Project Report

On

RESORT MANAGEMENT SYSTEM

Developed by

KANSAGARA MITSU NITIN,
DEPARTMENT OF IT, DD UNIVERSITY

KARUKALA TEJASWI REDDY,
DEPARTMENT OF IT, DD UNIVERSITY

Guided by
Internal Guide: Prof. Mukesh M. Goswami
Department of Information Technology,
Faculty of Technology,
DD University

Department of Information Technology,
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad-387001.
November-2020

DHARMSINH DESAI UNIVERSITY

NADIAD - 387001

1. CERTIFICATE

This is to certify that the project entitled **`RESORT MANAGEMENT SYSTEM`**
is a bonafide report of the work carried out by

1) **Ms. KANSAGARA MITSU NITIN**, Student ID No: **18ITUON118**
2) **Ms. KARUKALA TEJASWI REDDY**, Student ID No: **18ITUON112**

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

Prof. Mukesh M. Goswami
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

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

2. ACKNOWLEDGMENT

On the very outset of this report, we would like to extend our sincere & heartfelt obligations towards all the personages who have directly or indirectly helped in this endeavor. Without their active guidance, help, cooperation & encouragement, we wouldn't have made the headway in the project.

We are ineffably indebted to our instructor Mr. Mukesh M. Goswami for his guidance, encouragement and conscientiousness to accomplish this project. We are extremely thankful for his support.

We would also like to express our gratitude to our Head, Mr. Vipul Dabhi along with Dharmsinh Desai University for giving us this opportunity.

Last but not least, we would also like to acknowledge with a deep sense of reverence, our gratitude towards our family members who have always supported us morally.

Any omission in this brief acknowledgement does not mean lack of gratitude.

Thank You,
Kansagara Mitsu Nitin
Karukala Tejaswi Reddy

Tables Of Content

3. SYSTEM OVERVIEW

3.1. Current System

● This is a Resort Management System.
● The main objective of this Resort Management System is to develop and implement a reservation and management system for resort to help the management in making reservations and other transactions needed to be done.
● The customer selects the date and time of the service that they want to use. Also, they get acknowledgement as and when they book or cancel any service.
● The customer can avail different offers available during the time of festivals, centre anniversary or any other specific time of the year.
● The customer can also give their valuable feedback and suggestions without revealing their identity(for confidentiality purpose) so that we can improve our system as per their feedback.
● The main operations of the system are check-in, check-out and reservation management.
● The system will make the employee's job easier and faster with error-free transactions for the good of the business.
● Actors in this proposed system are Admin(receptionist) and the end-users.

3.2. SPECIFIC OBJECTIVES OF PROPOSED SYSTEM

- It accepts reservations.
  The system is capable of storing information for reservations on specified dates and times depending on the customer's availability.
- There is a feature that will identify room availability.
  The system will show the occupied and unoccupied rooms.
- This system will automatically calculate bills.
  Upon the customer's checkout the system automatically calculates the bills depending on the length of the customer's stay. The admin can also add bills for food and extra persons and damages.
- This system will check-in guests.
  The proposed system is capable of checking in reserved and newly arrived guests.
- This system will also check-out guests.
  The system can check-out guests that calculate their bills depending on the length of their stay. It will also calculate the extra charges for food or services being offered.

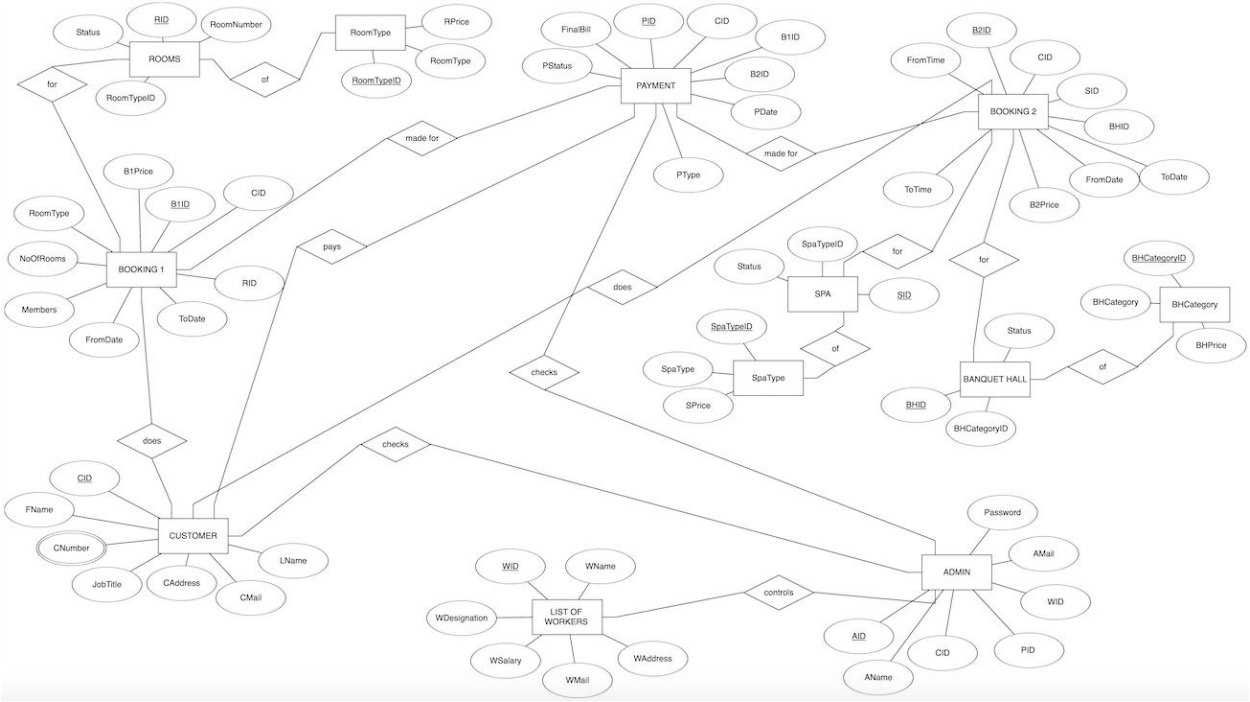
3.3. ADVANTAGES OF THIS SYSTEM

- Saves time for customers in quickly reserving all the facilities of the resort online.
- Incase of on the spot registration, offline bookings can also be done.
- The customer will get notified when the request for the service is placed.
- The customer can pay through their credit card or cash.
- The customers will also get a view about all the services that can be availed.

3.4. DIS-ADVANTAGES OF THIS SYSTEM

- Lack of security.
- Chances of human error.
- Not all features can be accessed by end users.
- Trust issues of admin and end users.
- Requires internet access.
- Failure of machine or servers.
- Heavy traffic leads to failure or long wait issues.

4. ER DIAGRAM FOR RESORT MANAGEMENT SYSTEM



4.1 LIST OF ENTITIES

1. CUSTOMER
2. ADMIN
3. BOOKING 1
4. BOOKING 2
5. ROOMS
6. ROOM TYPE
7. SPA
8. SPA TYPE
9. BANQUET HALL
10. BH CATEGORY
11. LIST OF WORKERS
12. PAYMENT

5. DATA DICTIONARY

CUSTOMER

Column	Type	Null	Description	Link to
CID	Auto increment	Not null	Primary key	
FName	Varchar(20)	Not null		
LName	Varchar(20)	Not null		
CMail	Varchar(40)	Not null		
CAddress	Varchar(45)	Not null		
JobTitle	Varchar(10)	Not null		
CNumber	Decimal(10,0)	Not null	Multivalued	

ADMIN

Column	Type	Null	Description	Link to
AID	Auto increment	Not null	Primary key	
AName	Varchar(45)	Not null		
Password	Varchar(20)	Not null		
AMail	Varchar(40)	Not null		
CID	Int		Foreign key	CUSTOMER
PID	Int		Foreign key	PAYMENT
WID	Int		Foreign key	LIST OF WORKERS

BOOKING 1

Column	Type	Null	Description	Link to
B1ID	Auto increment	Not null	Primary key	
CID	Int	Not null	Foreign key	CUSTOMER
RID	Int	Not null	Foreign key	ROOMS
FromDate	Date	Not null		
ToDate	Date	Not null		
Members	Int	Not null		
NoOfRooms	Int	Not null		
B1Price	Decimal(10,2)	Not null		

BOOKING 2

Column	Type	Null	Description	Link to
B2ID	Auto increment	Not null	Primary key	
CID	Int	Not null	Foreign key	CUSTOMER
SID	Int		Foreign key	SPA
BHID	Int		Foreign key	BANQUET HALL
FromDate	Date	Not null		
ToDate	Date	Not null		
FromTime	Time	Not null		
ToTime	Time	Not null		
B2Price	Decimal(10,2)	Not null		

LIST OF WORKERS

Column	Type	Null	Description	Link to
WID	Auto increment	Not null	Primary key	
WName	Varchar(30)	Not null		
WAddress	Varchar(40)	Not null		
WMail	Varchar(40)	Not null		
WSalary	Decimal	Not null		
WDesignation	Varchar(20)	Not null		

ROOM TYPE

Column	Type	Null	Description	Link to
RoomTypeID	Char	Not null	Primary key	
RoomType	Varchar(10)	Not null		
RPrice	Decimal	Not null		

ROOMS

Column	Type	Null	Description	Link to
RID	Auto increment	Not null	Primary key	
RoomTypeID	Char	Not null	Foreign key	ROOM TYPE
RoomNumber	Int	Not null		
Status	Tiny Int(1)	Not null		

BH CATEGORY

Column	Type	Null	Description	Link to
BHCategoryID	Char	Not null	Primary key	
BHCategory	varchar(20)	Not null		
BHPrice	Decimal(10,2)	Not null		

BANQUET HALL

Column	Type	Null	Description	Link to
BHID	Auto increment	Not null	Primary key	
BHCategoryID	Char	Not null	Foreign Key	BH CATEGORY
Status	TinyInt(1)	Not null		

SPA TYPE

Column	Type	Null	Description	Link to
SpaTypeID	Char	Not null	Primary key	
SpaType	Varchar(10)	Not null		
SPrice	Decimal(10,2)	Not null		

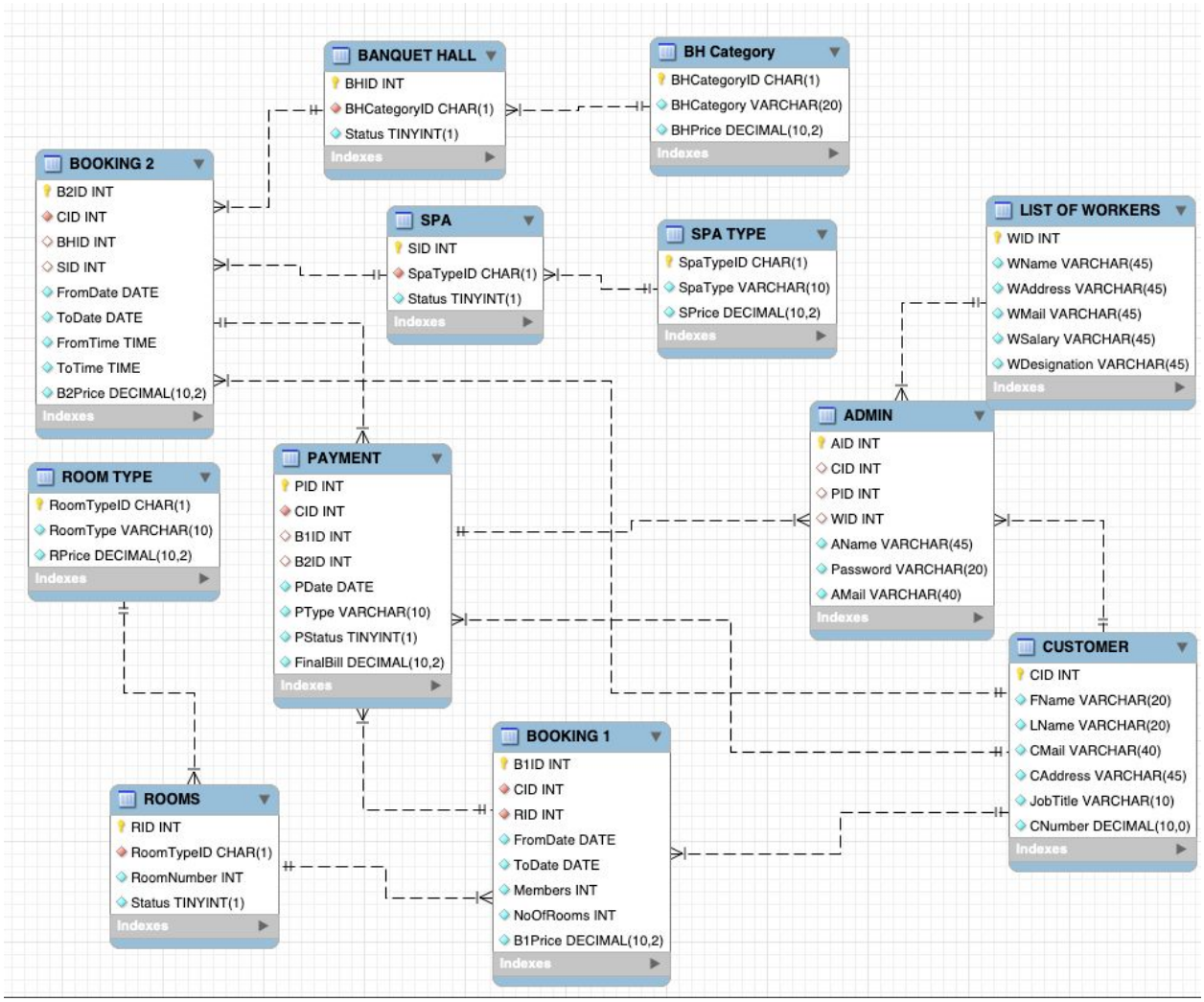
SPA

Column	Type	Null	Description	Link to
SID	Auto increment	Not null	Primary key	
SpaTypeID	Char	Not null	Foreign key	SPA TYPE
Status	Tiny Int(1)	Not null		

PAYMENT

Column	Type	Null	Description	Link to
PID	Auto increment	Not null	Primary key	
CID	Int	Not null	Foreign key	CUSTOMER
B1ID	Int		Foreign key	BOOKING 1
B2ID	Int		Foreign key	BOOKING 2
PDate	Date	Not null		
PType	Varchar(10)	Not null		
PStatus	Tiny Int(1)	Not null		
FinalBill	Decimal(10,2)	Not null		

6. SCHEMA DIAGRAM



7. DATABASE IMPLEMENTATION

7.1CREATING TABLE

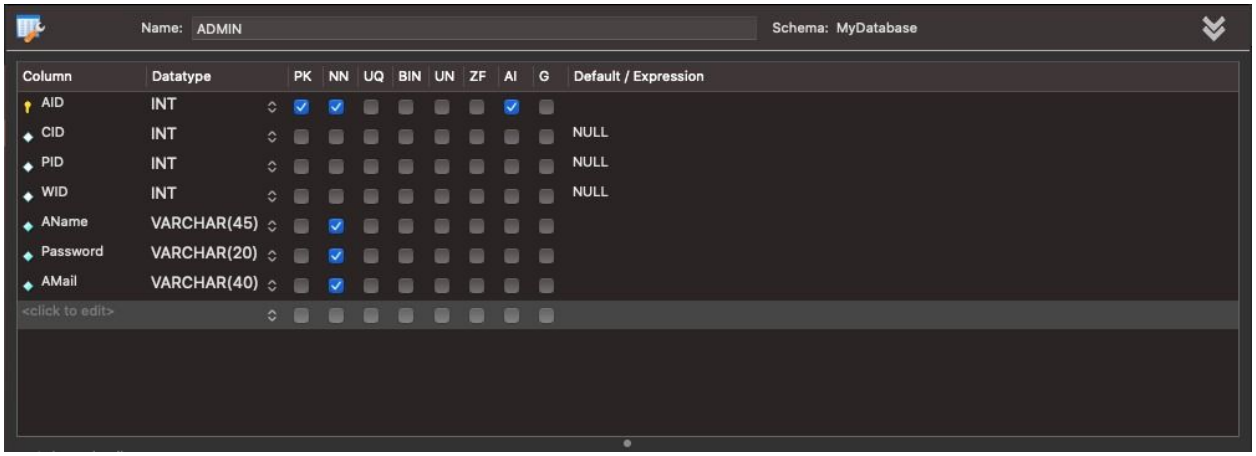
1. Customer

```
CREATE TABLE `CUSTOMER` (
        `CID` INT NOT NULL AUTO_INCREMENT,
        `FName` VARCHAR(20) NOT NULL,
        `LName` VARCHAR(20) NOT NULL,
        `CMail` VARCHAR(40) NOT NULL,
        `CAddress` VARCHAR(45) NOT NULL,
        `JobTitle` VARCHAR(10) NOT NULL
        `CNumber` DECIMAL(10,0) NOT NULL,
        PRIMARY KEY (`CID`)
);
```

2. Admin

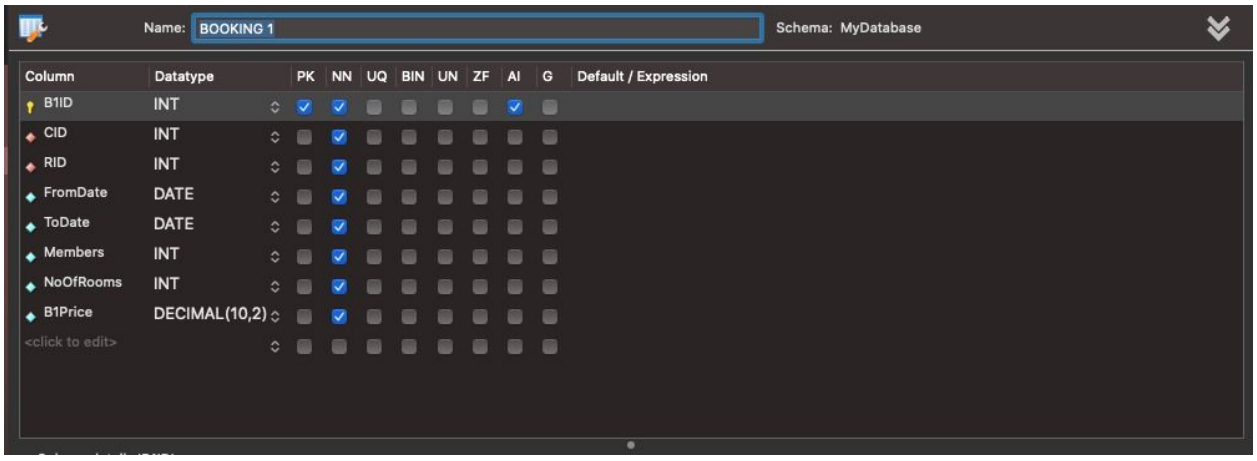
```
CREATE TABLE `ADMIN` (
        `AID` INT NOT NULL AUTO_INCREMENT,
        `CID` INT NULL,
        `PID` INT NULL,
        `WID` INT NULL,
        `AName` VARCHAR(45) NOT NULL,
        `Password` VARCHAR(20) NOT NULL,
        `AMail` VARCHAR(40) NOT NULL,
        PRIMARY KEY (`AID`)
```

);



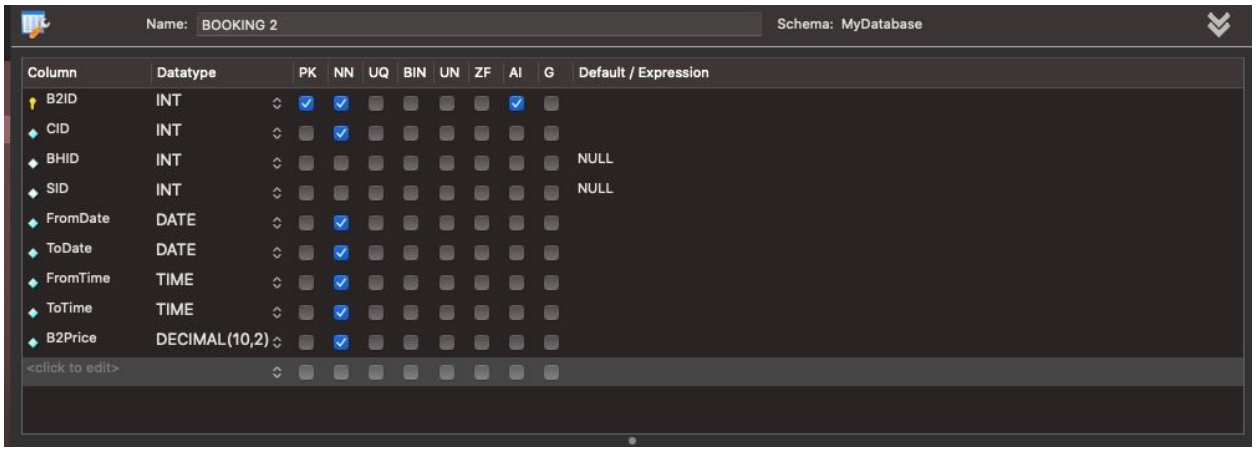
3. Booking1

CREATE TABLE `BOOKING 1` (
`B1ID` INT NOT NULL AUTO_INCREMENT,
`CID` INT NOT NULL,
`RID` int NOT NULL,
`FromDate` DATE NOT NULL,
`ToDate` DATE NOT NULL,
`Members` INT NOT NULL,
`NoOfRooms` INT NOT NULL,
`B1Price` DECIMAL(10,2) NOT NULL, PRIMARY KEY (`B1ID`)
);



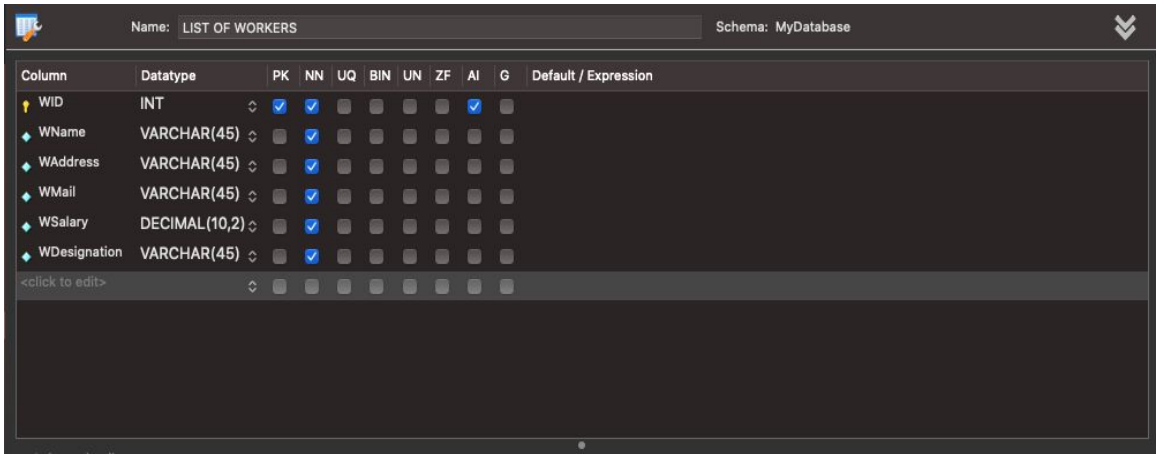
4. Booking 2

```
CREATE TABLE `BOOKING 2` (
        `B2ID` INT NOT NULL AUTO_INCREMENT,
        `CID` INT REFERENCES `CUSTOMER`(`CID`),
        `BHID` INT, `SID` ,
        `FromDate` DATE NOT NULL,
        `ToDate` DATE NOT NULL,
        `FromTime` TIME NOT NULL,
        `ToTime` TIME NOT NULL,
        `B2Price` DECIMAL(10,2) NOT NULL,
        PRIMARY KEY (`B2ID`)
);
```



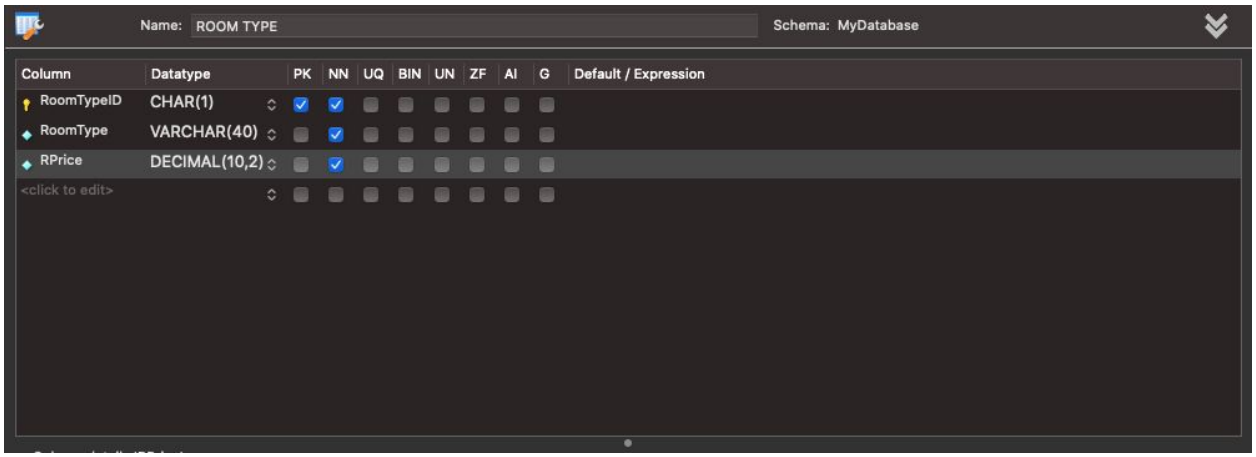
5. List of Workers

```
CREATE TABLE `LIST OF WORKERS` (
        `WID` INT NOT NULL AUTO_INCREMENT,
        `WName` VARCHAR(45) NOT NULL,
        `WAddress` VARCHAR(45) NOT NULL,
        `WMail` VARCHAR(45) NOT NULL,
        `WSalary` DECIMAL(10,2) NOT NULL,
        `WDesignation` VARCHAR(45) NOT NULL,
        PRIMARY KEY (`WID`)
);
```



6. Room Type

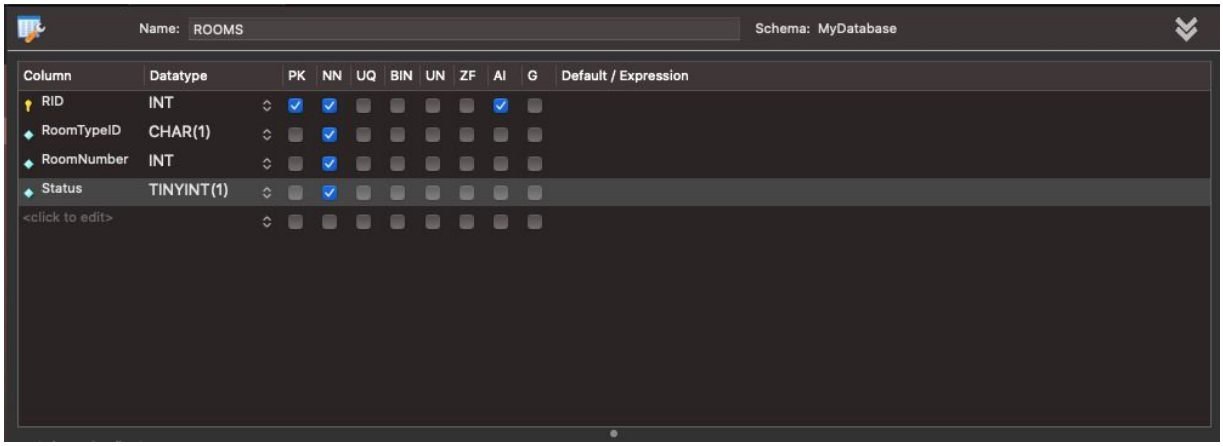
```
CREATE TABLE `ROOM TYPE` (
        `RoomTypeID` CHAR NOT NULL,
        `RoomType` VARCHAR() NOT NULL,
        `RPrice` DECIMAL(10,2) NOT NULL,
        PRIMARY KEY (`RoomTypeID`)
);
```



7. Rooms

```
CREATE TABLE `ROOMS` (
        `RID` INT NOT NULL AUTO_INCREMENT,
```

```
    `RoomTypeID` CHAR NOT NULL,
    `RoomNumber` INT NOT NULL,
    `Status` TINYINT(1) NOT NULL,
    PRIMARY KEY (`RID`)
);
```



8. Banquet Hall Category

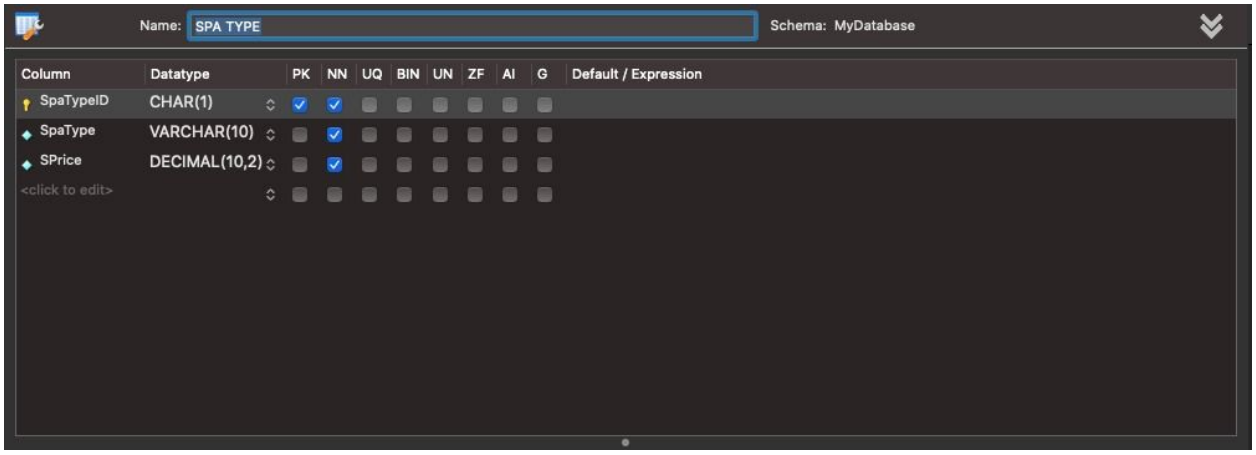
```
CREATE TABLE `BH Category` (
    `BHCategoryID` CHAR NOT NULL,
    `BHCategory` VARCHAR(20) NOT NULL,
    `BHPrice` DECIMAL(10,2) NOT NULL,
    PRIMARY KEY (`BHCategoryID`)
);
```

9. Banquet Hall

```
CREATE TABLE `BANQUET HALL` (
        `BHID` INT NOT NULL AUTO_INCREMENT,
        `BHCategoryID` CHAR NOT NULL,
        `Status` TINYINT(1) NOT NULL, PRIMARY KEY (`BHID`)
);
```

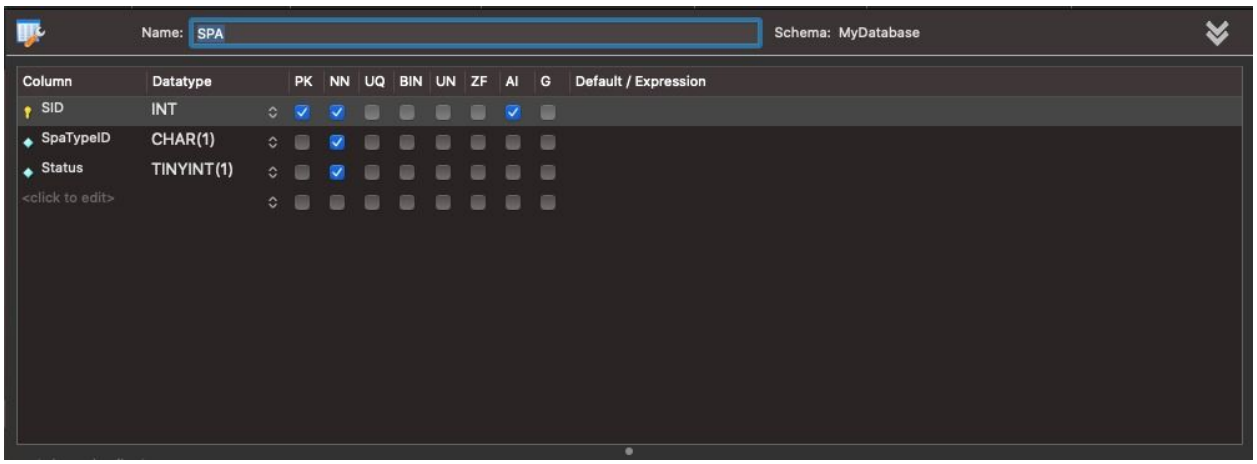
10. Spa Type

```
CREATE TABLE `SPA TYPE` (
        `SpaTypeID` CHAR NOT NULL,
        `SpaType` VARCHAR(10) NOT NULL,
        `SPrice` DECIMAL(10,2) NOT NULL, PRIMARY KEY (`SpaTypeID`)
);
```



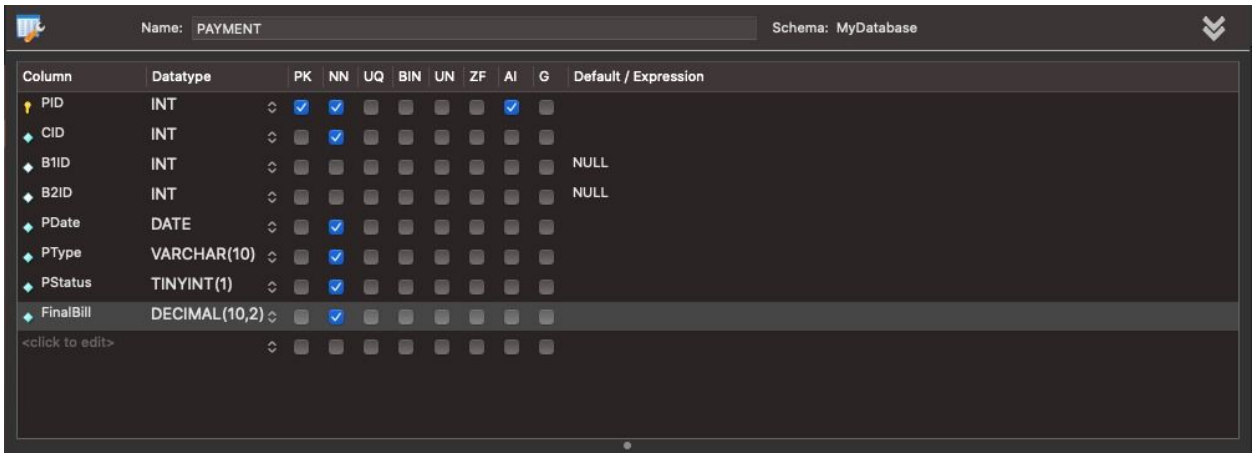
11.Spa

```
CREATE TABLE `SPA` (
        `SID` INT NOT NULL AUTO_INCREMENT,
        `SpaTypeID` CHAR NOT NULL,
        `Status` TINYINT(1) NOT NULL,  PRIMARY KEY (`SID`)
);
```



12. Payment

```
CREATE TABLE `PAYMENT` (
        `PID` INT NOT NULL AUTO_INCREMENT,
        `CID` INT NOT NULL,
        `B1ID` INT NULL,
        `B2ID` INT NULL,
        `PDate` DATE NOT NULL,
        `PType` VARCHAR(10) NOT NULL,
        `PStatus` TINYINT(1) NOT NULL,
        `FinalBill` DECIMAL(10,2) NOT NULL,  PRIMARY KEY (`PID`)
);
```



7.2 INSERTING VALUES IN TABLES

1. CUSTOMER

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('1', 'Ashish ', 'Patel', 'apatel@gmail.com', '201,Palitirth Flats,Subhanpura,Vadodara', 'Manager', '982577830');

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('2', 'Ramesh', 'Khanna', 'rkhanna@gmail.com', '1 Madhuban society , Anand', 'Manager', '9854983222');

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('3', 'Meera', 'Saxena', 'meera123@gmail.com', '23-Brindavan Apt , Panvel,Mumbai', 'Designer', '9835566270');

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('4', 'John', 'Mathews', 'mathewsj@gmail.com', '05-Dhiraj Apt ,Pune', 'Analyst', '8935577829');

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('5', 'Varun', 'Sandesh', 'vsandesh@gmail.com', '27 Jeevan society ,Hyderabad', 'Teacher', '6763322108');

INSERT INTO `CUSTOMER` (`CID`, `FName`, `LName`, `CMail`, `CAddress`, `JobTitle`, `CNumber`) VALUES ('6', 'Ayush', 'Miharia', 'aymiharia@gmail.com', '83 Cole Apt , Andheri ,Mumbai', 'Doctor', '9362201560');

	CID	FName	LNa...	CMail	CAddress	JobTitle	CNumber
	1	Ashish	Patel	apatel@gmail.com	201,Palitirth Flats,Subhanpura,Vadodara	Manager	982577830
	2	Ramesh	Kha...	rkhanna@gmail.com	1 Madhuban society , Anand	Manager	9854983222
	3	Meera	Sax...	meera123@gmail.com	23-Brindavan Apt , Panvel,Mumbai	Designer	9835566270
	4	John	Mat...	mathewsj@gmail.com	05-Dhiraj Apt ,Pune	Analyst	8935577829
	5	Varun	San...	vsandesh@gmail.com	27 Jeevan society ,Hyderabad	Teacher	6763322108
▶	6	Ayush	Mih...	aymiharia@gmail.com	83 Cole Apt , Andheri ,Mumbai	Doctor	9362201560

2. LIST OF WORKERS

INSERT INTO `LIST OF WORKERS` (`WID`, `WName`, `WAddress`, `WMail`, `WSalary`, `WDesignation`) VALUES ('1', 'Monal Sharma', '29 Twinkle Appt ,Anand', 'msharma@gmail.com', '20000', 'Cleaning Staff');

INSERT INTO `LIST OF WORKERS` (`WID`, `WName`, `WAddress`, `WMail`, `WSalary`, `WDesignation`) VALUES ('2', 'Abhijeet Mathur', '3 Madhur society,Ahmedabad', 'abhim@gmail.com', '25000', 'Receptionist');

INSERT INTO `LIST OF WORKERS` (`WID`, `WName`, `WAddress`, `WMail`, `WSalary`, `WDesignation`) VALUES ('3', 'Mohan Lal', '21 Nyay Appt ,Anand', 'mlal@gmail.com', '15000', 'Peon');

INSERT INTO `LIST OF WORKERS` (`WID`, `WName`, `WAddress`, `WMail`, `WSalary`, `WDesignation`) VALUES ('4', 'Shefali Shiva', '90 Blumoon Bunglows , Vadodara', 'Sshiva@gmail.com', '40000', 'Manager');

INSERT INTO `LIST OF WORKERS` (`WID`, `WName`, `WAddress`, `WMail`, `WSalary`, `WDesignation`) VALUES ('5', 'Ranbir Rathod', '1 Lotus Appt , Ahmedabad', 'rrathod@gmail.com', '35000', 'Head Cook');

WID	WName	WAddress	WMail	WSalary	WDesignation
1	Monal Sharma	29 Twinkle Appt ,Anand	msharma@gmail.com	20000.00	Cleaning Staff
2	Abhijeet Mathur	3 Madhur society,Ahmedabad	abhim@gmail.com	25000.00	Receptionist
3	Mohan Lal	21 Nyay Appt ,Anand	mlal@gmail.com	15000.00	Peon
4	Shefali Shiva	90 Blumoon Bunglows , Vadodara	Sshiva@gmail.com	40000.00	Manager
5	Ranbir Rathod	1 Lotus Appt , Ahmedabad	rrathod@gmail.com	35000.00	Head Cook

3. Room Type

INSERT INTO `ROOM TYPE` (`RoomTypeID`, `RoomType`, `RPrice`) VALUES ('A', 'Family room', '5000');

INSERT INTO `ROOM TYPE` (`RoomTypeID`, `RoomType`, `RPrice`) VALUES ('B', 'Honeymoon Suite', '7000');

INSERT INTO `ROOM TYPE` (`RoomTypeID`, `RoomType`, `RPrice`) VALUES ('C', 'Master Suit', '6000');

RoomTypeID	RoomType	RPrice
A	Family room	5000.00
B	Honeymoon Suite	7000.00
C	Master Suit	6000.00

4. Rooms

INSERT INTO `ROOMS` (`RID`, `RoomTypeID`, `RoomNumber`, `Status`) VALUES ('1', 'A', '105', '0');

INSERT INTO `ROOMS` (`RID`, `RoomTypeID`, `RoomNumber`, `Status`) VALUES ('2', 'A', '103', '1');

INSERT INTO `ROOMS` (`RID`, `RoomTypeID`, `RoomNumber`, `Status`) VALUES ('3', 'B', '306', '1');

INSERT INTO `ROOMS` (`RID`, `RoomTypeID`, `RoomNumber`, `Status`) VALUES ('4', 'C', '401', '1');

INSERT INTO `ROOMS` (`RID`, `RoomTypeID`, `RoomNumber`, `Status`) VALUES ('5', 'B', '307', '0');

RID	RoomTypeID	RoomNumber	Status
1	A	105	0
2	A	103	1
3	B	306	1
4	C	401	1
5	B	307	0

5. Banquet Hall Category

INSERT INTO `BH Category` (`BHCategoryID`, `BHCategory`, `BHPrice`) VALUES ('A', '< 100', '20000');

INSERT INTO `BH Category` (`BHCategoryID`, `BHCategory`, `BHPrice`) VALUES ('B', '< 250', '35000');

BHCategoryID	BHCategory	BHPrice
A	< 100	20000.00
B	< 250	35000.00

6. Banquet Hall

```
INSERT INTO `BANQUET HALL` (`BHID`, `BHCategoryID`, `Status`) VALUES ('1', 'A', '1');

INSERT INTO `BANQUET HALL` (`BHID`, `BHCategoryID`, `Status`) VALUES ('2', 'A', '1');

INSERT INTO `BANQUET HALL` (`BHID`, `BHCategoryID`, `Status`) VALUES ('3', 'B', '1');

INSERT INTO `BANQUET HALL` (`BHID`, `BHCategoryID`, `Status`) VALUES ('4', 'A', '1');

INSERT INTO `BANQUET HALL` (`BHID`, `BHCategoryID`, `Status`) VALUES ('5', 'B', '1');
```

BHID	BHCategoryID	Status
1	A	1
2	A	1
3	B	1
4	A	1
5	B	1

7. SPA TYPE

INSERT INTO `SPA TYPE` (`SpaTypeID`, `SpaType`, `SPrice`) VALUES ('A', 'Pamper Spa', '2500');

INSERT INTO `SPA TYPE` (`SpaTypeID`, `SpaType`, `SPrice`) VALUES ('B', 'Medi Spa', '3500');

INSERT INTO `SPA TYPE` (`SpaTypeID`, `SpaType`, `SPrice`) VALUES ('C', 'Day Spa', '2000');

INSERT INTO `SPA TYPE` (`SpaTypeID`, `SpaType`, `SPrice`) VALUES ('D', 'Stay Spa', '1500')

SpaTypeID	SpaType	SPrice
A	Pamper Spa	2500.00
B	Medi Spa	3500.00
C	Day Spa	2000.00
D	Stay Spa	1500.00

8. SPA

INSERT INTO `SPA` (`SID`, `SpaTypeID`, `Status`) VALUES ('1', 'A', '1');

INSERT INTO `SPA` (`SID`, `SpaTypeID`, `Status`) VALUES ('2', 'C', '1');

INSERT INTO `SPA` (`SID`, `SpaTypeID`, `Status`) VALUES ('3', 'D', '1');

INSERT INTO `SPA` (`SID`, `SpaTypeID`, `Status`) VALUES ('4', 'B', '1');

INSERT INTO `SPA` (`SID`, `SpaTypeID`, `Status`) VALUES ('5', 'B', '0')

SID	SpaTypeID	Status
1	A	1
2	C	1
3	D	1
4	B	1
5	B	0

9. BOOKING1

INSERT INTO `BOOKING 1` (`B1ID`, `CID`, `RID`, `FromDate`, `ToDate`, `Members`, `NoOfRooms`, `B1Price`) VALUES ('1', '1', '1', '2020-11-20', '2020-11-22', '3', '1', '10000');

INSERT INTO `BOOKING 1` (`B1ID`, `CID`, `RID`, `FromDate`, `ToDate`, `Members`, `NoOfRooms`, `B1Price`) VALUES ('2', '2', '2', '2020-12-15', '2020-12-16', '4', '1', '5000');

INSERT INTO `BOOKING 1` (`B1ID`, `CID`, `RID`, `FromDate`, `ToDate`, `Members`, `NoOfRooms`, `B1Price`) VALUES ('3', '3', '3', '2021-01-01', '2021-01-03', '2', '1', '14000');

INSERT INTO `BOOKING 1` (`B1ID`, `CID`, `RID`, `FromDate`, `ToDate`, `Members`, `NoOfRooms`, `B1Price`) VALUES ('4', '4', '4', '2020-12-25', '2020-12-26', '5', '2', '12000');

INSERT INTO `BOOKING 1` (`B1ID`, `CID`, `RID`, `FromDate`, `ToDate`, `Members`, `NoOfRooms`, `B1Price`) VALUES ('5', '5', '5', '2021-02-14', '2021-02-16', '4', '2', '28000');

B1ID	CID	RID	FromDate	ToDate	Members	NoOfRooms	B1Price
1	1	1	2020-11-20	2020-11-22	3	1	10000.00
2	2	2	2020-12-15	2020-12-16	4	1	5000.00
3	3	3	2021-01-01	2021-01-03	2	1	14000.00
4	4	4	2020-12-25	2020-12-26	5	2	12000.00
5	5	5	2021-02-14	2021-02-16	4	2	28000.00

10. Booking2

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `BHID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('1', '1', '1', '1', '2020-11-20', '2020-11-20', '18:00:00', '17:00:00', '22500');

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `BHID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('2', '2', '1', '2', '2020-12-15', '2020-12-15', '14:00:00', '15:00:00', '22000');

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `BHID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('3', '3', '3', '3', '2021-01-02', '2021-01-02', '20:00:00', '21:00:00', '36500');

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `BHID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('4', '5', '4', ' 3', '2021-02-14', '2021-02-16', '12:00:00', '10:00:00', '41500');

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `BHID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('5', '4', '5', '5', '2020-12-25', '2020-12-26', '18:00:00', '19:00:00', '38500');

INSERT INTO `BOOKING 2` (`B2ID`, `CID`, `SID`, `FromDate`, `ToDate`, `FromTime`, `ToTime`, `B2Price`) VALUES ('6', '5', '5', '2021-01-12', '2021-01-12', '12:00:00', '13:00:00', '3500');

B2ID	CID	BHID	SID	FromDate	ToDate	FromTime	ToTime	B2Price
1	1	1	1	2020-11-20	2020-11-20	18:00:00	17:00:00	22500.00
2	2	1	2	2020-12-15	2020-12-15	14:00:00	15:00:00	22000.00
3	3	3	3	2021-01-02	2021-01-02	20:00:00	21:00:00	36500.00
4	5	4	3	2021-02-14	2021-02-16	12:00:00	10:00:00	41500.00
5	4	5	5	2020-12-25	2020-12-26	18:00:00	19:00:00	38500.00
6	5	NULL	5	2021-01-12	2021-01-12	12:00:00	13:00:00	3500.00

11. Payment

INSERT INTO `PAYMENT` (`PID`, `CID`, `B1ID`, `PDate`, `PType`, `PStatus`, `FinalBill`) VALUES ('1', '1', '1', '2020-11-10', 'Cash', '0', '10000');

INSERT INTO `PAYMENT` (`PID`, `CID`, `B2ID`, `PDate`, `PType`, `PStatus`, `FinalBill`) VALUES ('2', '2', '1', '2020-11-21', 'Cash', '1', '22500');

INSERT INTO `PAYMENT` (`PID`, `CID`, `B1ID`, `B2ID`, `PDate`, `PType`, `PStatus`, `FinalBill`) VALUES ('3', '3', '2', '2', '2020-12-16', 'Cash', '1', '27000');

INSERT INTO `PAYMENT` (`PID`, `CID`, `B1ID`, `B2ID`, `PDate`, `PType`, `PStatus`, `FinalBill`) VALUES ('4', '4', '3', '3', '2021-01-03', 'Credit', '0', '50500');

INSERT INTO `PAYMENT` (`PID`, `CID`, `B1ID`, `PDate`, `PType`, `PStatus`, `FinalBill`) VALUES ('5', '5', '5', '2021-02-16', 'Cash', '1', '28000');

PID	CID	B1ID	B2ID	PDate	PType	PStatus	FinalBill
1	1	1	NULL	2020-11-10	Cash	0	10000.00
2	2	NULL	1	2020-11-21	Cash	1	22500.00
3	3	2	2	2020-12-16	Cash	1	27000.00
4	4	3	3	2021-01-03	Credit	0	50500.00
5	5	5	NULL	2021-02-16	Cash	1	28000.00

12. Admin

INSERT INTO `ADMIN` (`AID`, `AName`, `Password`, `AMail`) VALUES ('1', 'Mitsu', 'mitsu@1234', 'mitsuk@gmail.com');

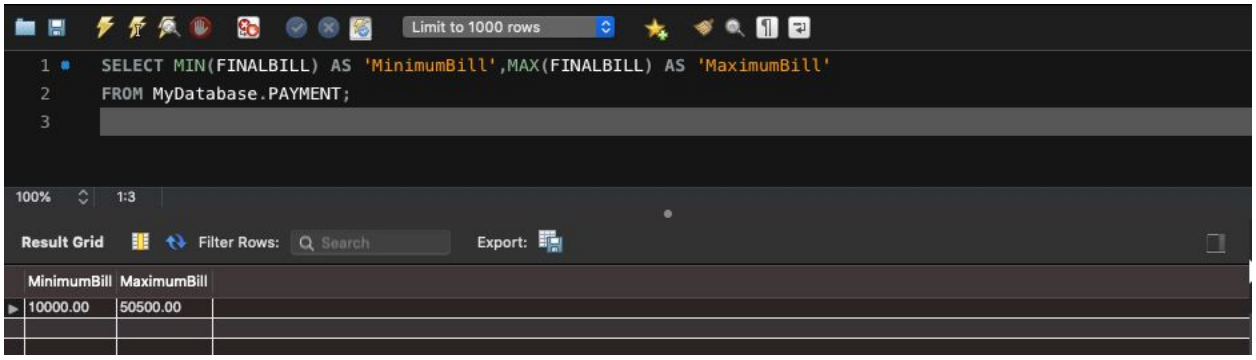
INSERT INTO `ADMIN` (`AID`, `AName`, `Password`, `AMail`) VALUES ('2', 'Tejaswi', 'tejaswi@12', 'tejaswik@gmail.com');

AID	CID	PID	WID	AName	Password	AMail
1	NULL	NULL	NULL	Mitsu	mitsu@1234	mitsuk@gmail.com
2	NULL	NULL	NULL	Tejaswi	tejaswi@12	tejaswik@gmail.com

7.3 QUERIES

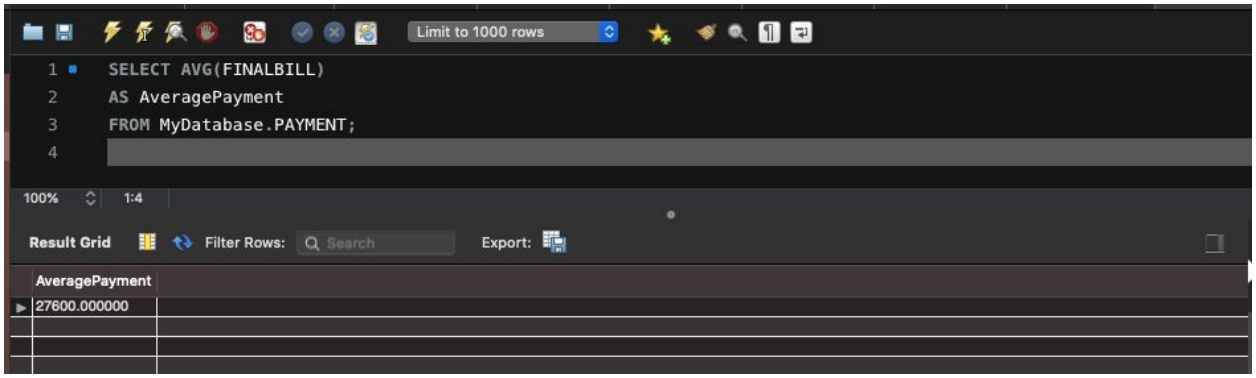
1. **Determine the MINIMUM and the MAXIMUM cost of service from PAYMENT.**

   SELECT MIN(FINALBILL) AS 'MinimumBill',MAX(FINALBILL) AS 'MaximumBill'
   FROM MyDatabase.PAYMENT;



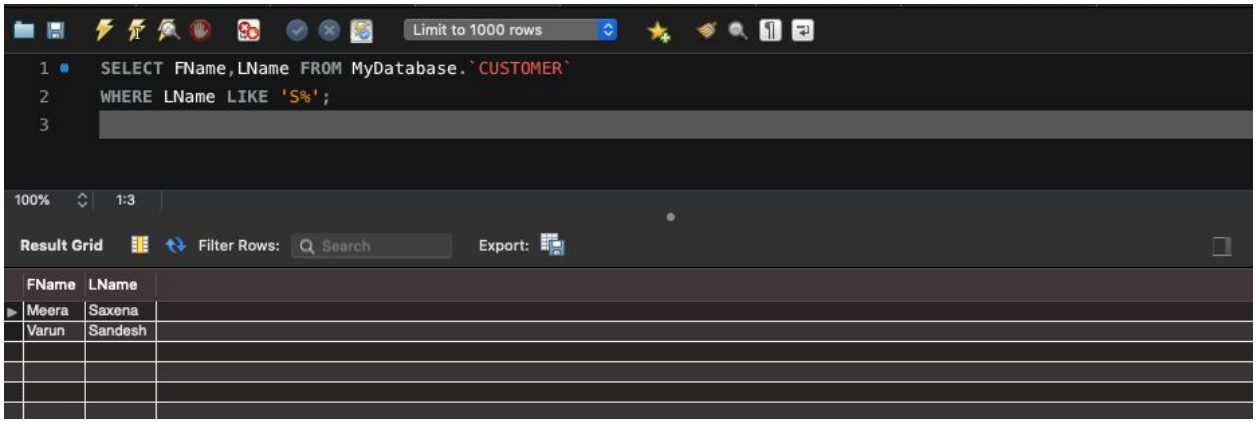
2. **Display the AVERAGE value of the FinalBill from PAYMENT table.**

   SELECT AVG(FINALBILL)
   AS AveragePayment
   FROM MyDatabase.PAYMENT;



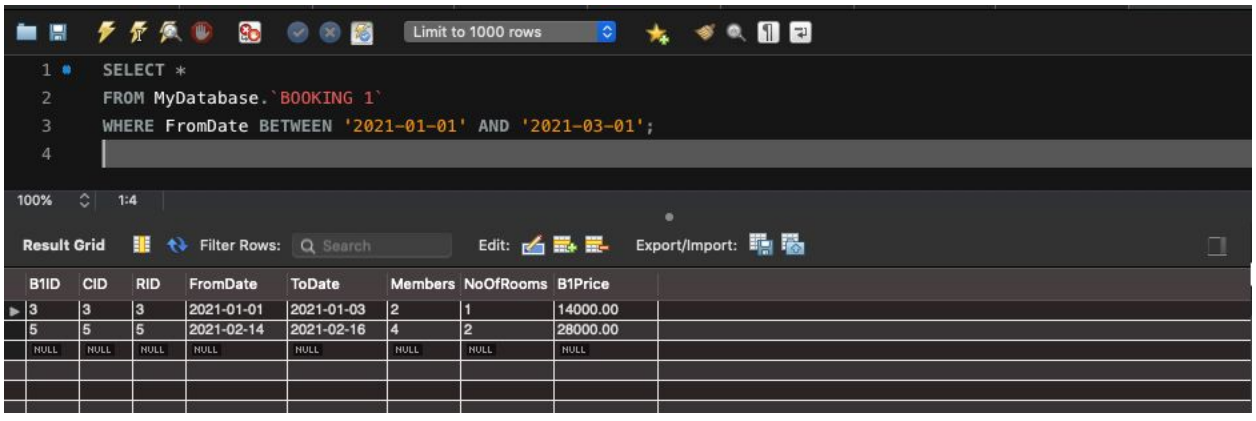
3. **Display the first and last names of CUSTOMERS where last name starts with S.**

```
SELECT FName,LName FROM MyDatabase.`CUSTOMER`
WHERE LName LIKE 'S%' ;
```



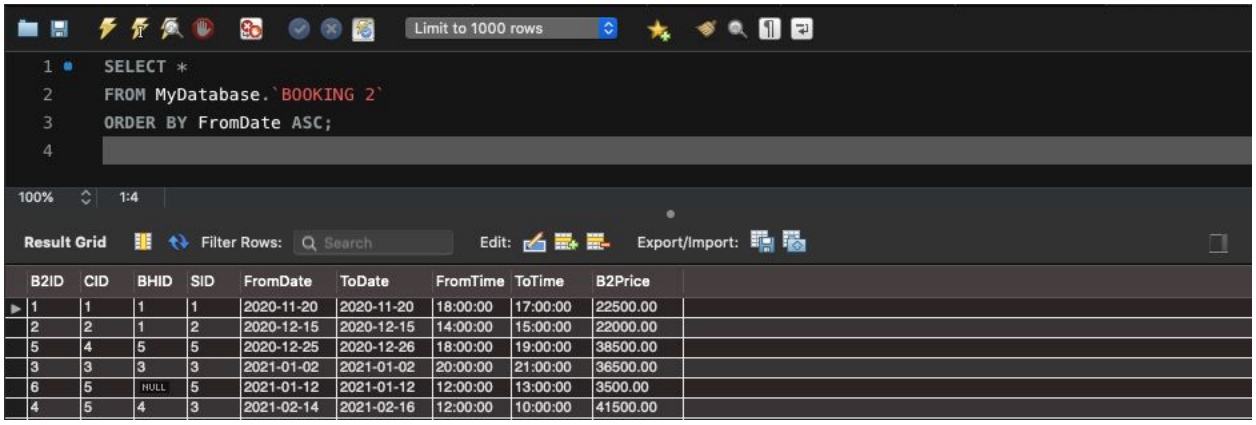
4. **Display the whole record from BOOKING 1 table where the FromDate is between 2021-01-01 and 2021-03-01**

```
SELECT   *
FROM MyDatabase.`BOOKING 1`
WHERE FromDate BETWEEN '2021-01-01' AND '2021-03-01' ;
```



5. Display all the records of BOOKING 2 table in ascending order of FromDate.

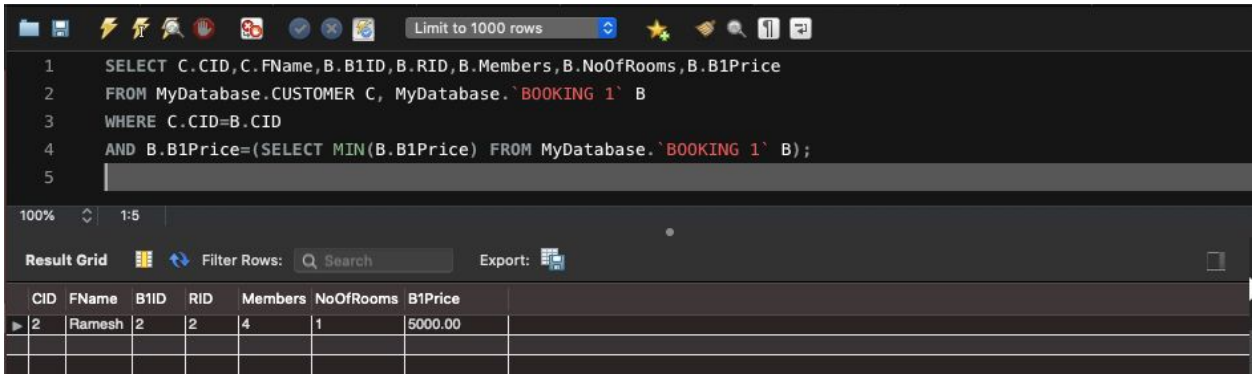
```
SELECT *
FROM MyDatabase.`BOOKING 2`
ORDER BY FromDate ASC;
```



B2ID	CID	BHID	SID	FromDate	ToDate	FromTime	ToTime	B2Price
1	1	1	1	2020-11-20	2020-11-20	18:00:00	17:00:00	22500.00
2	2	1	2	2020-12-15	2020-12-15	14:00:00	15:00:00	22000.00
5	4	5	5	2020-12-25	2020-12-26	18:00:00	19:00:00	38500.00
3	3	3	3	2021-01-02	2021-01-02	20:00:00	21:00:00	36500.00
6	5	NULL	5	2021-01-12	2021-01-12	12:00:00	13:00:00	3500.00
4	5	4	3	2021-02-14	2021-02-16	12:00:00	10:00:00	41500.00

6. Find The Address Of Customer whose Payment Is Minimum.

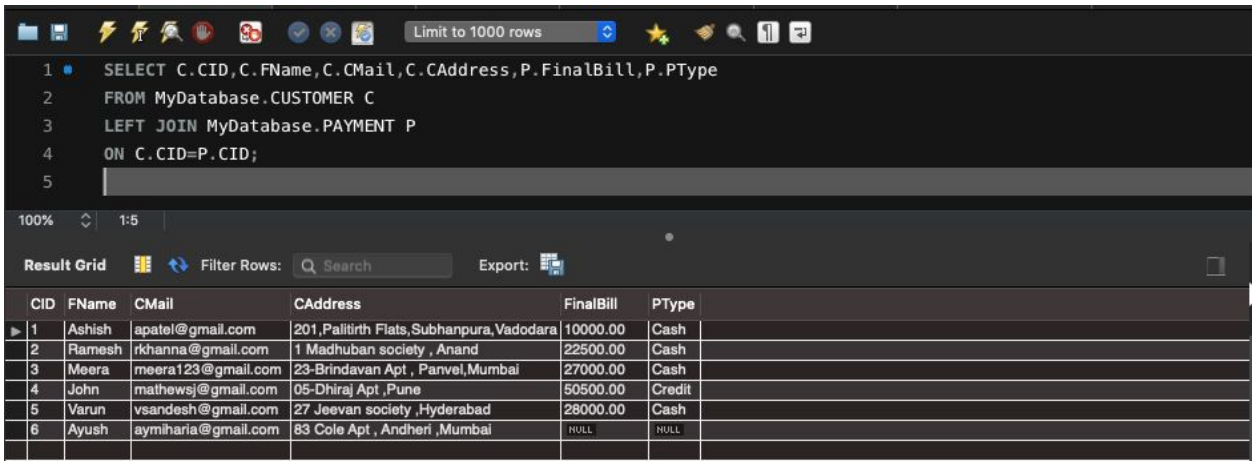
```
SELECT C.CID,C.FName,B.B1ID, B.RID,B.Members,B.NoOfRooms, B.B1Price
FROM MyDatabase.CUSTOMER C, MyDatabase.`BOOKING 1` B
WHERE C.CID = B.CID
AND B.B1Price =(SELECT MIN ( B.B1Price ) FROM MyDatabase.`BOOKING 1` B);
```



CID	FName	B1ID	RID	Members	NoOfRooms	B1Price
2	Ramesh	2	2	4	1	5000.00

7. Use of left join to join CUSTOMER and PAYMENT table.

```
SELECT C.CID,C.FName,C.CMail,C.CAddress,P.FinalBill,P.PType
FROM MyDatabase.CUSTOMER C
LEFT JOIN MyDatabase.PAYMENT P
ON C.CID=P.CID;
```



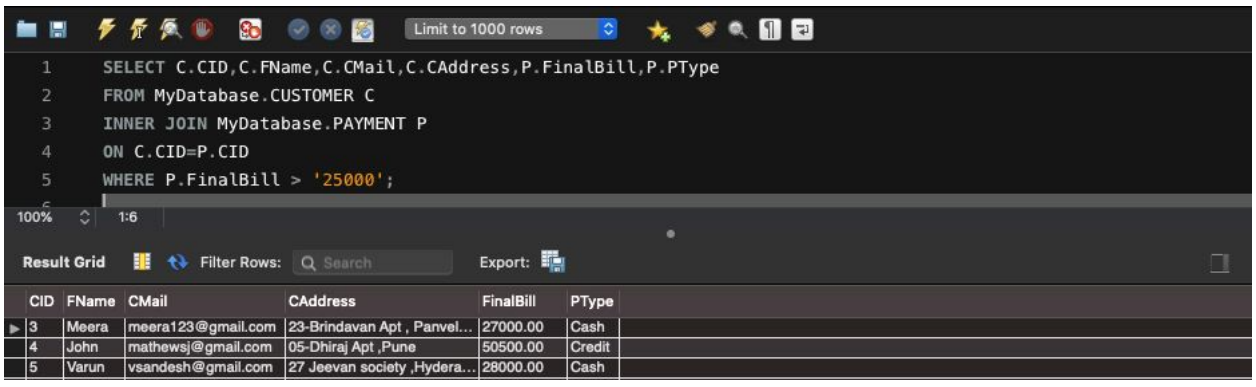
```
SELECT C.CID,C.FName,C.CMail,C.CAddress,P.FinalBill,P.PType
FROM MyDatabase.CUSTOMER C
LEFT JOIN MyDatabase.PAYMENT P
ON C.CID=P.CID;
```

Result Grid

CID	FName	CMail	CAddress	FinalBill	PType
1	Ashish	apatel@gmail.com	201,Palitirth Flats,Subhanpura,Vadodara	10000.00	Cash
2	Ramesh	rkhanna@gmail.com	1 Madhuban society , Anand	22500.00	Cash
3	Meera	meera123@gmail.com	23-Brindavan Apt , Panvel,Mumbai	27000.00	Cash
4	John	mathewsj@gmail.com	05-Dhiraj Apt ,Pune	50500.00	Credit
5	Varun	vsandesh@gmail.com	27 Jeevan society ,Hyderabad	28000.00	Cash
6	Ayush	aymiharia@gmail.com	83 Cole Apt , Andheri ,Mumbai	NULL	NULL

8. Use of inner join to join CUSTOMER and PAYMENT table where FinalBill >25000.

```
SELECT C.CID,C.FName,C.CMail,C.CAddress,P.FinalBill,P.PType
FROM MyDatabase.CUSTOMER C
INNER JOIN MyDatabase.PAYMENT P
ON C.CID=P.CID
WHERE P.FinalBill > '25000';
```



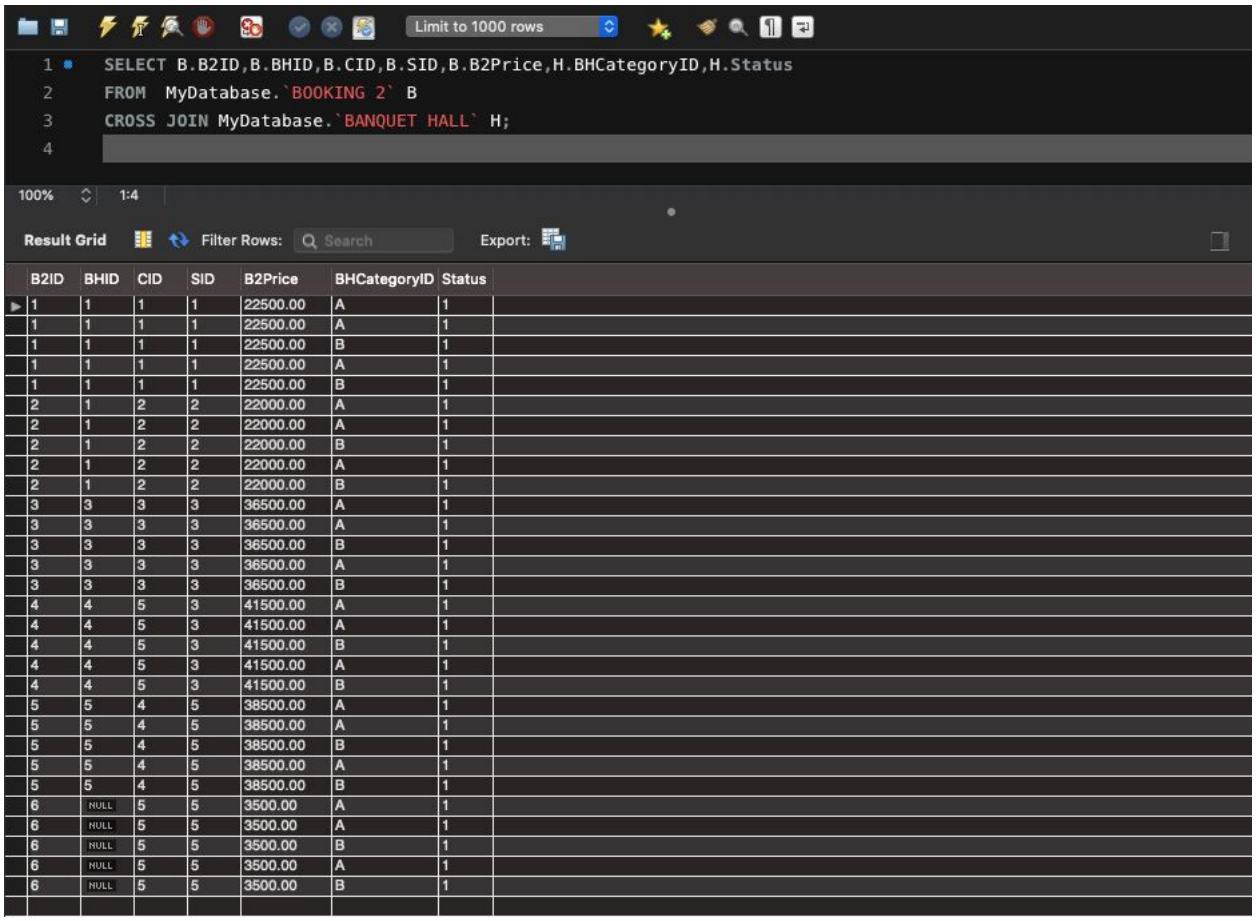
```
SELECT C.CID,C.FName,C.CMail,C.CAddress,P.FinalBill,P.PType
FROM MyDatabase.CUSTOMER C
INNER JOIN MyDatabase.PAYMENT P
ON C.CID=P.CID
WHERE P.FinalBill > '25000';
```

Result Grid

CID	FName	CMail	CAddress	FinalBill	PType
3	Meera	meera123@gmail.com	23-Brindavan Apt , Panvel...	27000.00	Cash
4	John	mathewsj@gmail.com	05-Dhiraj Apt ,Pune	50500.00	Credit
5	Varun	vsandesh@gmail.com	27 Jeevan society ,Hydera...	28000.00	Cash

9. Use of cross join to join BOOKING 2 and BANQUET HALL table.

```
SELECT B.B2ID,B.BHID,B.CID,B.SID,B.B2Price,H.BHCategoryID,H.Status
FROM  MyDatabase.`BOOKING 2` B
CROSS JOIN MyDatabase.`BANQUET HALL` H;
```



```
1  SELECT B.B2ID,B.BHID,B.CID,B.SID,B.B2Price,H.BHCategoryID,H.Status
2  FROM  MyDatabase.`BOOKING 2` B
3  CROSS JOIN MyDatabase.`BANQUET HALL` H;
4
```

Limit to 1000 rows

100% 1:4

Result Grid Filter Rows: Search Export:

B2ID	BHID	CID	SID	B2Price	BHCategoryID	Status
1	1	1	1	22500.00	A	1
1	1	1	1	22500.00	A	1
1	1	1	1	22500.00	B	1
1	1	1	1	22500.00	A	1
1	1	1	1	22500.00	B	1
2	1	2	2	22000.00	A	1
2	1	2	2	22000.00	A	1
2	1	2	2	22000.00	B	1
2	1	2	2	22000.00	A	1
2	1	2	2	22000.00	B	1
3	3	3	3	36500.00	A	1
3	3	3	3	36500.00	A	1
3	3	3	3	36500.00	B	1
3	3	3	3	36500.00	A	1
3	3	3	3	36500.00	B	1
4	4	5	3	41500.00	A	1
4	4	5	3	41500.00	A	1
4	4	5	3	41500.00	B	1
4	4	5	3	41500.00	A	1
4	4	5	3	41500.00	B	1
5	5	4	5	38500.00	A	1
5	5	4	5	38500.00	A	1
5	5	4	5	38500.00	B	1
5	5	4	5	38500.00	A	1
5	5	4	5	38500.00	B	1
6	NULL	5	5	3500.00	A	1
6	NULL	5	5	3500.00	A	1
6	NULL	5	5	3500.00	B	1
6	NULL	5	5	3500.00	A	1
6	NULL	5	5	3500.00	B	1

10. Display WORKER name whose DESIGNATION is Receptionist.

```
SELECT *
FROM MyDatabase.`LIST OF WORKERS`
WHERE WDesignation = 'Receptionist';
```

```
1  SELECT *
2  FROM MyDatabase.`LIST OF WORKERS`
3  WHERE WDesignation = 'Receptionist';
4
```

100% 1:4

Result Grid | Filter Rows: Search | Edit: | Export/Import:

WID	WName	WAddress	WMail	WSalary	WDesignation
2	Abhijeet Mathur	3 Madhur society,Ahmedabad	abhim@gmail.com	25000.00	Receptionist
NULL	NULL	NULL	NULL	NULL	NULL

11. UPDATE some values from table ROOMS where RoomTypeID is C.

```
UPDATE MyDatabase.ROOMS
SET RoomNumber = '403', Status = '0'
WHERE RoomTypeID = 'C';
SELECT * FROM MyDatabase.ROOMS ;
```

Limit to 1000 rows

```
1  UPDATE MyDatabase.ROOMS
2  SET RoomNumber = '403', Status = '0'
3  WHERE RoomTypeID = 'C';
4  SELECT * FROM MyDatabase.ROOMS ;
5
```

100% 1:5

Result Grid | Filter Rows: Search | Edit: | Export/Import:

RID	RoomTypeID	RoomNumber	Status
1	A	105	0
2	A	103	1
3	B	306	1
4	C	403	0
5	B	307	0

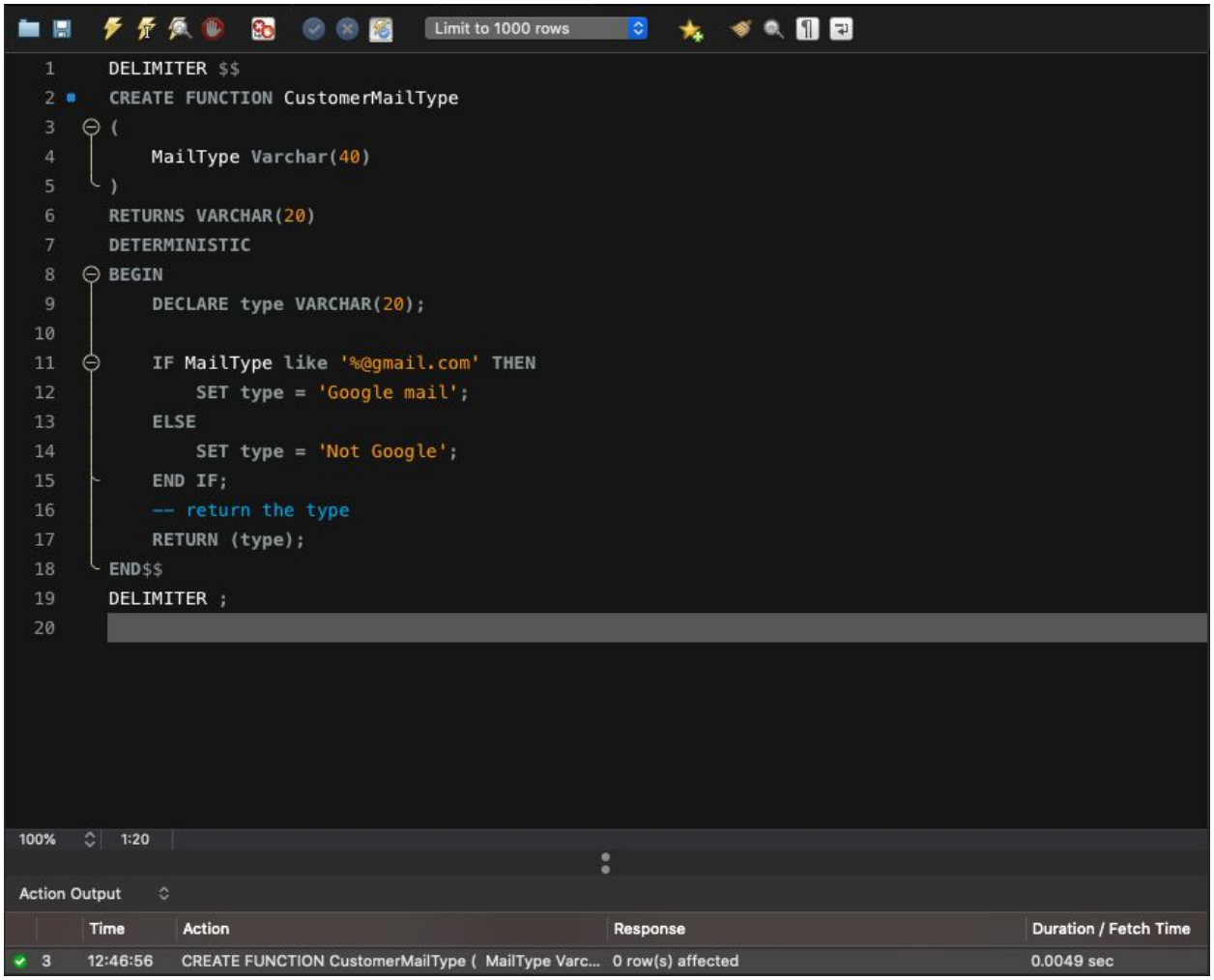
7.4FUNCTION:

1) SHOWS IF THE CUSTOMER HAS GOOGLE ACCOUNT OR NOT

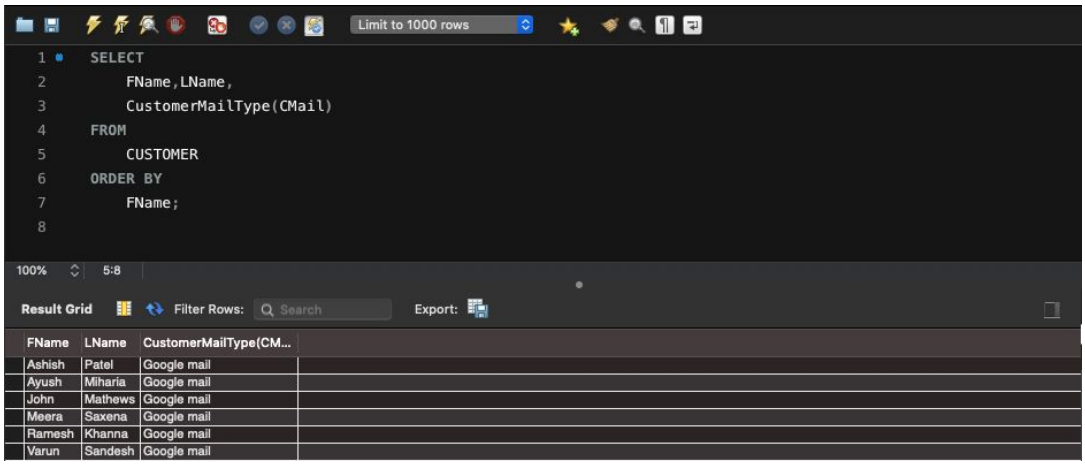
```
DELIMITER $$
CREATE FUNCTION CustomerMailType
(
        MailType Varchar(40)
)
RETURNS VARCHAR(20)
DETERMINISTIC
BEGIN
   DECLARE type VARCHAR(20);

   IF MailType like '%@gmail.com' THEN
                SET type = 'Google mail';
        ELSE
      SET type = 'Not Google';
   END IF;
        -- return the type
        RETURN (type);
END$$
```

DELIMITER ;



SELECT FName, LName, CustomerMailType(CMail)
FROM CUSTOMER
ORDER BY FName;



7.5 PROCEDURE AND EXCEPTION HANDLING

1) INSERTING WORKERS VALUES ONLY IF THE WORKER HAS A GOOGLE MAIL ACCOUNT

```
DELIMITER $$
create procedure workers_Data(IN WName varchar(10),IN WAddress varchar(40),IN
WMail varchar(20),
IN WSalary decimal(10,0),IN WDesignation varchar(20))
begin
DECLARE counter varchar(4);
        select count(WID) into counter from `LIST OF WORKERS`;
        set counter = counter + 1;
        IF WMail like '%@gmail.com' THEN
                INSERT into `LIST OF WORKERS` VALUES
(WName,WAddress,WMail,WSalary,WDesignation);
                SELECT 'YOUR DATA IS INSERTED' AS MESSAGE;
        ELSE
                SELECT 'YOUR Email-ID is not valid' AS ERROR;
        END IF;
END $$
DELIMITER;
```

```
DELIMITER $$
create procedure workers_Data(IN WName varchar(10),IN WAddress varchar(40),IN WMail varchar(20),
IN WSalary decimal(10,0),IN WDesignation varchar(20))
begin
DECLARE counter varchar(4);
        select count(WID) into counter from `LIST OF WORKERS`;
        set counter = counter + 1;
    IF WMail like '%@gmail.com' THEN
        INSERT into `LIST OF WORKERS` VALUES (WName,WAddress,WMail,WSalary,WDesignation);
        SELECT 'YOUR DATA IS INSERTED' AS MESSAGE;
    ELSE
        SELECT 'YOUR Email-ID is not valid' AS ERROR;
    END IF;
END $$
DELIMITER ;
```

```
Action Output
Time | Action | Response | Duration / Fetch Time
3 | 13:38:55 | create procedure workers_Data(IN WName varchar(10),IN WAddress varchar(40),I... | 0 row(s) affected | 0.0046 sec
```

CALL workersData ('Arjun','26-BeeHoles Academy,Surat','arjun@yahoo.com',12000,'Manager');

```
CALL workers_Data ('Arjun','26-BeeHoles Academy,Surat','arjun@yahoo.com',12000,'Manager');
```

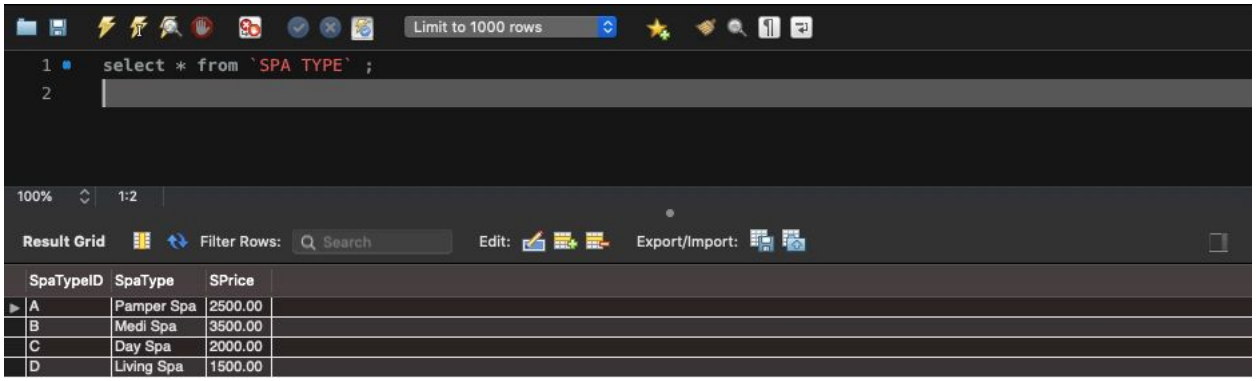
```
Result Grid | Filter Rows: | Export:
ERROR
YOUR Email-ID is not valid
```

7.6 TRIGGERS:

1) UPDATE SPECIFIC SPA TYPE

```
DELIMITER $$
CREATE TRIGGER t1 AFTER UPDATE ON `SPA TYPE` FOR EACH ROW
BEGIN
DECLARE v1 VARCHAR(20);
        SELECT SpaType INTO v1 FROM `SPA TYPE` WHERE
SpaType=NEW.SpaType;
UPDATE `BOOKING 2` SET `SpaType`=v1 WHERE SpaType=OLD.SpaType;
END $$
DELIMITER ;

UPDATE `SPA TYPE` SET `SpaType`="Living Spa" WHERE SpaType="Stay Spa";
```

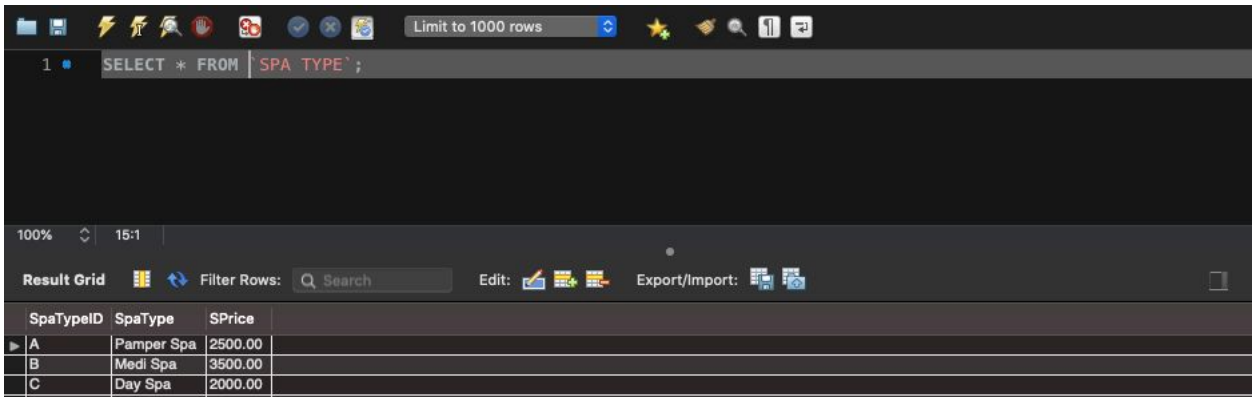


2) DELETE SPECIFIC SPA TYPE FROM SPA TABLE AND BOOKING 2 TABLE

```
DELIMITER $$
CREATE TRIGGER t2 AFTER DELETE ON `SPA TYPE` FOR EACH ROW
BEGIN
DECLARE v1 VARCHAR(20);
 SELECT SpaType INTO v1 FROM `SPA TYPE` WHERE
```

```
SpaType=NEW.SpaType;
DELETE FROM `BOOKING 2` WHERE SpaType=OLD.SpaType;
END $$
DELIMITER ;

DELETE FROM `SPA TYPE` WHERE SpaType="Living Spa";
```

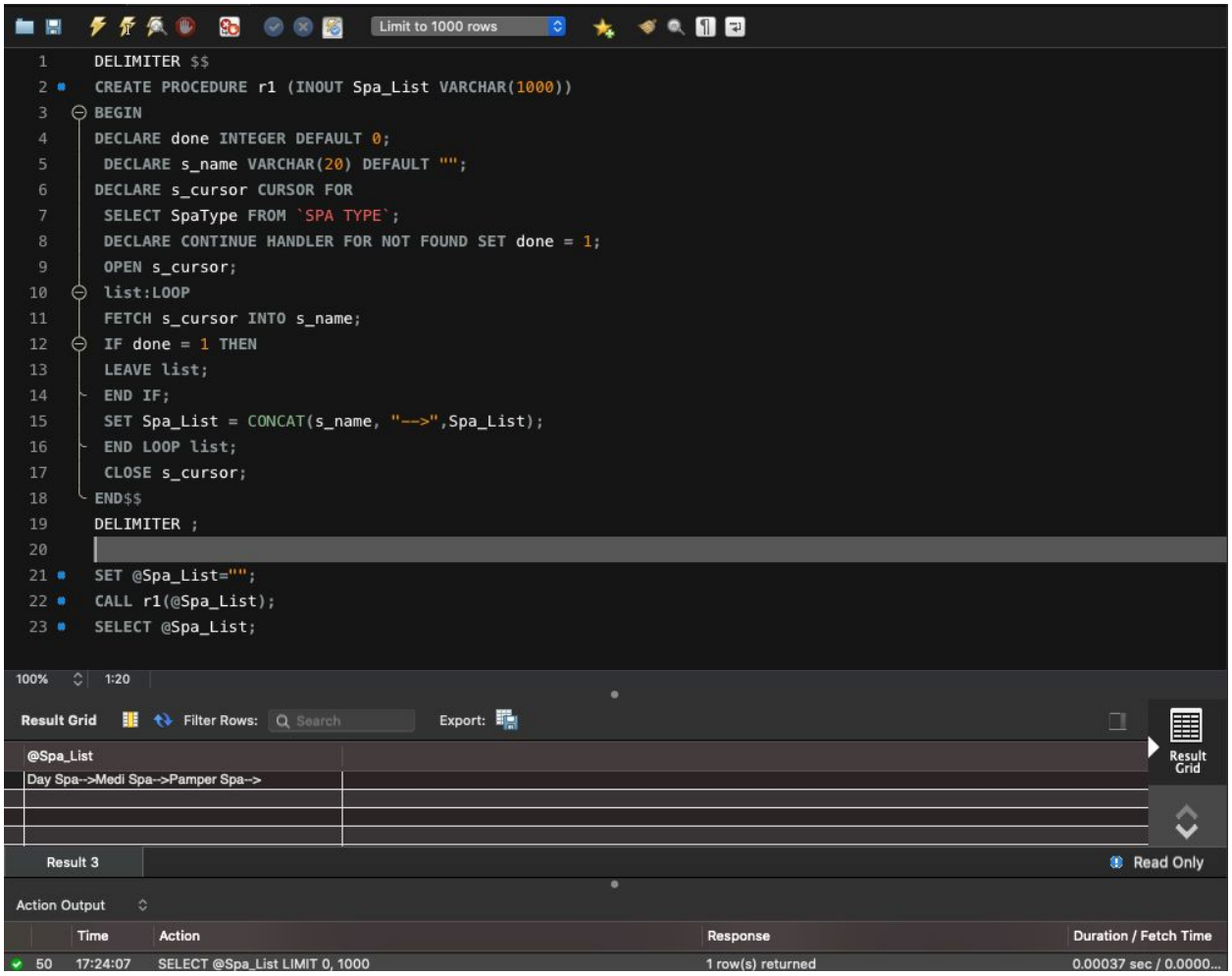


7.7 CURSOR:

1) SHOW THE NAMES OF ALL TYPES OF SPA

```
DELIMITER $$
CREATE PROCEDURE r1 (INOUT Spa_List VARCHAR(1000))
BEGIN
DECLARE done INTEGER DEFAULT 0;
 DECLARE s_name VARCHAR(20) DEFAULT "";
DECLARE s_cursor CURSOR FOR
 SELECT SpaType FROM `SPA TYPE`;
 DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
 OPEN s_cursor;
 list:LOOP
 FETCH s_cursor INTO s_name;
 IF done = 1 THEN
 LEAVE list;
 END IF;
 SET Spa_List = CONCAT(s_name, "-->",Spa_List);
 END LOOP list;
 CLOSE s_cursor;
END$$
DELIMITER ;

SET @Spa_List="";
CALL r1(@Spa_List);
SELECT @Spa_List;
```


Limit to 1000 rows
DELIMITER $$
CREATE PROCEDURE r1 (INOUT Spa_List VARCHAR(1000))
BEGIN
DECLARE done INTEGER DEFAULT 0;
DECLARE s_name VARCHAR(20) DEFAULT "";
DECLARE s_cursor CURSOR FOR
SELECT SpaType FROM `SPA TYPE`;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
OPEN s_cursor;
list:LOOP
FETCH s_cursor INTO s_name;
IF done = 1 THEN
LEAVE list;
END IF;
SET Spa_List = CONCAT(s_name, "-->",Spa_List);
END LOOP list;
CLOSE s_cursor;
END$$
DELIMITER ;
SET @Spa_List="";
CALL r1(@Spa_List);
SELECT @Spa_List;
100%
1:20
Result Grid
Filter Rows:
Search
Export:
@Spa_List
Day Spa-->Medi Spa-->Pamper Spa-->
Result Grid
Result 3
Read Only
Action Output
Time
Action
Response
Duration / Fetch Time
50
17:24:07
SELECT @Spa_List LIMIT 0, 1000
1 row(s) returned
0.00037 sec / 0.0000...

8. FUTURE ENHANCEMENT OF THE SYSTEM

- We plan on designing the front-end design of this system using Python.
- Keep on changing the UI-Design to make the website look more attractive.
- Making the servers and database more consistent, efficient and easy to implement huge amount of data.
- Transforming this web-based application to mobile application for better portability and making it more reliable.
- User data input, methods and output will be easier after the implementation of GUI.
- For security purposes, new registration can be taken using OTP.

9. BIBLIOGRAPHY

- For the successful implementation of this project we referred to many websites and books.
- The schema was designed by taking various ideas from websites og google.com.
- We created the ER Diagram on ‘erdplus.com’.
- Schema Diagram and further implementation of database is done on MySql Workbench.
- Mostly we referred the online material for syntax of procedures, triggers, Exception and cursors.

- **Reference book:**

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

PL/SQL Programming
-Ivan Bayross

- **Reference Websites:**

https://stackoverflow.com/
https://www.w3schools.com/
https://www.tutorialspoint.com/
http://www.mysqltutorial.org/
https://erdplus.com/