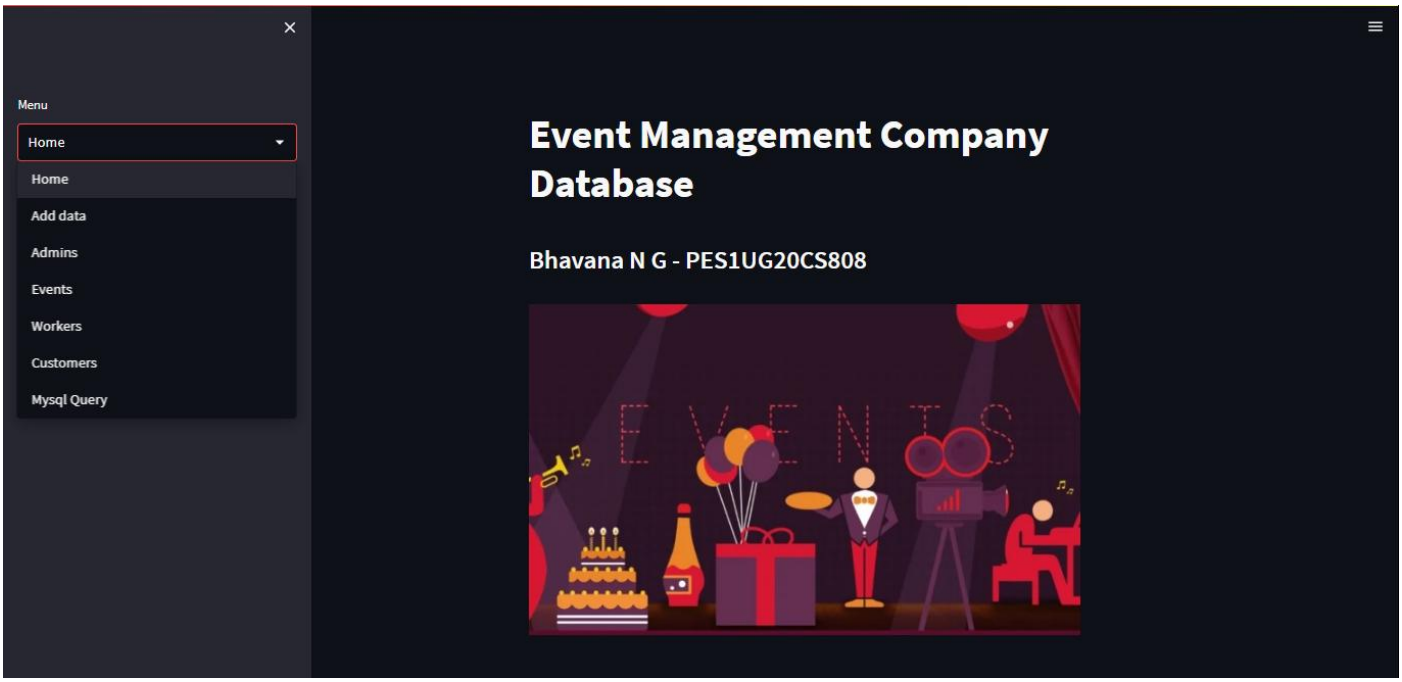
```
MariaDB [ems]> CREATE VIEW WorkerEvent AS
    -> SELECT w_name AS Worker,e_type AS Event
    -> FROM worker,event
    -> WHERE worker.w_id=event.w_id;
Query OK, 0 rows affected (0.014 sec)
```

```
MariaDB [ems]> select *from WorkerEvent;
+-----------+-------------+
| Worker    | Event       |
+-----------+-------------+
| Venkatesh | ENGAGEMENT  |
| Vishvesh  | MEHENDI     |
| Lokesh    | ENGAGEMENT  |
| Lokesh    | BIRTHDAY    |
| Ramesh    | ANNIVERSARY |
+-----------+-------------+
5 rows in set (0.006 sec)
```

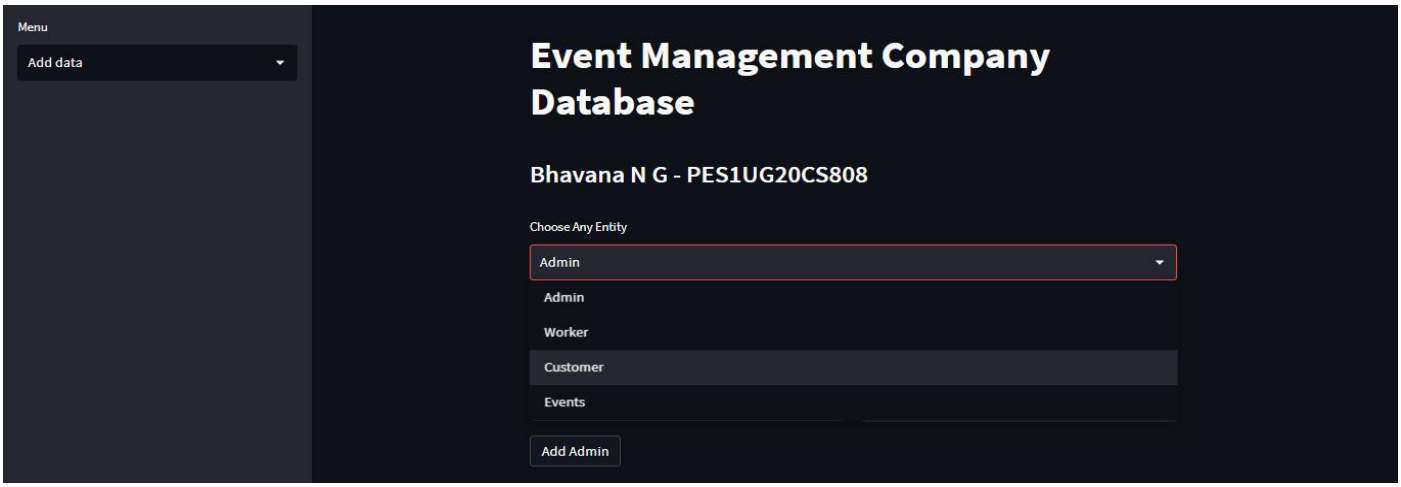
13)USER INTERFACE FOR THE EVENT MANAGEMENT COMPANY DATABASE.

- USED STREAMLIT FOR IMPLEMENTING THE FRONT END

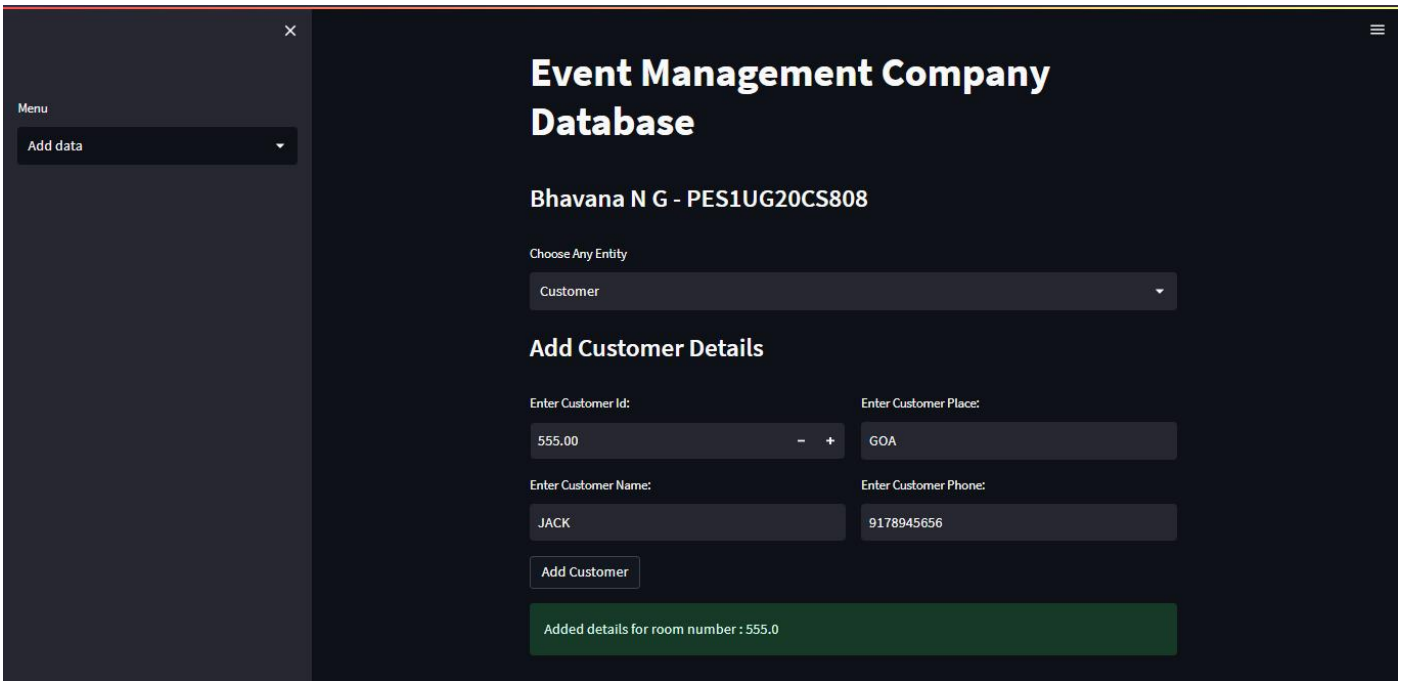
HOME PAGE AND MENU :



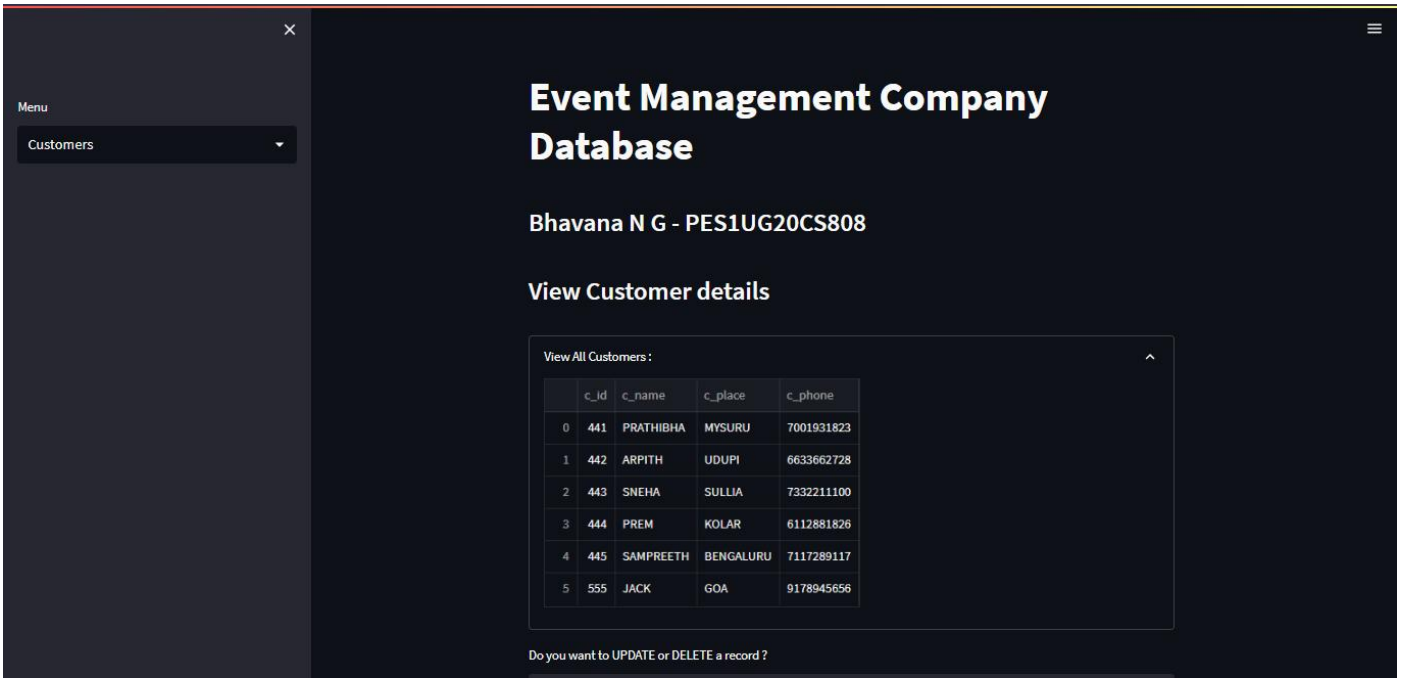
SELECTING TABLE FOR ADDING DATA :



INSERTING INTO CUSTOMER TABLE:

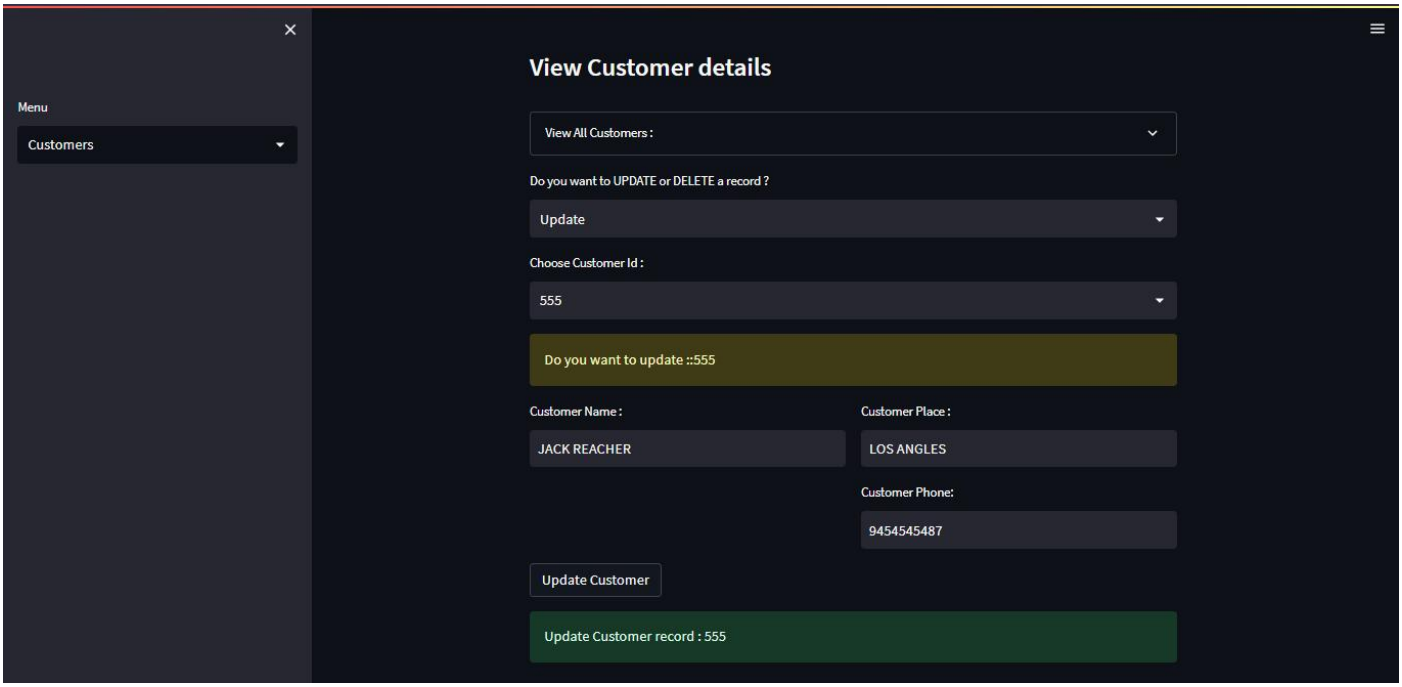


VIEW INSERTED DATA IN THE TABLE:

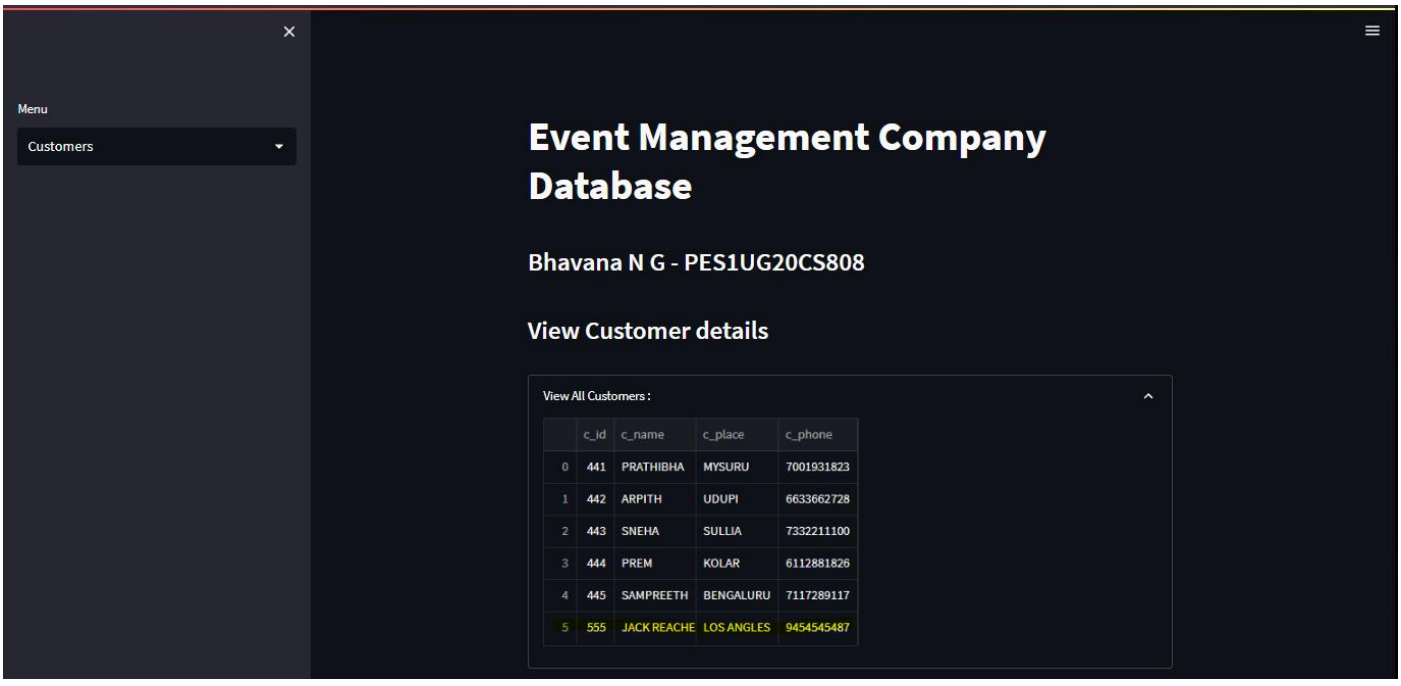


USER CAN SELECT THE OPTION BETWEEN UPDATE AND DELETE IN THE SAME PAGE BELOW THE VIEW TABLE AND THEN USER CAN SELECT ID OF SPECIFIC DATA FROM THE RESPECTIVE TABLE TO PERFORM OPERATION.

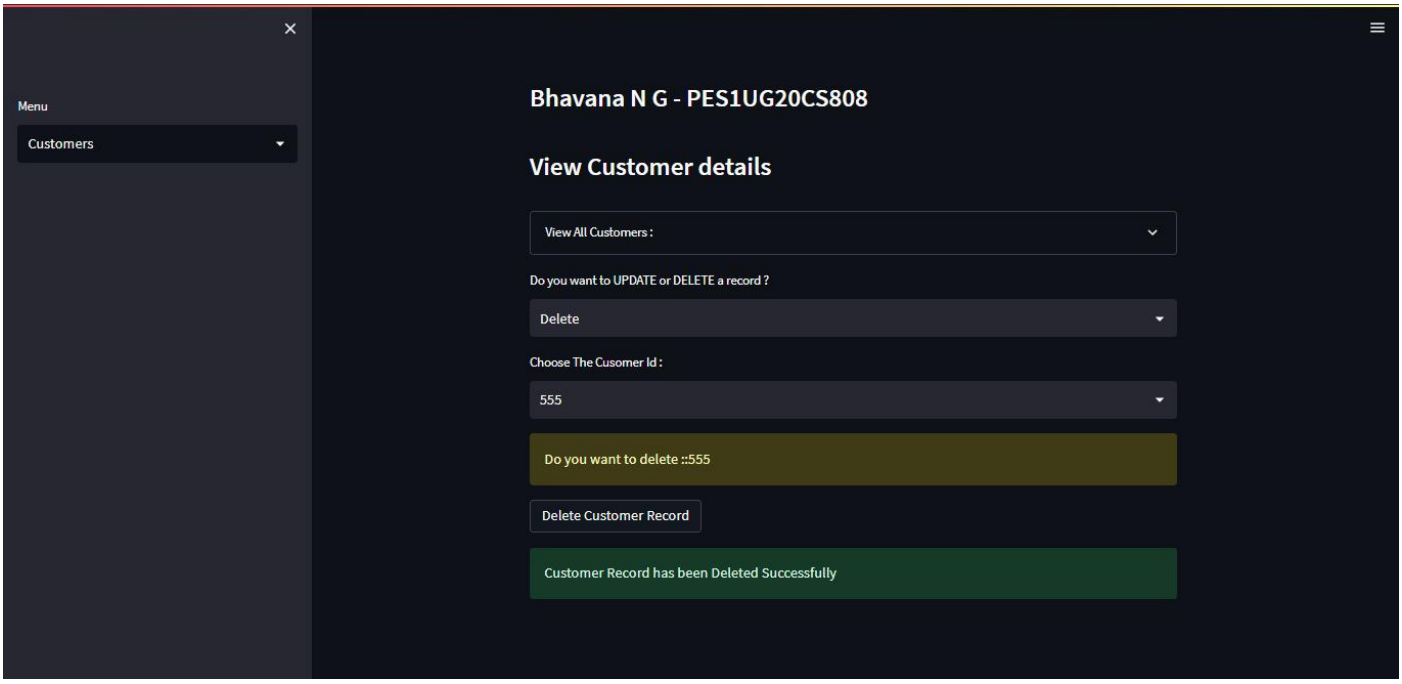
UPDATING THE CUSTOMER DATA OF ID : 555 WHICH WE PREVIOSLY INSERTED :



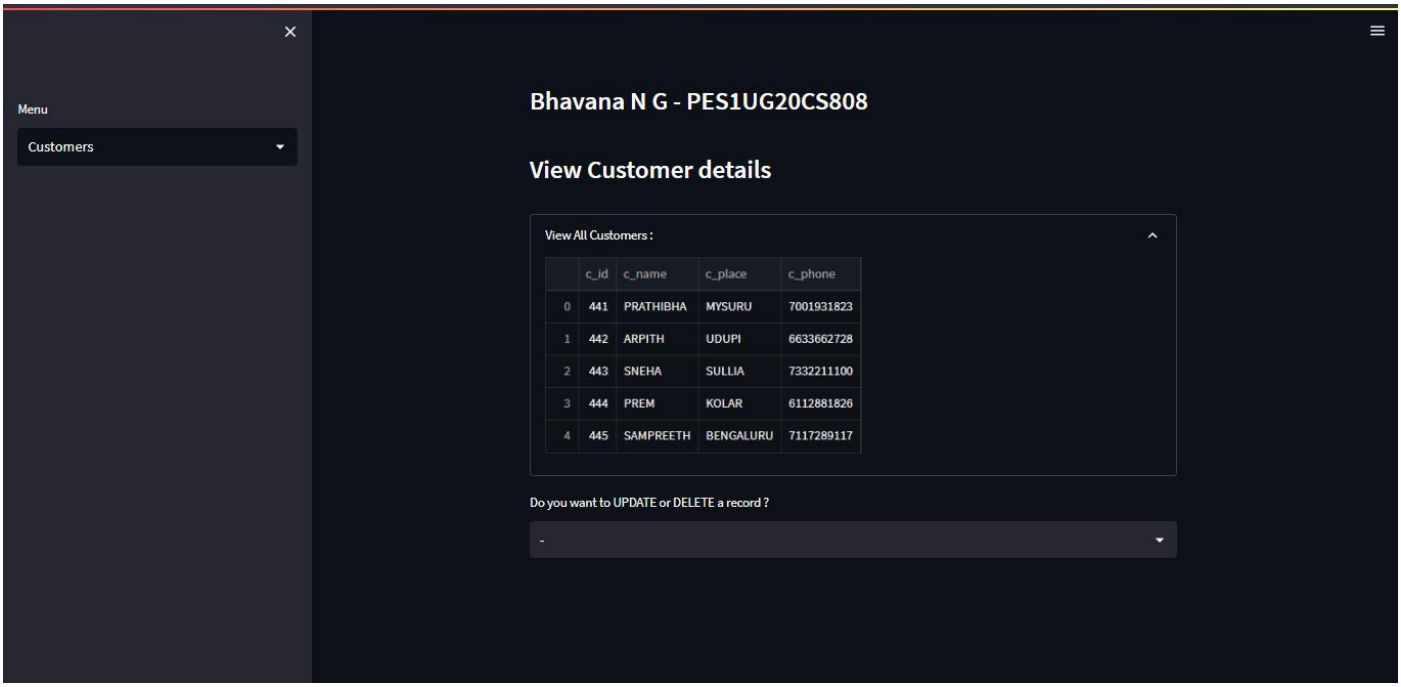
AFTER THE UPDATION:



DELETING THE SAME RECORD OF ID: 555 OF CUSTOMER TABLE WHICH WE RECENTLY UPDATED:



AFTER THE DELETION OF CUSTOMER ID::555 :



RECORD HAS BEEN DELETED.

IN THE MENU THERE IS SPECIFIC WINDOW GIVEN TO EXECUTE ANY CUSTOM SQL QUERIES THAT USER WISHES TO EXECUTE.

EXCUTING SQL QUERY : WHICH WILL LIST OUT EVENT AND CUSTOMER ASSOSIATED WITH THAT EVENT

SELECT E_TYPE AS EVENT,C_NAME AS CUSTOMER FROM EVENT,CUSTOMER WHERE EVENT.C_ID = CUSTOMER.C_ID;