DBMS MINI PROJECT

RESORT MANAGEENT SYSTEM : -

Submitted By :

SATISH G HUDDAR

PES1UG20CS590

V Semester

J-Section

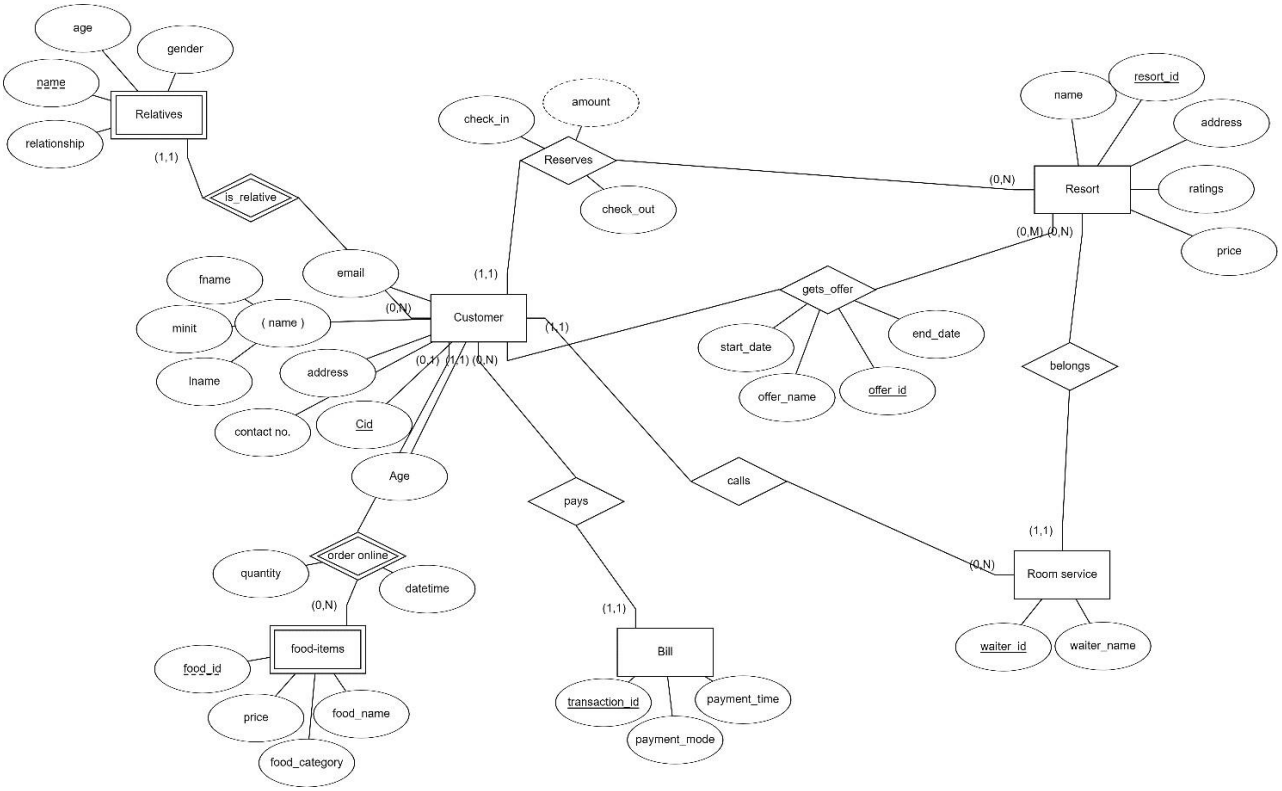
Project Description :

Traditionally, all the information of the customers who have booked a particular resort is stored in books/ledgers. This is very inefficient and labour intensive. This problem can be solved using SQL database, which is stored digitally and information is safe.
The database consists of several entities like customer, resort, offers, food_item, reservation, bill,. Etc which contain all the necessary information.

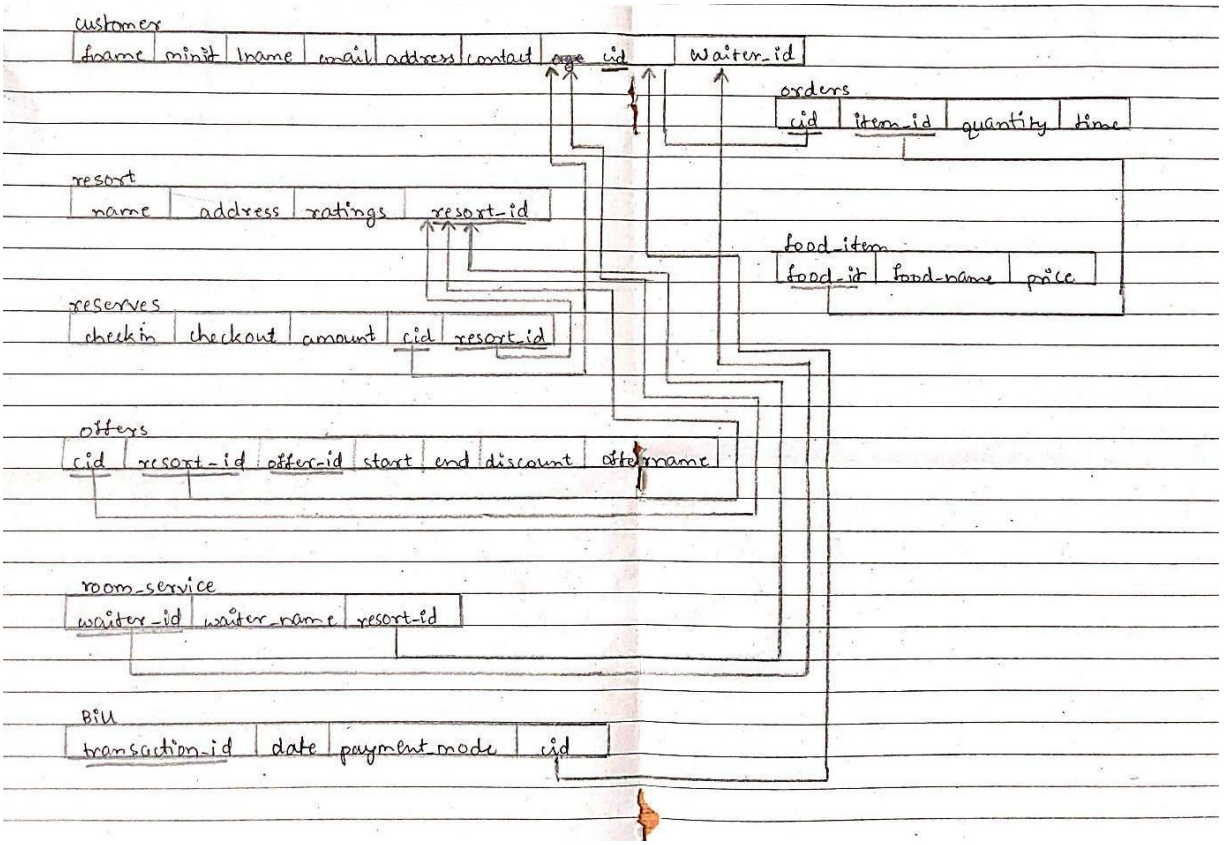
Scope :

To create a website system for managing resort reservations and make it easier for the user to book the resort and buy ticket

Entity Relationship (ER) Diagram :



Relational Schema :



DDL Statements-Building the Database

```
CREATE TABLE resort (
   resort_id DECIMAL(4, 0) PRIMARY KEY CHECK
   (resort_id > 0),resort_name varchar(50) NOT
   NULL,
   address varchar(40) NOT
   NULL,rating
   DECIMAL(3,2),
   price_per_day float
);
CREATE TABLE room_service (
   waiter_id DECIMAL(3, 0) PRIMARY KEY,
   waiter_name
   varchar(20),resort_id
   DECIMAL(4, 0),
   FOREIGN KEY (resort_id) REFERENCES resort(resort_id) ON
   DELETE CASCADE
);
CREATE TABLE customer (
   cid Decimal(4, 0) PRIMARY KEY
   CHECK (cid > 0),fname
   VARCHAR(20),
```

```
   minit
   CHAR(1),
   lname
   VARCHAR(20),
   address
   varchar(30),
   email
   varchar(30),
   contactNo
   DECIMAL(10, 0),
   waiter_id DECIMAL(3, 0),
   FOREIGN KEY (waiter_id) REFERENCES
room_service(waiter_id) ON DELETE CASCADE
);
CREATE TABLE offers
   ( offer_id
   DECIMAL(2, 0),
   offer_name
   varchar(20),
   cid DECIMAL(4, 0) CHECK (cid > 0),
   resort_id DECIMAL(4, 0) CHECK
   (resort_id > 0),discount int,
   startdate
   DATE,
   enddate
   DATE,
   FOREIGN KEY (resort_id) REFERENCES resort(resort_id)
   ON DELETE CASCADE,FOREIGN KEY (cid) REFERENCES
   customer(cid) ON DELETE CASCADE,
   PRIMARY KEY(resort_id, cid, offer_id)
);
CREATE TABLE reservation(
   cid DECIMAL(4, 0) CHECK (cid > 0),
   resort_id DECIMAL(4, 0) CHECK
   (resort_id > 0),checkin DATE NOT
   NULL,
```

```
    checkout DATE NOT
    NULL, amount
    float,
    CHECK (checkout > checkin),
    FOREIGN KEY (cid) REFERENCES customer(cid) ON DELETE
    CASCADE,
    FOREIGN KEY (resort_id) REFERENCES resort(resort_id)
    ON DELETE CASCADE, PRIMARY KEY(resort_id, cid)
);
CREATE TABLE
    relatives( cid
    DECIMAL(4, 0),
    relative_name
    VARCHAR(20), gender
    char(1),
    relationship
    VARCHAR(20),
    FOREIGN KEY (cid) REFERENCES customer(cid) ON
DELETE CASCADE ON UPDATE CASCADE,
    PRIMARY KEY (cid, relative_name)
);
CREATE TABLE
    food_item (
    food_id
    decimal(2, 0),
    food_name
    varchar(20),
    price numeric
    CHECK (
        price BETWEEN 0.00 AND 500.00
```

```
   ),
   PRIMARY KEY(food_id)
);
 = customer orders food via some food
delivery appCREATE TABLE orders (
   cid DECIMAL(4, 0),
   item_id
   DECIMAL(3, 0),
   quantity INT,
   time DATETIME,
   FOREIGN KEY(cid) REFERENCES customer(cid) ON DELETE
   RESTRICT,
   FOREIGN KEY(item_id) REFERENCES food_item(food_id)
   ON DELETE RESTRICT,PRIMARY KEY (cid, item_id)
);
 = number of digits in transaction Id varies, 12
is most commonCREATE TABLE bill (
   transaction_id DECIMAL(12, 0)
   PRIMARY KEY,date DATE,
   cid DECIMAL(4, 0),
   paymentmode varchar(10),
   FOREIGN KEY(cid) REFERENCES customer(cid) ON DELETE RESTRICT
);
```

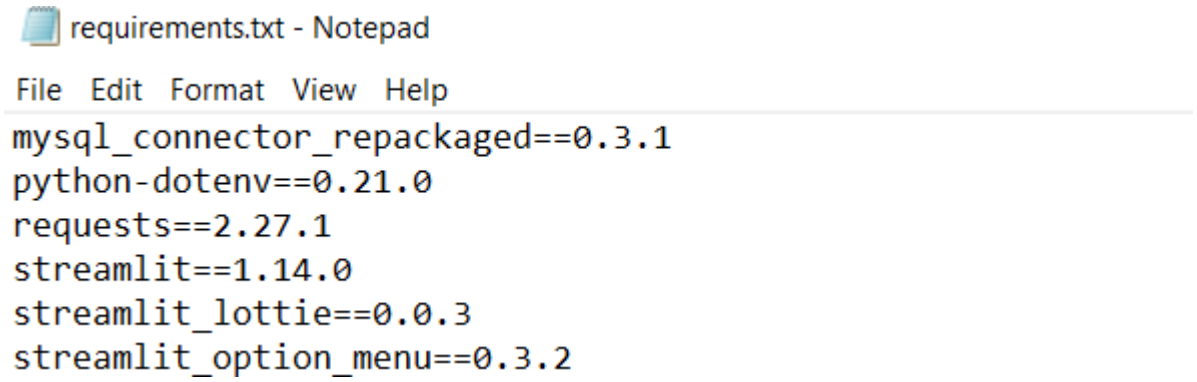
DML Statements-Populating the Database

```
insert into food_item Values('01','thaali','400');
insert into food_item Values('02','pizza','250');
insert into food_item Values('03','ghee rice','120');
insert into food_item Values('04','schezwan fried rice','120');
insert into food_item Values('05','chicken biryani','180');
insert into food_item Values('06','mutton thaali','320');
insert into food_item Values('07','surma fish','240');


insert into offers Values('01','winter vaction', '1001','1006','20','2022-12-20','2022-12-
30');
insert into offers Values('02','diwali offer', '1005','1001','15','2022-10-01','2022-12-
31');
insert into offers Values('03','special offer', '1007','1007','25','2022-11-01','2022-12-
31');


insert into resort Values ('1001',"The Dukes Retreat",'Lonavala',5,1999);
insert into resort Values ('1002',"Ferreira Resort",'Lonavala',4,1799);
insert into resort Values ('1003',"Villa San Lorentz",'Lonavala',5,1699);
insert into resort Values ('1004',"Misty Meadows",'Lonavala',4,1499);
insert into resort Values ('1005',"Sunshine Resort",'Lonavala',5,1699);
insert into resort Values ('1006',"Dandeli Jungle Resort",'Dandeli',5,1499);
insert into resort Values ('1007',"Wild Planet Jungle Resort",'Dandeli',5,1299);
insert into resort Values ('1008',"Swast-Mast Resort",'Lonavala',4,1399);
insert into resort Values ('1009',"Alurkar Resort",'Belgaum',5,1499);
insert into resort Values ('1010',"Gavkari",'Belgaum',5,999);


insert into room_service Values('101','chotu','1001');
insert into room_service Values('102','bhola','1002');
insert into room_service Values('103','brijesh','1003');
insert into room_service Values('104','ajay','1004');
insert into room_service Values('105','shukh','1005');
insert into room_service Values('106','vikalp','1006');
insert into room_service Values('107','alam','1007');
insert into room_service Values('108','suresh','1008');
insert into room_service Values('109','keshav','1006');
insert into room_service Values('110','sharad','1009');
insert into room_service Values('111','munna','1010');

insert into customer Values('1001','narendra','','modi','gujarat' ,'modi@gmail.com'
,'1234567890','106');
insert into customer Values('1002','amit','','shah','gujarat' ,'shah@gmail.com'
,'1234567890','102');
```

```
insert into customer Values('1003','atal','b','vajpayee','bihar' ,'vajpayee@gmail.com'
,'1234567890','103');
insert into customer Values('1004','abdul','','kalam','patna' ,'aniket@gmail.com'
,'1234567890','104');
insert into customer Values('1005','yogi','','adityanath','uttar pradesh' ,'yogi@gmail.com'
,'1234567890','102');
insert into customer Values('1006','balasaheb','','thakre','mumbai' ,'thakre@gmail.com'
,'1234567890','101');
insert into customer Values('1007','basavraj','','bommai','karnataka' ,'bommai@gmail.com'
,'1234567890','102');


insert into bill Values('250707244234','2022-11-21','1001','upi');
insert into bill Values('202622637838','2022-11-22','1002','debit card');
insert into bill Values('644161913172','2022-11-15','1003','credit card');
insert into bill Values('988602103725','2022-11-21','1004','upi');
insert into bill Values('859741505883','2022-11-21','1005','cash');
insert into bill Values('346488919858','2022-11-26','1006','credit card');


insert into orders Values('1001','1','3','2022-11-21  10:55:05');
insert into orders Values('1002','4','4','2022-11-21  13:56:55');
insert into orders Values('1003','5','2','2022-11-10  12:35:05');
insert into orders Values('1004','7','8','2022-11-20  21:55:05');
insert into orders Values('1002','3','1','2022-11-18  07:24:05');
insert into orders Values('1006','4','4','2022-11-25  09:18:05');


insert into relatives Values('1002','bhaskar bhat','m','friend');
insert into relatives Values('1003','vijay verma','m','friend');
insert into relatives Values('1003','sujay patil','m','friend');
insert into relatives Values('1004','sharad shukla','m','friend');
insert into relatives Values('1004','anil desai','m','son');
insert into relatives Values('1005','manish gupta','m','colleague');
insert into relatives Values('1005','kartik singh','m','colleague');
insert into relatives Values('1005','shilpa trivedi','f','colleague');
insert into relatives Values('1006','manthan patil','m','friend');


insert into reservation values('1001','1001','2022-11-11','2022-11-21',15592);
insert into reservation values('1002','1002','2022-11-15','2022-11-22',12593);
insert into reservation values('1003','1003','2022-11-15','2022-11-227',11893);
insert into reservation values('1004','1004','2022-11-16','2022-11-21',7495);
insert into reservation values('1005','1005','2022-11-18','2022-11-21',4077.6);
insert into reservation values('1006','1006','2022-11-20','2022-11-26',8994);
```

Tools Used

- UI for database operations – streamlit
- Database connection – mysql-connector-python
- Xampp



QUERIES

JOIN QUERIES

1. **Display first name of customer and name waiter assigned to him**

select fname, waiter_name from customer as c join room_service as w on c.waiter_id = w.waiter_id;

```
MariaDB [resort_590]> select fname, waiter_name from customer as
 c join room_service as w on c.waiter_id = w.waiter_id;
+-----------+-------------+
| fname     | waiter_name |
+-----------+-------------+
| narendra  | vikalp      |
| amit      | bhola       |
| atal      | brijesh     |
| abdul     | ajay        |
| yogi      | bhola       |
| balasaheb | chotu       |
| basavraj  | bhola       |
+-----------+-------------+
7 rows in set (0.001 sec)

MariaDB [resort_590]>
```

2. **Display food_name, price and quantity of all food orders made by customer with cid=1001**

```
MariaDB [resort_590]>
MariaDB [resort_590]> select f.food_name, f.price, o.quantity fr
om food_item f join orders o on f.food_id=o.item_id where o.cid=
1001;
+-----------+-------+----------+
| food_name | price | quantity |
+-----------+-------+----------+
| thaali    |   400 |        3 |
+-----------+-------+----------+
1 row in set (0.068 sec)

MariaDB [resort_590]>
```

3. **Display resort_id, resort name and address of all those resorts where the number of bookings is greater than 1.**

select r.resort_id, r.resort_name,r.address from resort r where r.resort_id in (
select re.resort_id from reservation re group by re.resort_id having count(*) > 1
);

```
MariaDB [resort_590]> select r.resort_id, r.resort_name,r.addres
s from resort r where r.resort_id in (
    -> select re.resort_id from reservation re group by re.resor
t_id having count(*) > 1
    -> );
+-----------+-----------------+----------+
| resort_id | resort_name     | address  |
+-----------+-----------------+----------+
|      1001 | The Dukes Retreat | Lonavala |
|      1002 | Ferreira Resort   | Lonavala |
+-----------+-----------------+----------+
2 rows in set (0.001 sec)
```

4. **Display information of resort which are available for booking**

   select r.resort_id, r.resort_name, r.address from resort r where not exists (select * from reservation re where re.resort_id=r.resort_id);

```
MariaDB [resort_590]>
MariaDB [resort_590]> select r.resort_id, r.resort_name, r.addre
ss from resort r where not exists (select * from reservation re
where re.resort_id=r.resort_id);
+-----------+--------------------------+----------+
| resort_id | resort_name              | address  |
+-----------+--------------------------+----------+
|      1003 | Villa San Lorentz        | Lonavala |
|      1007 | Wild Planet Jungle Resort | Dandeli  |
|      1008 | Swast-Mast Resort        | Lonavala |
|      1009 | Alurkar Resort           | Belgaum  |
|      1010 | Gavkari                  | Belgaum  |
|      1234 | sat30                    | Dandeli  |
+-----------+--------------------------+----------+
6 rows in set (0.004 sec)
```

AGGREGATE FUNCTIONS

1. **Display total bookings for each resort**

   select r.resort_id, r.resort_name, count(*) as total_bookings from resort r join reservation rs on r.resort_id=rs.resort_id group by resort_id;

```
MariaDB [resort_590]> select r.resort_id, r.resort_name, count(*
) as total_bookings from resort r join reservation rs on r.resor
t_id=rs.resort_id group by resort_id;
+-----------+----------------------+----------------+
| resort_id | resort_name          | total_bookings |
+-----------+----------------------+----------------+
|      1001 | The Dukes Retreat    |              2 |
|      1002 | Ferreira Resort      |              2 |
|      1004 | Misty Meadows        |              1 |
|      1005 | Sunshine Resort      |              1 |
|      1006 | Dandeli Jungle Resort |              1 |
|      5465 | sat22                |              1 |
+-----------+----------------------+----------------+
6 rows in set (0.001 sec)

MariaDB [resort_590]>
```

2. **List all resorts city wise**

   select r.address as location, count(*) total_resorts from resort r group by r.address;

```
MariaDB [resort_590]>
MariaDB [resort_590]> select r.address as location, count(*) tot
al_resorts from resort r group by r.address;
+----------+---------------+
| location | total_resorts |
+----------+---------------+
| Belgaum  |             2 |
| Dandeli  |             4 |
| Lonavala |             6 |
+----------+---------------+
3 rows in set (0.002 sec)
```

3. To find number of payments done through different modes

select paymentmode,count(*) from bill GROUP BY paymentmode;

```
MariaDB [resort_590]>
MariaDB [resort_590]> select paymentmode,count(*) from bill GROUP BY paymentmode;
+-------------+----------+
| paymentmode | count(*) |
+-------------+----------+
| cash        |        1 |
| credit car  |        2 |
| debit card  |        1 |
| upi         |        2 |
+-------------+----------+
4 rows in set (0.001 sec)

MariaDB [resort_590]>
```

4. Number of resorts Rating wise

select rating,count(*) from resort GROUP BY rating;

```
MariaDB [resort_590]> select rating,count(*) from resort GROUP BY rating;
+--------+----------+
| rating | count(*) |
+--------+----------+
|   2.35 |        1 |
|   4.00 |        3 |
|   4.30 |        1 |
|   4.50 |        2 |
|   5.00 |        7 |
+--------+----------+
5 rows in set (0.001 sec)
```

SET OPERATIONS

1. **List all customers who have booked a resort with “Diwali offer” in Lonavala.**

```
(select re.cid from reservation re natural join resort r where address="lonavala")
Union
(select c.cid from customer c natural join offers o where offer_name="diwali offer");
```

```
MariaDB [resort_590]>
MariaDB [resort_590]> (select re.cid from reservation re natural
 join resort r where address="lonavala")
    -> Union
    -> (select c.cid from customer c natural join offers o where
 offer_name="diwali offer");
+------+
| cid  |
+------+
| 1001 |
| 1002 |
| 1004 |
| 1005 |
| 3310 |
| 9510 |
+------+
6 rows in set (0.056 sec)

MariaDB [resort_590]>
```

2. **List all customers who have NOT paid the bill but have ordered food item**

```
MariaDB [resort_590]> (select c.cid,c.fname,c.lname from custome
r c where c.cid not in (select b.cid from bill b))
    -> intersect
    -> (select c.cid, c.fname, c.lname from customer c where c.c
id in (select o.cid from orders o));
+------+-------+-------+
| cid  | fname | lname |
+------+-------+-------+
| 9510 | aqaz  | aqz   |
+------+-------+-------+
1 row in set (0.001 sec)

MariaDB [resort_590]>
```

3. List Customer who not got any offer

Select cid,fname from customer except select cid from offers;

```
MariaDB [resort_590]> Select cid from customer except select cid from offers;
+------+
| cid  |
+------+
| 1804 |
| 2222 |
| 3310 |
| 3551 |
| 4444 |
| 6800 |
| 9510 |
| 1006 |
| 1002 |
| 1003 |
| 1004 |
+------+
```

4. To find resort which not have reserved

Select resort_id from resort EXCEPT Select resort_id from reservation;

```
MariaDB [resort_590]> Select resort_id from resort EXCEPT Select resort_id from reservation;
+-----------+
| resort_id |
+-----------+
|      1008 |
|      1009 |
|      1010 |
|      1234 |
|      4952 |
+-----------+
5 rows in set (0.001 sec)
```

FUNCTIONS

1. To convert rating to feedback

```
DELIMITER $$
CREATE FUNCTION ratef(rate decimal(1))
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
IF rate < 3 THEN
RETURN "Average";

ELSE
RETURN "Very Good";
END IF;
END$$
DELIMITER ;
```

```
MariaDB [resort_590]> select resort_name, ratef(rating) as feedback from resort;
+--------------------------+-----------+
| resort_name              | feedback  |
+--------------------------+-----------+
| The Dukes Retreat        | Very Good |
| Ferreira Resort          | Very Good |
| Villa San Lorentz        | Very Good |
| Misty Meadows            | Very Good |
| Sunshine Resort          | Very Good |
| Dandeli Jungle Resort    | Very Good |
| Wild Planet Jungle Resort | Very Good |
| Swast-Mast Resort        | Very Good |
| Alurkar Resort           | Very Good |
| Gavkari                  | Very Good |
| Vishal12                 | Very Good |
| sat30                    | Very Good |
| vivianr                  | Average   |
| sat22                    | Very Good |
+--------------------------+-----------+
14 rows in set, 4 warnings (0.009 sec)

MariaDB [resort_590]>
```

STORED PROCEDURE

1. display information of all resorts in a given city

```
DELIMITER $$
CREATE OR REPLACE PROCEDURE get__resort_count(IN city varchar(15), OUT r_count integer)
BEGIN
SELECT COUNT(*) into r_count FROM resort where address=city; END $$
DELIMITER ;

SET @r_count=0;
CALL get__resort_count(“Nanadi”, @r_count); SELECT @r_count;
```

```
MariaDB [resort_590]> DELIMITER $$
MariaDB [resort_590]> CREATE OR REPLACE PROCEDURE get__resort_count(IN city varchar(15), OUT r_count integer)
    -> BEGIN
    -> SELECT COUNT(*) into r_count FROM resort where address=city; END $$
Query OK, 0 rows affected (0.007 sec)

MariaDB [resort_590]> DELIMITER ;
MariaDB [resort_590]>
MariaDB [resort_590]> SET @r_count=0;
Query OK, 0 rows affected (0.000 sec)

MariaDB [resort_590]> CALL get__resort_count(“Nanadi”, @r_count); SELECT @r_count;
Query OK, 1 row affected (0.035 sec)

+----------+
| @r_count |
+----------+
|        1 |
+----------+
1 row in set (0.000 sec)

MariaDB [resort_590]> select *from resort where address="nanadi";
+-----------+-------------+---------+--------+---------------+
| resort_id | resort_name | address | rating | price_per_day |
+-----------+-------------+---------+--------+---------------+
|      1208 | Vishal12    | Nanadi  |   4.50 |        100000 |
+-----------+-------------+---------+--------+---------------+
1 row in set (0.001 sec)
```

TRIGGERS

1. **display error message when a new entry is added to a resort which is already booked.**

```
DELIMITER $$
CREATE OR REPLACE TRIGGER valid_reservation_on_insert BEFORE INSERT
ON reservation FOR EACH ROW BEGIN
DECLARE error_msg VARCHAR(255);
SET error_msg = ("Resort is already booked!");
-- if checkin for new entry is before checkout for that resort IF NEW.checkin < (select
checkout from reservation where
resort_id=NEW.resort_id) THEN SIGNAL SQLSTATE='45000'
SET MESSAGE_TEXT = error_msg; END IF;
END $$

CREATE OR REPLACE TRIGGER valid_reservation_on_update BEFORE UPDATE
ON reservation FOR EACH ROW BEGIN
DECLARE error_msg VARCHAR(255);
SET error_msg = ("Resort is already booked!");
-- if checkin for new entry is before checkout for that resort IF NEW.checkin < (select
checkout from reservation where
resort_id=NEW.resort_id) THEN SIGNAL SQLSTATE '45000'
SET MESSAGE_TEXT = error_msg; END IF;
END $$ DELIMITER ;
```

```
MariaDB [resort_590]> DELIMITER $$
MariaDB [resort_590]> CREATE OR REPLACE TRIGGER valid_reservation_on_insert
    -> BEFORE INSERT
    -> ON reservation FOR EACH ROW
    -> BEGIN
    -> DECLARE error_msg VARCHAR(255);
    -> SET error_msg = ("Resort is already booked!");
    -> -- if checkin for new entry is before checkout for that resort
    -> IF NEW.checkin < (select checkout from reservation where
    -> resort_id=NEW.resort_id) THEN
    -> SIGNAL SQLSTATE '45000'
    -> SET MESSAGE_TEXT = error_msg;
    -> END IF;
    -> END $$
Query OK, 0 rows affected (0.015 sec)

MariaDB [resort_590]> CREATE OR REPLACE TRIGGER valid_reservation_on_update
    -> BEFORE UPDATE
    -> ON reservation FOR EACH ROW
    -> BEGIN
    -> DECLARE error_msg VARCHAR(255);
    -> SET error_msg = ("Resort is already booked!");
    -> -- if checkin for new entry is before checkout for that resort
    -> IF NEW.checkin < (select checkout from reservation where
    -> resort_id=NEW.resort_id) THEN
    -> SIGNAL SQLSTATE '45000'
    -> SET MESSAGE_TEXT = error_msg;
    -> END IF;
    -> END $$
Query OK, 0 rows affected (0.306 sec)

MariaDB [resort_590]> DELIMITER ;
```

Example

```
MariaDB [resort_590]> insert into customer values (4444,"Sourabh","","varma
","karnataka","sad@gmail.com",1234566789,NULL);
Query OK, 1 row affected (0.005 sec)

MariaDB [resort_590]> select * from reservation;
+------+-----------+------------+------------+--------+
| cid  | resort_id | checkin    | checkout   | amount |
+------+-----------+------------+------------+--------+
| 1001 |      1001 | 2022-11-11 | 2022-11-21 |  15592 |
| 3310 |      1001 | 2023-02-08 | 2023-02-21 |  25987 |
| 1002 |      1002 | 2022-11-15 | 2022-11-22 |  12593 |
| 9510 |      1002 | 2023-02-08 | 2023-02-21 |  23387 |
| 1004 |      1004 | 2022-11-16 | 2022-11-21 |   7495 |
| 1005 |      1005 | 2022-11-18 | 2022-11-21 | 4077.6 |
| 1006 |      1006 | 2022-11-20 | 2022-11-26 |   8994 |
| 6800 |      1007 | 2023-02-08 | 2023-02-21 |  16887 |
| 1804 |      5465 | 2022-12-01 | 2022-12-07 |  60000 |
+------+-----------+------------+------------+--------+
9 rows in set (0.000 sec)

MariaDB [resort_590]> insert into reservation values(4444,1003,"2022-11-25"
,"2022-11-30",1699,000000);
ERROR 1136 (21S01): Column count doesn't match value count at row 1
MariaDB [resort_590]> insert into reservation values(4444,1003,"2022-11-25"
,"2022-11-30",1699.000000);
Query OK, 1 row affected (0.003 sec)

MariaDB [resort_590]>
```

Successful insertion

Case when error is triggered

```
MariaDB [resort_590]> insert into customer values (2222,"pratik","","katti"
,"karnataka","sadadd@gmail.com",1111566789,NULL);
Query OK, 1 row affected (0.004 sec)
```

```
MariaDB [resort_590]> insert into reservation values(2222,1003,"2022-11-28"
,"2022-12-04",1699.000000);
ERROR 1644 (45000): Resort is already booked!
MariaDB [resort_590]> select * from reservation where resort_id=1003;
+------+-----------+------------+------------+--------+
| cid  | resort_id | checkin    | checkout   | amount |
+------+-----------+------------+------------+--------+
| 4444 |      1003 | 2022-11-25 | 2022-11-30 |   1699 |
+------+-----------+------------+------------+--------+
1 row in set (0.000 sec)

MariaDB [resort_590]>
```

CURSORS

Cursor :
To Create backup table for food item

```
CREATE TABLE `Curbackup_foodItem` (

 `food_id` int(11) DEFAULT NULL,
 `food_name` varchar(30) DEFAULT NULL,
 `price` int(11) DEFAULT NULL
);

DELIMITER $$
CREATE PROCEDURE curs()
BEGIN
DECLARE done INT DEFAULT 0;
DECLARE foodid , price int(10);
DECLARE foodname VARCHAR(20);
DECLARE cur CURSOR FOR SELECT * FROM food_item;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
OPEN cur;
label: LOOP
FETCH cur INTO foodid ,  foodname , price;
INSERT INTO Curbackup_foodItem VALUES(foodid ,  foodname , price);
IF done = 1 THEN LEAVE label;
END IF;
END LOOP;
CLOSE cur;
END;$$
DELIMITER ;
```

Developing a Frontend

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table – Done ✅
2. There should be a window to accept and run any SQL statement and display the result – Done ✅

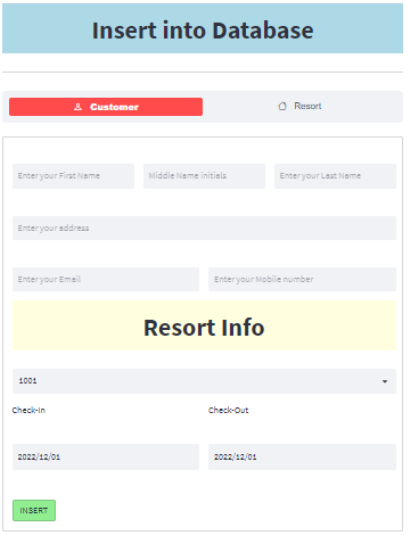
Frontend made using Streamlit

Frontend Visuals
›

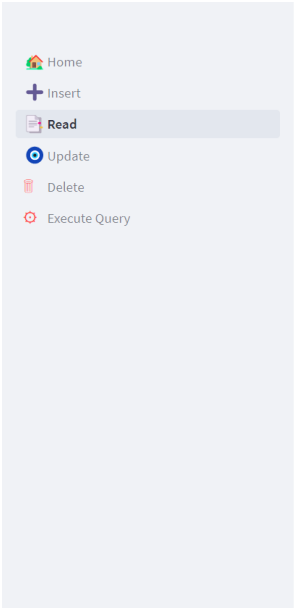
Sat30 Resorts

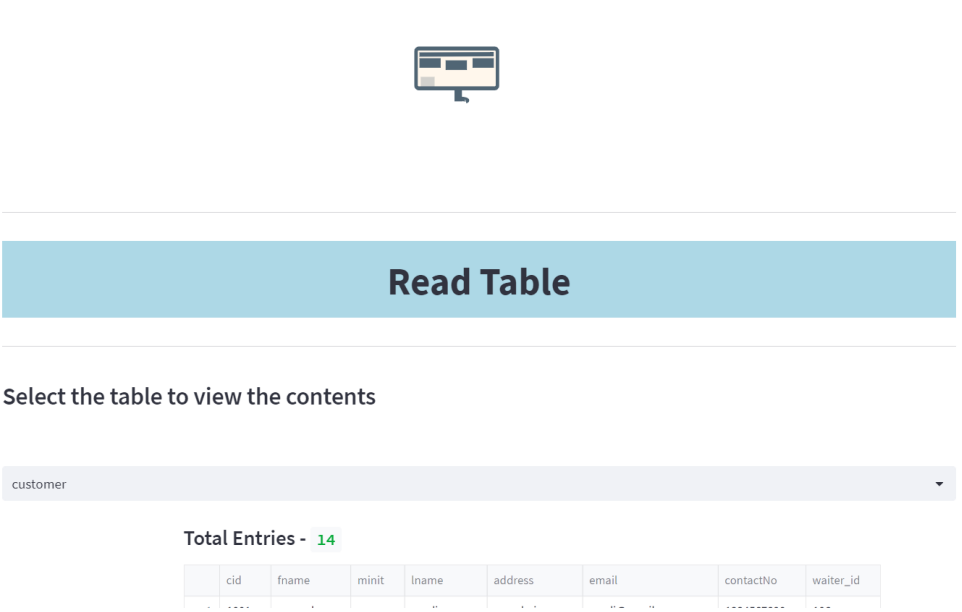
Off to get some vitamin sea!

INSERT - OPERATION

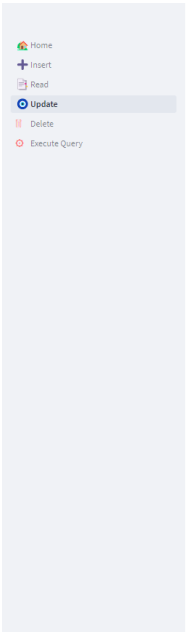


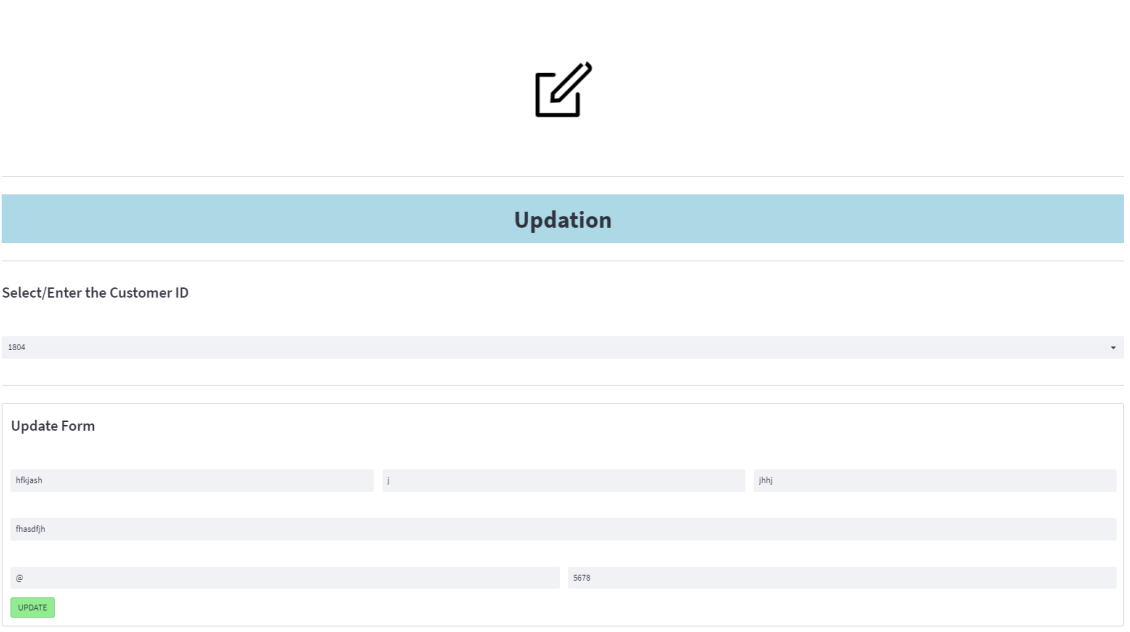
READ-OPERATION



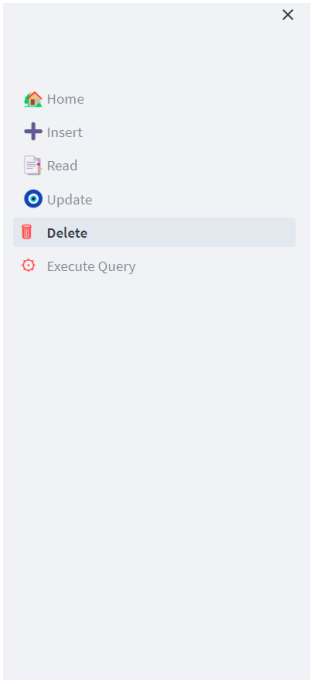


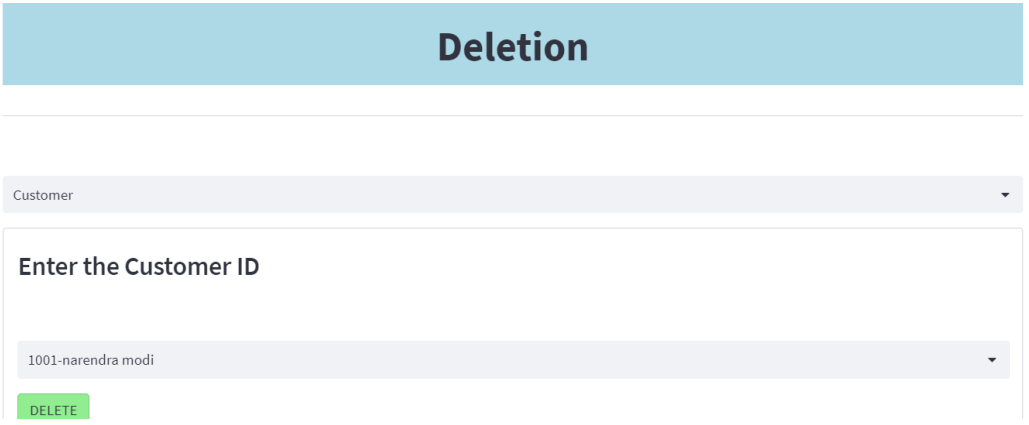
UPDATE-OPERATION





DELETE-OPERATION





QUERY-EXECUTION