Project Report on

# Hotel Management System

Developed by

**Pankhania Aanandi (IT-081) Department of IT, DD University**
**Nadpara Princy (IT-074) Department of IT, DD University**

**Guided By**
**Internal Guide:**
**Prof. Vidhi B. Chaudhari**
**Department of Information Technology**
**Faculty of Technology**
**DD University**

**Department of Information Technology**
**Faculty of Technology, Dharmsinh Desai University**
**College Road, Nadiad-387001**
**October-2020**

# DHARMSINH DESAI UNIVERSITY
## NADIAD-387001, GUJARAT

# CERTIFICATE

This is to certify that the project entitled "**Hotel Management System**" is bonafied report of the work carried out by

1) **Ms. Aanandi Pankhania,** Student ID No : **18ITUBS057**

2) **Ms. Princy Nadpara**, Student ID No : **18ITUOS107**

of Department of Information Technology, semester V, under the guidance and supervision for the subject Database Management System. They were involved in Project training during academic year 2020-2021.

**Prof. Vidhi B. Chaudhari**
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad Date:

**Prof. Vipul Dabhi**
Head, Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad Date:

# ACKNOWLEDGEMENT

We would like to give our sincere acknowledgement to everybody responsible for the successful completion of our project “HOTEL MANAGEMENT SYSTEM”.

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of this project.

We owe our deep gratitude to our project guide Prof. Vidhi B. Chaudhari, who took been interest on our project work and guided us all along till the completion of our project work by providing all the necessary help for developing a good Database System.

We would also like to thank all our lecturers.

Finally we convoy our acknowledgement to all our friends and family members who directly or indirectly associated with us in the successful completion of the project.We thank one and all.

Thanking You,
Aanandi Pankhania (IT-081)
Princy Nadpara (IT-074)

# TABLE OF CONTENTS

# 1.SYSTEM OVERVIEW

## 1.1CURRENT SYSTEM

- The Current File Based System of Hotel Management requires a lot of time to store a record in a file or to Manage The data. The insertion is easy but Sort the data is more difficult and That would take much more time to do that.

- Hotel Management would have lot of data like rooms' details, room's category, facilities provided by hotel, customer's details, booking history, etc. Using Current Hotel Management above data can't be retrieved easily and operations will not be easy to be done. So, we need a system which would take less efforts to do operations and Store the data will be easy and won't require human hands.

## 1.2 OBJECTIVES OF THE PROPOSED SYSTEM

- This project intends to introduce more user friendliness in the various activities such as record updating, maintenance, and searching.

- The searching of record has been made quite simple as all the details of the customer can be obtained by simply keying in the identification of that customer.

- Similarly, Record Maintenance and updating can also be accomplished by using the identification of the customer with all the details being automatically generated. These details are also being promptly automatically updates in the master file thus keeping the record absolutely up-to-date.

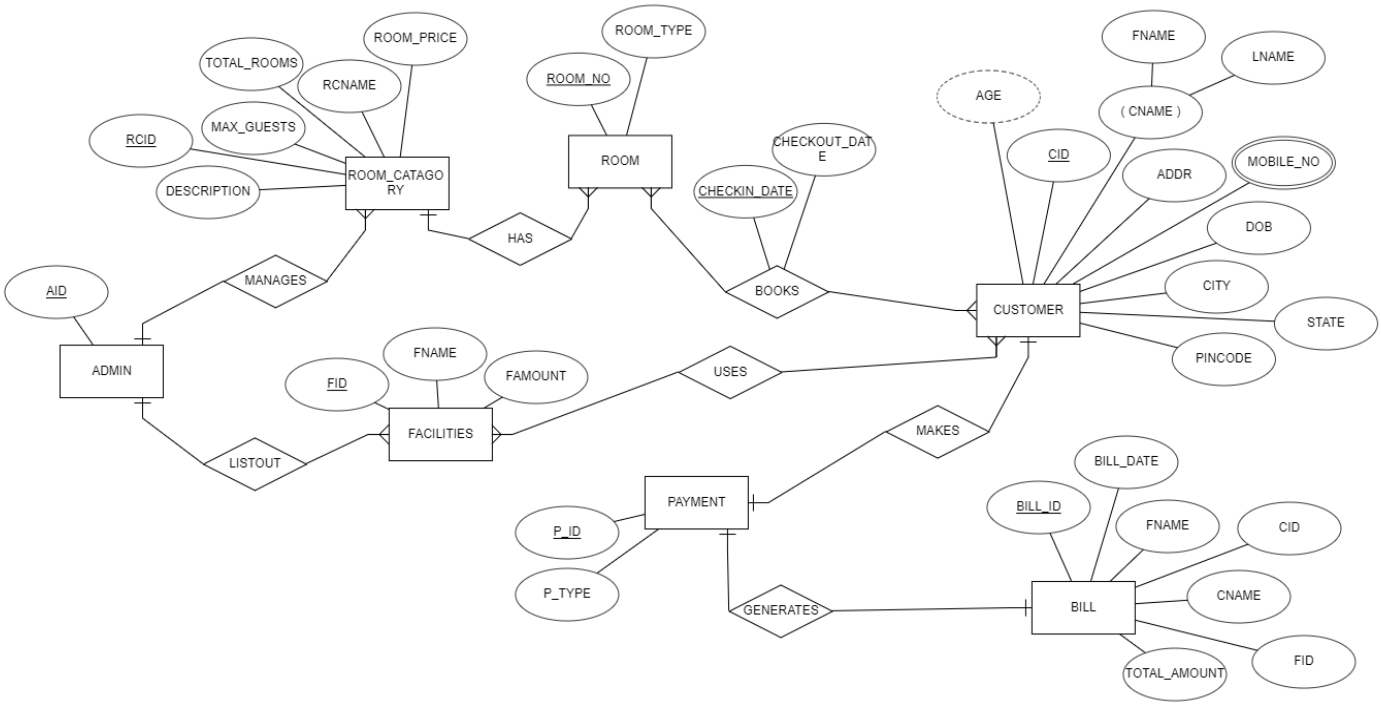
- The Main objective of the entire Activity is to automate the process of day to day activities of Hotel like:

    1. Add a New Customer
    2. Assign a Room and required facilities as per Customer's demand
    3. Admin can add new Room categories and Facilities whenever needed.
    4. Computing the total Bill having charges of both Stay and Facilities too.

## 1.3 ADVANTAGES OF THE PROPOSED) SYSTEM

- you will have all the information to better understand your guests, their preferences and what upgrades you sell the most.
- Payments are also greatly eased with an online booking platform. You can require from your guests to pay when they book, further increasing your income, and not worry about payments when they arrive.
- Online hotel management systems reduce workloads for your staff and optimize customer service. These platforms can make sure that bookings are synced and the availability is updated with each booking processing.

# 2.E-R DIAGRAM


ROOM_PRICE
TOTAL_ROOMS
RCNAME
RCID
MAX_GUESTS
DESCRIPTION
ROOM_CATAGORY
ROOM_TYPE
ROOM_NO
ROOM
HAS
MANAGES
AID
ADMIN
CHECKOUT_DATE
CHECKIN_DATE
BOOKS
FNAME
LNAME
AGE
( CNAME )
CID
ADDR
MOBILE_NO
DOB
CITY
STATE
PINCODE
CUSTOMER
FNAME
FID
FAMOUNT
FACILITIES
USES
LISTOUT
MAKES
PAYMENT
P_ID
P_TYPE
GENERATES
BILL_DATE
BILL_ID
FNAME
CID
CNAME
BILL
TOTAL_AMOUNT
FID

# 3.DATA DICTIONARY

## 3.1 ADMIN

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | AID 🔑 | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.2 ROOM_CATAGORY

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | RCNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | RCID 🔑 | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 3 | MAX_GUESTS | decimal(2,0) | | | No | *None* | | |
| ☐ | 4 | TOTAL_ROOMS | int(11) | | | No | *None* | | |
| ☐ | 5 | ROOM_PRICE | int(11) | | | No | *None* | | |
| ☐ | 6 | DESCRIPTION | varchar(50) | utf8mb4_general_ci | | Yes | *NULL* | | |
| ☐ | 7 | AID 🔗 | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.3 ROOM

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | ROOM_NO 🔑 | int(11) | | | No | *None* | | |
| ☐ | 2 | RCID 🔗 | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.4 FACILITIES

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | FID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | FNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 3 | FAMOUNT | int(11) | | | No | *None* | | |
| ☐ | 4 | AID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.5 CUSTOMER

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | FNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 3 | LNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 4 | ADDR | varchar(30) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 5 | PINCODE | decimal(6,0) | | | No | *None* | | |
| ☐ | 6 | CITY | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 7 | STATE | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 8 | DOB | date | | | No | *None* | | |
| ☐ | 9 | P_ID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.6 BILL

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | BILL_ID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | BILL_DATE | date | | | No | *None* | | |
| ☐ | 3 | TOTAL_AMOUNT | int(11) | | | No | *None* | | |
| ☐ | 4 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 5 | FID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.7 PAYMENT

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | P_ID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| 2 | P_TYPE | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| 3 | BILL_ID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.8 BOOKS

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | CHECKIN_DATE | date | | | No | *None* | | |
| 2 | CHECKOUT_DATE | date | | | No | *None* | | |
| 3 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| 4 | ROOM_NO | int(11) | | | No | *None* | | |

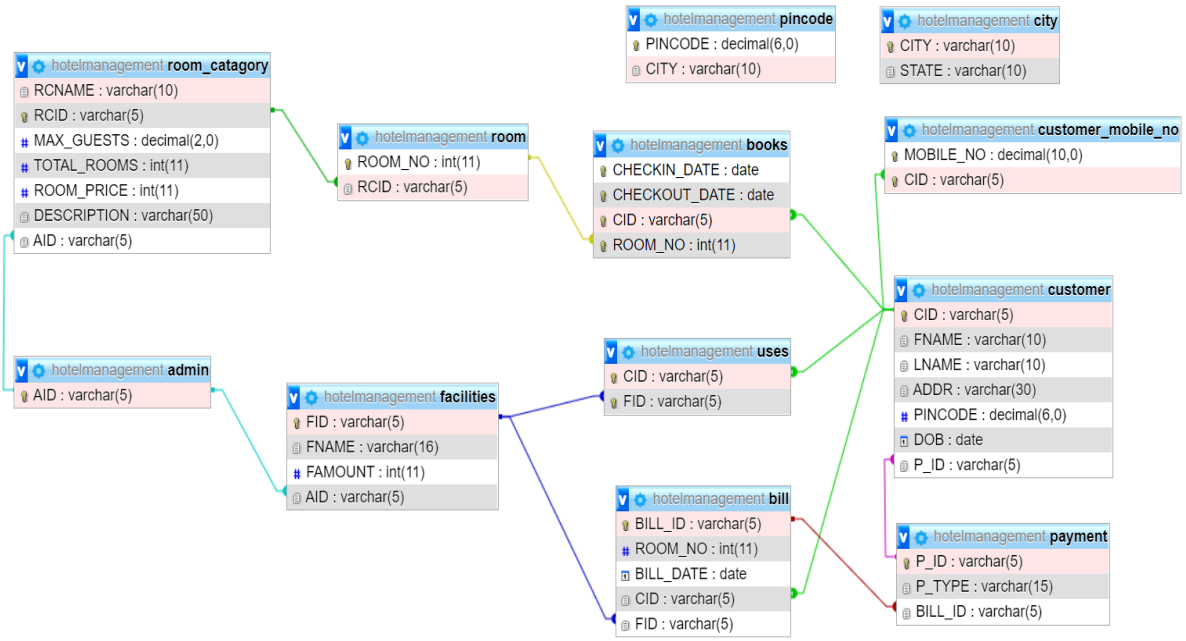
## 3.9 USES

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| 2 | FID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## 3.10 CUSTOMER_MOBILE_NO

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | MOBILE_NO | decimal(10,0) | | | No | *None* | | |
| 2 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

# 4.SCHEMA DIAGRAM



## AFTER NORMALISATION (TO 3NF):

## CUSTOMER→

## CUTOMER:

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | CID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | FNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 3 | LNAME | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 4 | ADDR | varchar(30) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 5 | PINCODE | decimal(6,0) | | | No | *None* | | |
| ☐ | 6 | DOB | date | | | No | *None* | | |
| ☐ | 7 | P_ID | varchar(5) | utf8mb4_general_ci | | No | *None* | | |

## PINCODE:

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | PINCODE | decimal(6,0) | | | No | *None* | | |
| ☐ | 2 | CITY | varchar(10) | utf8mb4_general_ci | | No | *None* | | |

## CITY:

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | CITY | varchar(10) | utf8mb4_general_ci | | No | *None* | | |
| ☐ | 2 | STATE | varchar(10) | utf8mb4_general_ci | | No | *None* | | |

# 5.DATABASE IMPLEMENTATION

## 5.1 CREATE SCHEMA

### 5.1.1 ADMIN:

```
CREATE TABLE ADMIN
(
  AID VARCHAR(5) NOT NULL,
  PRIMARY KEY (AID)
);
```

### 5.1.2 FACILITIES:

```
CREATE TABLE FACILITIES
(
  FID VARCHAR(5) NOT NULL,
  FNAME VARCHAR(10) NOT NULL,
  FAMOUNT INT NOT NULL,
  AID VARCHAR(5) NOT NULL,
  PRIMARY KEY (FID),
  FOREIGN KEY (AID) REFERENCES ADMIN(AID)
);
```

### 5.1.3 ROOM_CATAGORY:

```
CREATE TABLE ROOM_CATAGORY
(
  RCNAME VARCHAR(10) NOT NULL,
  RCID VARCHAR(5) NOT NULL,
  MAX_GUESTS NUMERIC(2) NOT NULL,
  TOTAL_ROOMS INT NOT NULL,
  ROOM_PRICE INT NOT NULL,
  DESCRIPTION VARCHAR(15),
  AID VARCHAR(5) NOT NULL,
  PRIMARY KEY (RCID),
  FOREIGN KEY (AID) REFERENCES ADMIN(AID)
);
```

### 5.1.4 ROOM:

```
CREATE TABLE ROOM
(
  ROOM_NO INT NOT NULL,
  ROOM_TYPE VARCHAR(10) NOT NULL,
  RCID VARCHAR(5) NOT NULL,
  PRIMARY KEY (ROOM_NO),
  FOREIGN KEY (RCID) REFERENCES ROOM_CATAGORY(RCID)
);
```

### 5.1.5 CUSTOMER:

```
CREATE TABLE CUSTOMER
(
  CID VARCHAR(5) NOT NULL,
  FNAME VARCHAR(10) NOT NULL,
  LNAME VARCHAR(10) NOT NULL,
  ADDR VARCHAR(20) NOT NULL,
  PINCODE NUMERIC(6) NOT NULL,
  DOB DATE NOT NULL,
  P_ID VARCHAR(5) NOT NULL,
  PRIMARY KEY (CID),
  FOREIGN KEY (P_ID) REFERENCES PAYMENT(P_ID);
);
```

### 5.1.6 PINCODE:

```
CREATE TABLE PINCOE
(
  PINCODE NUMERIC(6) NOT NULL,
  CITY VARCHAR(10) NOT NULL,
  PRIMARY KEY (PINCODE)
);
```

### 5.1.7 CITY:

```
CREATE TABLE CITY
(
  CITY VARCHAR(10) NOT NULL,
  STATE VARCHAR(10) NOT NULL,
  PRIMARY KEY (CITY)
);
```

### 5.1.8 BILL:

```
CREATE TABLE BILL
(
  BILL_ID VARCHAR(5) NOT NULL,
  CNAME VARCHAR(10) NOT NULL,
  FNAME VARCHAR(10) NOT NULL,
  BILL_DATE DATE NOT NULL,
  TOTAL_AMOUNT INT NOT NULL,
  CID VARCHAR(5) NOT NULL,
  FID VARCHAR(5) NOT NULL,
  PRIMARY KEY (BILL_ID),
  FOREIGN KEY (CID) REFERENCES CUSTOMER(CID),
  FOREIGN KEY (FID) REFERENCES FACILITIES(FID)
);
```

### 5.1.9 PAYMENT:

```
CREATE TABLE PAYMENT
(
  P_ID VARCHAR(5) NOT NULL,
  P_TYPE VARCHAR(10) NOT NULL,
  BILL_ID VARCHAR(5) NOT NULL,
  PRIMARY KEY (P_ID),
  FOREIGN KEY (BILL_ID) REFERENCES BILL(BILL_ID)
);
```

### 5.1.10 BOOKS:

```
CREATE TABLE BOOKS
(
  CHECKIN_DATE DATE NOT NULL,
  CHECKOUT_DATE DATE NOT NULL,
  CID VARCHAR(5) NOT NULL,
  ROOM_NO INT NOT NULL,
  PRIMARY KEY (CHECKIN_DATE),
  FOREIGN KEY (CID) REFERENCES CUSTOMER(CID),
  FOREIGN KEY (ROOM_NO) REFERENCES ROOM(ROOM_NO),
  UNIQUE (ROOM_NO, CID)
);
```

### 5.1.11 USES:

```
CREATE TABLE USES
(
  CID VARCHAR(5) NOT NULL,
  FID VARCHAR(5) NOT NULL,
  PRIMARY KEY (CID, FID),
  FOREIGN KEY (CID) REFERENCES CUSTOMER(CID),
  FOREIGN KEY (FID) REFERENCES FACILITIES(FID)
);
```

### 5.1.12 CUSTOMER_MOBILE_NO:

```
CREATE TABLE CUSTOMER_MOBILE_NO
(
  MOBILE_NO NUMERIC(10) NOT NULL,
  CID VARCHAR(5) NOT NULL,
  PRIMARY KEY (MOBILE_NO, CID),
  FOREIGN KEY (CID) REFERENCES CUSTOMER(CID)
);
```

## 5.2 INSERT DATA VALUE

### 5.2.1 ADMIN

```
INSERT INTO ADMIN
VALUES('A1');
```

| AID |
| --- |
| A1 |

### 5.2.2 FACILITIES

```
INSERT INTO FACILITIES(FID,FNAME,FAMOUNT,AID) VALUES

('F01', 'SPA', 5000, 'A1'),
('F02', 'SWIMMING POOL', 3000, 'A1'),
('F03', 'PARKING', 1000, 'A1'),
('F04', 'DANCE BAR', 4000, 'A1'),
('F05', 'TRANSPORTATION', 6000, 'A1'),
('F06', 'GAME ZONE', 1000, 'A1'),
('F07', 'BANQUET HALL', 5000, 'A1');
```

| FID | FNAME | FAMOUNT | AID |
|---|---|---|---|
| F01 | SPA | 5000 | A1 |
| F02 | SWIMMING POOL | 3000 | A1 |
| F03 | PARKING | 1000 | A1 |
| F04 | DANCE BAR | 4000 | A1 |
| F05 | TRANSPORTATION | 6000 | A1 |
| F06 | GAME ZONE | 1000 | A1 |
| F07 | BANQUET HALL | 5000 | A1 |

### 5.2.3 ROOM_CATAGORY

INSERT INTO ROOM_CATAGORY( RCNAME, RCID, MAX_GUESTS, TOTAL_ROOMS, ROOM_PRICE, DESCRIPTION, AID) VALUES

('SINGLE', 'R01', 1, 15, 2500, 'WHICH HAS SINGLE BED', 'A1'),
('TRIPLE', 'R02', 3, 10, 4000, 'THREE TWIN BEDS', 'A1'),
('QUAD', 'R03', 4, 10, 5500, 'MORE THAN TWO BEDS', 'A1'),
('QUEEN', 'R04', 2, 20, 8500, 'QUEEN SIZED BED', 'A1'),
('KING', 'R05', 2, 15, 10000, 'KING SIZED BED', 'A1'),
('TWIN', 'R06', 2, 10, 3000, 'TWO TWIN BEDS', 'A1'),
('DUPLEX', 'R07', 4, 10, 8000, 'TWO BEDS WITH TWO FLOORS', 'A1');

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| SINGLE | R01 | 1 | 15 | 2500 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| QUEEN | R04 | 2 | 20 | 8500 | QUEEN SIZED BED | A1 |
| KING | R05 | 2 | 15 | 10000 | KING SIZED BED | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |
| DUPLEX | R07 | 4 | 10 | 8000 | TWO BEDS WITH TWO FLOORS | A1 |

### 5.2.4 ROOM

INSERT INTO ROOM(ROOM_NO,RCID) VALUES

(101, 'R01'),
(102, 'R01'),
(103, 'R04'),
(104, 'R02'),
(105, 'R06'),
(106, 'R06'),
(107, 'R07'),
(108, 'R03'),
(109, 'R05'),
(110, 'R02'),
(111, 'R04'),
(112, 'R03'),
(113, 'R01'),
(114, 'R07'),
(115, 'R01');

| ROOM_NO | RCID |
|---|---|
| 101 | R01 |
| 102 | R01 |
| 113 | R01 |
| 115 | R01 |
| 104 | R02 |
| 110 | R02 |
| 108 | R03 |
| 112 | R03 |
| 103 | R04 |
| 111 | R04 |
| 109 | R05 |
| 105 | R06 |
| 106 | R06 |
| 107 | R07 |
| 114 | R07 |

### 5.2.5 CUSTOMER

INSERT INTO CUSTOMER (CID, FNAME, LNAME, ADDR, PINCODE, DOB, P_ID) VALUES

('C01', 'Aanandi', 'Pankhania', '26 - B, Timiliyawad Rd, Mehar Park', 641009, DATE '2000-04-22', 'P01'),
('C02', 'Princy', 'Nadpara', 'INS Hospital, near valentino business hub', 360005, DATE '1998-02-28', 'P02'),
('C03', 'Vinay', 'Ghediya', '415, 4th floor, infinity tower, Ayurvedic collage', 110035, DATE '1997-07-26', 'P03'),
('C04', 'Sapna', 'Kumar', '26 - B, Timiliyawad Rd, Mehar Park', 400001, DATE '1997-03-22', 'P04'),
('C05', 'Rahul', 'Khunt', '305, Kashi Plaza Complex Opp, Dayalji Ashram Marg', 641009, DATE '2001-08-29', 'P05'),
('C06', 'Sneha', 'Malla', 'New Rander Rd, Giriraj Society, Adajan', 700007, DATE '2001-02-28', 'P06'),
('C07', 'Aaditya', 'Roy', '115-B, First floor, Zenon Beside Kiran motors', 110048, DATE '1999-04-07', 'P07'),
('C08', 'Sakshi', 'Sosa', 'A-101, Maan Darwaja, Udhna darwaja', 395002, DATE '1998-01-01', 'P08'),
('C09', 'Verna', 'Sevak', 'M-59, second floor, Shreeji Arcade', 387001, DATE '1997-12-07', 'P09'),
('C10', 'Jeel', 'Patel', 'Apple Hospital, Udhna Dazwaja, Ring Road', 395001, DATE '2000-06-07', 'P10'),
('C11', 'Yashvi', 'Vora', 'Swami Narayan Complex, G-4 , 4th Floor', 176605, DATE '2001-12-27', 'P11'),
('C12', 'Kumbhan', 'Pandya', '1st floor, sangini square, near kashi', 302011, DATE '1973-08-16', 'P12'),
('C13', 'Shyam', 'Joshi', 'Civil Char Rasta to Sosyo Circle Lane, Ring rd', 560008, DATE '1980-10-10', 'P13'),
('C14', 'Kapil', 'Dev', 'Behind Bhulka Bhavan School, Anand Mahal Rd', 641009, DATE '1990-09-29', 'P14'),
('C14', 'Kapil', 'Dev', 'Behind Bhulka Bhavan School, Anand Mahal Rd', 641009, DATE'1990-09-29', 'P14'),
('C15', 'Urvi', 'Joshi', 'House no. 6, First Floor New Camp Majnu-ka-tilla', 176605, DATE '2000-01-01', 'P15');

| CID | FNAME | LNAME | ADDR | PINCODE | DOB | P_ID |
|---|---|---|---|---|---|---|
| C01 | Aanandi | Pankhania | 26 - B, Timiliyawad Rd, Mehar | 641009 | 2000-04-22 | P01 |
| C02 | Princy | Nadpara | INS Hospital, near valentino b | 360005 | 1998-02-28 | P02 |
| C03 | Vinay | Ghediya | 415,4th floor, infinity tower, | 110035 | 1997-07-26 | P03 |
| C04 | Sapna | Kumar | 26 - B, Timiliyawad Rd, Mehar | 400001 | 1997-03-22 | P04 |
| C05 | Rahul | Khunt | 305, Kashi Plaza Complex Opp, | 641009 | 2001-08-29 | P05 |
| C06 | Sneha | Malla | New Rander Rd, Giriraj Society | 700007 | 2001-02-28 | P06 |
| C07 | Aaditya | Roy | 115-B, First floor, Zenon Besi | 110048 | 1999-04-07 | P07 |
| C08 | Sakshi | Sosa | A-101, Maan Darwaja, Udhna dar | 395002 | 1998-01-01 | P08 |
| C09 | Verna | Sevak | M-59, second floor, Shreeji Ar | 387001 | 1997-12-07 | P09 |
| C10 | Jeel | Patel | Apple Hospital, Udhna Dazwaja, | 395001 | 2000-06-07 | P10 |
| C11 | Yashvi | Vora | Swami Narayan Complex, G-4 , 4 | 176605 | 2001-12-27 | P11 |
| C12 | Kumbhan | Pandya | 1st floor, sangini square, nea | 302011 | 1973-08-16 | P12 |
| C13 | Shyam | Joshi | Civil Char Rasta to Sosyo Circ | 560008 | 1980-10-10 | P13 |
| C14 | Kapil | Dev | Behind Bhulka Bhavan School, A | 641009 | 1990-09-29 | P14 |
| C15 | Urvi | Joshi | House no. 6, First Floor New C | 176605 | 2000-01-01 | P15 |

### 5.2.6 PINCODE

INSERT INTO PINCODE(PINCODE,CITY) VALUES

(641009, 'Coimbatore'), (110035, 'Delhi'),
(110048, 'Delhi'), (360005, 'Rajkot'),
(360006, 'Rajkot'), (360007, 'Rajkot'),
(395001, 'Surat'), (395002, 'Surat'),
(400001, 'Mumbai'), (400002, 'Mumbai'),
(400009, 'Mumbai'), (700007, 'Kolkata'),
(302001, 'Jaipur'), (302011, 'Jaipur'),
(302098, 'Udaipur'), (176605, 'Jamnagar'),
(387001, 'Nadiad'), (322749, 'Pune'),
(560008, 'Bangalore');

| PINCODE | CITY |
|---|---|
| 110035 | Delhi |
| 110048 | Delhi |
| 176605 | Jamnagar |
| 302001 | Jaipur |
| 302011 | Jaipur |
| 302098 | Udaipur |
| 322749 | Pune |
| 360005 | Rajkot |
| 360006 | Rajkot |
| 360007 | Rajkot |
| 387001 | Nadiad |
| 395001 | Surat |
| 395002 | Surat |
| 400001 | Mumbai |
| 400002 | Mumbai |
| 400009 | Mumbai |
| 560008 | Bangalore |
| 641009 | Coimbatore |
| 700007 | Kolkata |

### 5.2.7 CITY

INSERT INTO CITY(CITY,STATE) VALUES

('Coimbatore', 'Tamil Nadu'),
('Delhi', 'Delhi'),
('Rajkot', 'Gujarat'),
('Jamnagar', 'Gujarat'),
('Surat', 'Gujarat'),
('Nadiad', 'Gujarat'),
('Mumbai', 'Maharashtra'),
('Pune', 'Maharashtra'),
('Kolkata', 'West Bengal'),
('Bangalore', 'Karnataka'),
('Jaipur', 'Rajasthan'),
('Udaipur', 'Rajasthan');

| CITY | STATE |
|---|---|
| Bangalore | Karnataka |
| Coimbatore | Tamil Nadu |
| Delhi | Delhi |
| Jaipur | Rajasthan |
| Jamnagar | Gujarat |
| Kolkata | West Benga |
| Mumbai | Maharashtr |
| Nadiad | Gujarat |
| Pune | Maharashtr |
| Rajkot | Gujarat |
| Surat | Gujarat |
| Udaipur | Rajasthan |

### 5.2.8 BILL

INSERT INTO BILL(BILL_ID,ROOM_NO,BILL_DATE,CID,FID) VALUES

('B01', 101, '2019-04-10', 'C01', 'F01'),
('B02', 102, '2019-02-20', 'C02', 'F01'),
('B03', 103, '2019-05-01', 'C03', 'F03'),
('B04', 104, '2020-03-02', 'C04', 'F03'),
('B05', 105, '2018-08-10', 'C05', 'F06'),
('B06', 106, '2019-02-05', 'C06', 'F02'),
('B07', 107, '2019-04-10', 'C07', 'F07'),
('B08', 108, '2017-01-06', 'C08', 'F03'),
('B09', 109, '2015-04-10', 'C09', 'F05'),
('B10', 110, '2016-03-10', 'C10', 'F04'),
('B11', 111, '2019-09-03', 'C11', 'F03'),
('B12', 112, '2020-01-01', 'C12', 'F01'),
('B13', 113, '2019-10-02', 'C13', 'F02'),
('B14', 114, '2018-11-06', 'C14', 'F05'),
('B15', 115, '2018-12-30', 'C15', 'F03');

| BILL_ID | ROOM_NO | BILL_DATE | CID | FID |
|---|---|---|---|---|
| B01 | 101 | 2019-04-10 | C01 | F01 |
| B02 | 102 | 2019-02-20 | C02 | F01 |
| B03 | 103 | 2019-05-01 | C03 | F03 |
| B04 | 104 | 2020-03-02 | C04 | F03 |
| B05 | 105 | 2018-08-10 | C05 | F06 |
| B06 | 106 | 2019-02-05 | C06 | F02 |
| B07 | 107 | 2019-04-10 | C07 | F07 |
| B08 | 108 | 2017-01-06 | C08 | F03 |
| B09 | 109 | 2015-04-10 | C09 | F05 |
| B10 | 110 | 2016-03-10 | C10 | F04 |
| B11 | 111 | 2019-09-03 | C11 | F03 |
| B12 | 112 | 2020-01-01 | C12 | F01 |
| B13 | 113 | 2019-10-02 | C13 | F02 |
| B14 | 114 | 2018-11-06 | C14 | F05 |
| B15 | 115 | 2018-12-30 | C15 | F03 |

### 5.2.9 PAYMENT

INSERT INTO PAYMENT(P_ID,P_TYPE,BILL_ID) VALUES

('P01', 'CASH', 'B01'),
('P02', 'DEBIT CARD', 'B02'),
('P03', 'CREDIT CARD', 'B03'),
('P04', 'CASH', 'B04'),
('P05', 'CASH', 'B05'),
('P06', 'CREDIT CARD', 'B06'),
('P07', 'DEBIT CARD', 'B07'),
('P08', 'PAYTM', 'B08'),
('P09', 'CREDIT CARD', 'B09'),
('P10', 'PAYTM', 'B10'),
('P11', 'CASH', 'B11'),
('P12', 'CREDIT CARD', 'B12'),
('P13', 'DEBIT CARD', 'B13'),
('P14', 'CASH', 'B14'),
('P15', 'PAYTM', 'B15');

| P_ID | P_TYPE | BILL_ID |
|---|---|---|
| P01 | CASH | B01 |
| P02 | DEBIT CARD | B02 |
| P03 | CREDIT CARD | B03 |
| P04 | CASH | B04 |
| P05 | CASH | B05 |
| P06 | CREDIT CARD | B06 |
| P07 | DEBIT CARD | B07 |
| P08 | PAYTM | B08 |
| P09 | CREDIT CARD | B09 |
| P10 | PAYTM | B10 |
| P11 | CASH | B11 |
| P12 | CREDIT CARD | B12 |
| P13 | DEBIT CARD | B13 |
| P14 | CASH | B14 |
| P15 | PAYTM | B15 |

### 5.2.10 BOOKS

INSERT INTO BOOKS(CHECKIN_DATE,CHECKOUT_DATE,CID,ROOM_NO) VALUES

('2019-01-12', '2019-01-16', 'C01', 101),
('2019-02-20', '2019-02-22', 'C02', 102),
('2019-05-01', '2019-05-06', 'C03', 103),
('2020-03-02', '2020-03-06', 'C04', 104),
('2018-08-10', '2018-08-12', 'C05', 105),
('2019-02-05', '2019-02-15', 'C06', 106),
('2019-04-10', '2019-04-15', 'C07', 107),
('2017-01-06', '2017-01-10', 'C08', 108),
('2015-04-10', '2015-04-15', 'C09', 109),
('2016-03-10', '2016-03-15', 'C10', 110),
('2019-09-03', '2019-09-06', 'C11', 111),
('2020-01-01', '2020-01-03', 'C12', 112),
('2019-10-02', '2019-10-06', 'C13', 113),
('2018-11-06', '2018-11-09', 'C14', 114),
('2018-12-30', '2019-01-01', 'C15', 115);

| CHECKIN_DATE | CHECKOUT_DATE | CID | ROOM_NO |
|---|---|---|---|
| 2019-01-12 | 2019-01-16 | C01 | 101 |
| 2019-02-20 | 2019-02-22 | C02 | 102 |
| 2019-05-01 | 2019-05-06 | C03 | 103 |
| 2020-03-02 | 2020-03-06 | C04 | 104 |
| 2018-08-10 | 2018-08-12 | C05 | 105 |
| 2019-02-05 | 2019-02-15 | C06 | 106 |
| 2019-04-10 | 2019-04-15 | C07 | 107 |
| 2017-01-06 | 2017-01-10 | C08 | 108 |
| 2015-04-10 | 2015-04-15 | C09 | 109 |
| 2016-03-10 | 2016-03-15 | C10 | 110 |
| 2019-09-03 | 2019-09-06 | C11 | 111 |
| 2020-01-01 | 2020-01-03 | C12 | 112 |
| 2019-10-02 | 2019-10-06 | C13 | 113 |
| 2018-11-06 | 2018-11-09 | C14 | 114 |
| 2018-12-30 | 2019-01-01 | C15 | 115 |

### 5.2.11 USES

INSERT INTO USES(CID, FID) VALUES

('C01','F01'), ('C01','F02'),
('C01','F05'), ('C02','F01'),
('C02','F03'), ('C03','F03'),
('C03','F06'), ('C04','F01'),
('C04','F03'), ('C04','F04'),
('C05','F03'), ('C05','F06'),
('C06','F03'), ('C07','F02'),
('C07','F03'), ('C08','F03'),
('C09','F02'), ('C09','F03'),
('C10','F01'), ('C10','F02'),
('C10','F05'), ('C11','F01'),
('C11','F06'), ('C12','F01'),
('C12','F03'), ('C13','F03'),
('C14','F03'), ('C14','F06'),
('C15','F03');

| CID | FID |
|---|---|
| C01 | F01 |
| C01 | F02 |
| C01 | F05 |
| C02 | F01 |
| C02 | F03 |
| C03 | F03 |
| C03 | F06 |
| C04 | F01 |
| C04 | F03 |
| C04 | F04 |
| C05 | F03 |
| C05 | F06 |
| C06 | F03 |
| C07 | F02 |
| C07 | F03 |

| CID | FID |
|---|---|
| C08 | F03 |
| C09 | F02 |
| C09 | F03 |
| C10 | F01 |
| C10 | F02 |
| C10 | F05 |
| C11 | F01 |
| C11 | F06 |
| C12 | F01 |
| C12 | F03 |
| C13 | F03 |
| C14 | F03 |
| C14 | F06 |
| C15 | F03 |

### 5.2.12 CUSTOMER_MOBILE_NO

INSERT INTO CUSTOMER_MOBILE_NO(MOBILE_NO,CID) VALUES

(9851644854, 'C01'), (9482515465, 'C01'),
(9685155131, 'C02'), (7875468641, 'C03'),
(9965846546, 'C04'), (7954658465, 'C04'),
(8554879644, 'C05'), (9235146846, 'C06'),
(9475628582, 'C06'), (8724524527, 'C06'),
(9298414734, 'C07'), (7631571376, 'C08'),
(9137413641, 'C09'), (9823853434, 'C10'),
(9343873164, 'C11'), (8731734613, 'C11'),
(6731673654, 'C12'), (8164835484, 'C13'),
(7813483746, 'C14'), (8316435464, 'C15');

| MOBILE_NO | CID ▲ 1 |
|---|---|
| 9482515465 | C01 |
| 9851644854 | C01 |
| 9685155131 | C02 |
| 7875468641 | C03 |
| 7954658465 | C04 |
| 9965846546 | C04 |
| 8554879644 | C05 |
| 8724524527 | C06 |
| 9235146846 | C06 |
| 9475628582 | C06 |
| 9298414734 | C07 |
| 7631571376 | C08 |
| 9137413641 | C09 |
| 9823853434 | C10 |
| 8731734613 | C11 |
| 9343873164 | C11 |
| 6731673654 | C12 |
| 8164835484 | C13 |
| 7813483746 | C14 |
| 8316435464 | C15 |

# 5.3 QUERIES

## 5.3.1 Display Room Price with lowest and highest amount From Room category. //MIN, MAX

SELECT MIN(ROOM_PRICE) AS LOWEST_AMOUNT, MAX(ROOM_PRICE) AS HIGHEST_AMOUNT
FROM ROOM_CATAGORY;

| LOWEST_AMOUNT | HIGHEST_AMOUNT |
|---|---|
| 2500 | 10000 |

## 5.3.2 Display The Details Of Room Category Sorted by Room price in ascending order. //ORDER BY

SELECT RCID,RCNAME,ROOM_PRICE
FROM ROOM_CATAGORY
ORDER BY ROOM_PRICE ASC;

| RCID | RCNAME | ROOM_PRICE ▲ 1 |
|---|---|---|
| R01 | SINGLE | 2500 |
| R06 | TWIN | 3000 |
| R02 | TRIPLE | 4000 |
| R03 | QUAD | 5500 |
| R07 | DUPLEX | 8000 |
| R04 | QUEEN | 8500 |
| R05 | KING | 10000 |

## 5.3.3 Details of Customer whose name starts with 'A'. //LIKE

SELECT * FROM CUSTOMER
WHERE FNAME LIKE 'A%' or FNAME LIKE 'a%';

| CID | FNAME | LNAME | ADDR | PINCODE | DOB | P_ID |
|---|---|---|---|---|---|---|
| C01 | Aanandi | Pankhania | 26 - B, Timiliyawad Rd, Mehar | 641009 | 2000-04-22 | P01 |
| C07 | Aaditya | Roy | 115-B, First floor, Zenon Besi | 110048 | 1999-04-07 | P07 |

### 5.3.4 Count How many Rooms are there in Hotel. //COUNT

```
SELECT COUNT(ROOM_NO)
FROM ROOM;
```

| COUNT(ROOM_NO) |
|---|
| 15 |

### 5.3.5 Details of Room Category whose room rent is below 6000. //BETWEEN

```
SELECT * FROM ROOM_CATAGORY
WHERE ROOM_PRICE BETWEEN 0 AND 6000;
```

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| SINGLE | R01 | 1 | 15 | 2500 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |

### 5.3.6 Count the total no. of City available in a particular state and sort it by count of no. of city. //GROUP BY, ORDER BY

```
SELECT COUNT(CITY), STATE
FROM CITY
GROUP BY STATE;
ORDER BY COUNT(CITY) ASC;
```

| COUNT(CITY) ▲ 1 | STATE |
|---|---|
| 1 | Karnataka |
| 1 | Tamil Nadu |
| 1 | West Benga |
| 1 | Delhi |
| 2 | Maharashtr |
| 2 | Rajasthan |
| 4 | Gujarat |

### 5.3.7 Display Details of Customers Name, customer id with their respective Payment id, Payment Type in ascending order with respect to payment type. //INNER JOIN

```
SELECT C.CID, C.FNAME, C.LNAME, P.P_ID, P.P_TYPE
FROM CUSTOMER C
INNER JOIN PAYMENT P
ON C.P_ID = P.P_ID
ORDER BY P_TYPE ASC;
```

| CID | FNAME | LNAME | P_ID | P_TYPE ▲ 1 |
|---|---|---|---|---|
| C01 | Aanandi | Pankhania | P01 | CASH |
| C05 | Rahul | Khunt | P05 | CASH |
| C04 | Sapna | Kumar | P04 | CASH |
| C11 | Yashvi | Vora | P11 | CASH |
| C14 | Kapil | Dev | P14 | CASH |
| C09 | Verna | Sevak | P09 | CREDIT CARD |
| C12 | Kumbhan | Pandya | P12 | CREDIT CARD |
| C03 | Vinay | Ghediya | P03 | CREDIT CARD |
| C06 | Sneha | Malla | P06 | CREDIT CARD |
| C13 | Shyam | Joshi | P13 | DEBIT CARD |
| C07 | Aaditya | Roy | P07 | DEBIT CARD |
| C02 | Princy | Nadpara | P02 | DEBIT CARD |
| C08 | Sakshi | Sosa | P08 | PAYTM |
| C15 | Urvi | Joshi | P15 | PAYTM |
| C10 | Jeel | Patel | P10 | PAYTM |

### 5.3.8 Give Details of Pincode and State from same City using Left outer join. //LEFT OUTER JOIN

```
SELECT P.PINCODE, P.CITY,C.STATE
FROM PINCODE P
LEFT OUTER JOIN CITY C
ON P.CITY= C.CITY ;
```

| PINCODE | CITY | STATE |
|---|---|---|
| 110035 | Delhi | Delhi |
| 110048 | Delhi | Delhi |
| 176605 | Jamnagar | Gujarat |
| 302001 | Jaipur | Rajasthan |
| 302011 | Jaipur | Rajasthan |
| 302098 | Udaipur | Rajasthan |
| 322749 | Pune | Maharashtr |
| 360005 | Rajkot | Gujarat |
| 360006 | Rajkot | Gujarat |
| 360007 | Rajkot | Gujarat |
| 387001 | Nadiad | Gujarat |
| 395001 | Surat | Gujarat |
| 395002 | Surat | Gujarat |
| 400001 | Mumbai | Maharashtr |
| 400002 | Mumbai | Maharashtr |
| 400009 | Mumbai | Maharashtr |
| 560008 | Bangalore | Karnataka |
| 641009 | Coimbatore | Tamil Nadu |
| 700007 | Kolkata | West Benga |

### 5.3.9 Display Details of Customers' first name and pincode who are living in same city. //SELF-JOIN

```
SELECT A.FNAME AS CUSTOMER1, B.FNAME AS CUSTOMER2, A.PINCODE
FROM CUSTOMER A, CUSTOMER B
WHERE A.CID != B.CID
AND A.PINCODE = B.PINCODE
ORDER BY A.FNAME ASC;
```

| Customer1 | Customer2 | pincode |
|---|---|---|
| Aanandi | Rahul | 641009 |
| Aanandi | Kapil | 641009 |
| Kapil | Rahul | 641009 |
| Kapil | Aanandi | 641009 |
| Rahul | Aanandi | 641009 |
| Rahul | Kapil | 641009 |
| Urvi | Yashvi | 176605 |
| Yashvi | Urvi | 176605 |

### 5.3.10 Display Details of room category id, name and description of Room number 101. //SUBQUERY

```
SELECT RC.RCID, RC.RCNAME, RC.DESCRIPTION
FROM ROOM_CATAGORY RC
WHERE ( SELECT ROOM_NO FROM ROOM R WHERE R.RCID=RC.RCID AND
ROOM_NO=101 );
```

| RCID | RCNAME | DESCRIPTION |
|---|---|---|
| R01 | SINGLE | WHICH HAS SINGLE BED |

# 5.4 PROCEDURES AND EXCEPTION HANDLING

### 5.4.1 This "CustData" Procedure Works Like Insert Query But If Pincode Is Greater Than 6 Then It will not Add To Customer Table Otherwise Data Will Be Inserted in Customer Table.

```
DELIMITER $$
create procedure CustData(IN CID varchar(5),IN FNAME varchar(10),IN LNAME
varchar(10),IN ADDR varchar(30),IN PINCODE decimal(6,0),IN DOB DATE,IN P_ID
varchar(5))
begin
        DECLARE Counter varchar(4);
        select count(CID) into Counter from CUSTOMER;
        set counter = counter + 1;
                IF length(PINCODE) <=> 6 THEN
                        INSERT into CUSTOMER VALUES
(CID,FNAME,LNAME,ADDR,PINCODE,DOB,P_ID);
                        SELECT 'YOUR DATA IS INSERTED' AS MESSAGE;
                        ELSE
                                SELECT 'Enter Valid Pincode' AS ERROR;
                        END IF;
END $$
DELIMITER ;

Call CustData('C16','Neha','Makvana','E-15, Rajiv Chowk, Block E, Connaught
Place',360001,DATE'2000-12-12','P16');
```

## Output Message:

| MESSAGE |
| --- |
| YOUR DATA IS INSERTED |

## Customer Table:

| | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| C10 | Jeel | Patel | Apple Hospital, Udhna Dazwaja, | 395001 | 2000-06-07 | P10 |
| C11 | Yashvi | Vora | Swami Narayan Complex, G-4 , 4 | 176605 | 2001-12-27 | P11 |
| C12 | Kumbhan | Pandya | 1st floor, sangini square, nea | 302011 | 1973-08-16 | P12 |
| C13 | Shyam | Joshi | Civil Char Rasta to Sosyo Circ | 560008 | 1980-10-10 | P13 |
| C14 | Kapil | Dev | Behind Bhulka Bhavan School, A | 641009 | 1990-09-29 | P14 |
| C15 | Urvi | Joshi | House no. 6, First Floor New C | 176605 | 2000-01-01 | P15 |
| C16 | Neha | Makvana | E-15, Rajiv Chowk, Block E, Co | 360001 | 2000-12-12 | P16 |

Call CustData('C17','Neha','Makvana','E-15, Rajiv Chowk, Block E, Connaught Place',36001,DATE'2000-12-12','P17');

**Output Message:**

| ERROR |
| --- |
| Enter Valid Pincode |

## 5.5 FUNCTIONS

### 5.5.1 This "Total_Rooms" Function Is Calculating the Total No. of Rooms each Room category is having.

```
DELIMITER $$
create function TOTAL_ROOMS(RCID varchar(5)) RETURNS int DETERMINISTIC
begin
        DECLARE no int;
        select count(ROOM_NO) into no from ROOM R WHERE R.RCID=RCID;
        RETURN no;
end $$
DELIMITER ;

SELECT RCID, TOTAL_ROOMS(RCID) FROM ROOM GROUP BY RCID;
```

**Output:**

| RCID | TOTAL_ROOMS(RCID) |
| --- | --- |
| R01 | 4 |
| R02 | 2 |
| R03 | 2 |
| R04 | 2 |
| R05 | 1 |
| R06 | 2 |
| R07 | 2 |

# 5.6 TRIGGERS

**5.6.1 This "New_RC" Trigger Called Before Any Update Is Inserted Into Room category Table. This Trigger Insert The Track of modification of previos values i.e. before any modification apply in a different Table named RC_LOG.**

```
CREATE TABLE RC_LOG (
   id INT AUTO_INCREMENT PRIMARY KEY,
   RCID VARCHAR(5) NOT NULL,
   RCNAME VARCHAR(10) NOT NULL,
   changedate DATETIME DEFAULT NULL,
   action VARCHAR(50) DEFAULT NULL
);

DELIMITER $$
CREATE TRIGGER NEW_RC
   BEFORE UPDATE ON room_catagory
   FOR EACH ROW
        BEGIN
              INSERT INTO rc_log
              SET action = 'update',
              RCID = OLD.RCID,
              RCNAME = OLD.RCNAME,
              changedate = NOW();
        END $$
DELIMITER ;
```

## Before Updating the Value:

## Room_catagory Table:

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| SINGLE | R01 | 1 | 15 | 2500 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| QUEEN | R04 | 2 | 20 | 8500 | QUEEN SIZED BED | A1 |
| KING | R05 | 2 | 15 | 10000 | KING SIZED BED | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |
| DUPLEX | R07 | 4 | 10 | 8000 | TWO BEDS WITH TWO FLOORS | A1 |

## Update Value in Table:

```
UPDATE ROOM_CATAGORY
SET
   RCNAME = 'ONE_BED'
WHERE
   RCID='R01';
```

## RESULT:

## Room_Catagory Table:

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| ONE_BED | R01 | 1 | 15 | 2500 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| QUEEN | R04 | 2 | 20 | 8500 | QUEEN SIZED BED | A1 |
| KING | R05 | 2 | 15 | 10000 | KING SIZED BED | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |
| DUPLEX | R07 | 4 | 10 | 8000 | TWO BEDS WITH TWO FLOORS | A1 |

## RC_LOG:

| id | RCID | RCNAME | changedate | action |
|---|---|---|---|---|
| 1 | R01 | SINGLE | 2020-10-30 01:57:39 | update |

**5.6.2 This "PRICE_LOG" Trigger Called After updating The Value Into Room_price Field of Room_catagory Table. This Trigger Keeps the data Of the Previous amount of room and the updated amount so that we can keep track of updation.**

```
CREATE TABLE Pricelog (
   id INT AUTO_INCREMENT PRIMARY KEY,
   RCID VARCHAR(5),
   RCNAME VARCHAR(10),
   OldPrice INT,
   NewPrice INT,
   changedAt TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
);
```

```
DELIMITER $$
CREATE OR REPLACE TRIGGER PRICE_LOG
AFTER UPDATE ON ROOM_CATAGORY
FOR EACH ROW
BEGIN
    INSERT INTO Pricelog(RCID,RCNAME,OldPrice,NewPrice)
    VALUES(old.rcid, old.rcname, old.room_price, new.room_price);
END $$
DELIMITER ;
```

## Before updating The Value:

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| SINGLE | R01 | 1 | 15 | 2500 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| QUEEN | R04 | 2 | 20 | 8500 | QUEEN SIZED BED | A1 |
| KING | R05 | 2 | 15 | 10000 | KING SIZED BED | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |
| DUPLEX | R07 | 4 | 10 | 8000 | TWO BEDS WITH TWO FLOORS | A1 |

## Update The Value in Room_catagory Table:

```
UPDATE  `room_catagory` SET `ROOM_PRICE` = 1100 WHERE `RCID` = 'R01';
```

## After Updating The Value:

## Room_Catagory Table:

| RCNAME | RCID | MAX_GUESTS | TOTAL_ROOMS | ROOM_PRICE | DESCRIPTION | AID |
|---|---|---|---|---|---|---|
| SINGLE | R01 | 1 | 15 | 1100 | WHICH HAS SINGLE BED | A1 |
| TRIPLE | R02 | 3 | 10 | 4000 | THREE TWIN BEDS | A1 |
| QUAD | R03 | 4 | 10 | 5500 | MORE THAN TWO BEDS | A1 |
| QUEEN | R04 | 2 | 20 | 8500 | QUEEN SIZED BED | A1 |
| KING | R05 | 2 | 15 | 10000 | KING SIZED BED | A1 |
| TWIN | R06 | 2 | 10 | 3000 | TWO TWIN BEDS | A1 |
| DUPLEX | R07 | 4 | 10 | 8000 | TWO BEDS WITH TWO FLOORS | A1 |

## PriceLog Table:

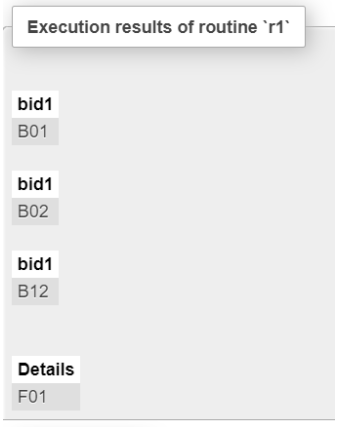
| id | RCID | RCNAME | OldPrice | NewPrice | changedAt |
|---|---|---|---|---|---|
| 1 | R01 | SINGLE | 2500 | 1100 | 2020-10-30 09:40:10 |

# 5.7 CURSORS

**5.7.1 This "r1" Cursor Gives The Information of Cutomer's bill id who have used this Facility 'F01' And further we can fetch cutomer's details from their bill Id using payment id.**

```
DELIMITER $$
CREATE PROCEDURE r1 (INOUT Details VARCHAR(1000))
BEGIN
   DECLARE bid1,fid1 TEXT;
   DECLARE exit_loop BOOLEAN DEFAULT FALSE;
   DECLARE bid_cursor CURSOR FOR SELECT BILL_ID,FID FROM bill;
   DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop=TRUE;
   OPEN bid_cursor;
   bid_loop: LOOP
         FETCH FROM bid_cursor INTO bid1,fid1;
         IF exit_loop THEN
                LEAVE bid_loop;
         END IF;
         IF fid1='F01' THEN
         SELECT bid1;
         END IF;
   END LOOP bid_loop;
   CLOSE bid_cursor;
END$$
DELIMITER ;
```

**OUTPUT:**



# 6.FUTURE ENHANCEMENTS OF THE SYSTEM

- We will design Front-end Design in HTML, CSS, JavaScript.
- We've used localhost/phpMyAdmin to create the Back-end Design and we also can use Python or PHP Languages in Future.
- For security purpose New Entries is done using OTP.
- New facilities can be shared via SMS to customers.
- In future only admin can decide which Facility and Room category to provide.
- We will make database more consistent and We are making this database efficient and easy to implement with huge data capacity. Methods and user data input will be lot easy after the implement of GUI.
- We will also add some extra features so that the users can get answer for their complaints as fast as possible.

# 7.BIBLIOGRAPHY

- For the successful implementation of this Hotel Management project we referred to many websites and books.
- We created the ER Diagram on "erdplus.com" and Schema Diagram on "localhost/phpMyAdmin".
- Mostly we referred the online material for syntax of procedures, triggers, Exception and cursors.

## Reference Book:

Books: Database System Concepts
By: Henry F. Korth

PL/SQL Programming
By: Ivan Bayross

## Reference Website:

1. https://www.w3schools.com/sql
2. https://www.tutorialspoint.com/plsql
3. https://www.mysqltutorial.org/

**Submitted By,**
IT081- Pankhania Aanandi
IT074- Nadpara Princy