



INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

INFO 2203

Database Programming
Project

Restaurant Management System

1. Khan Nasik Sami 1638153
2. Zian Md Afique Amin 1631005
3. MD Sariful Islam 16266674.
4. Abrar Habib Haque 1636029

1.0 Project Background

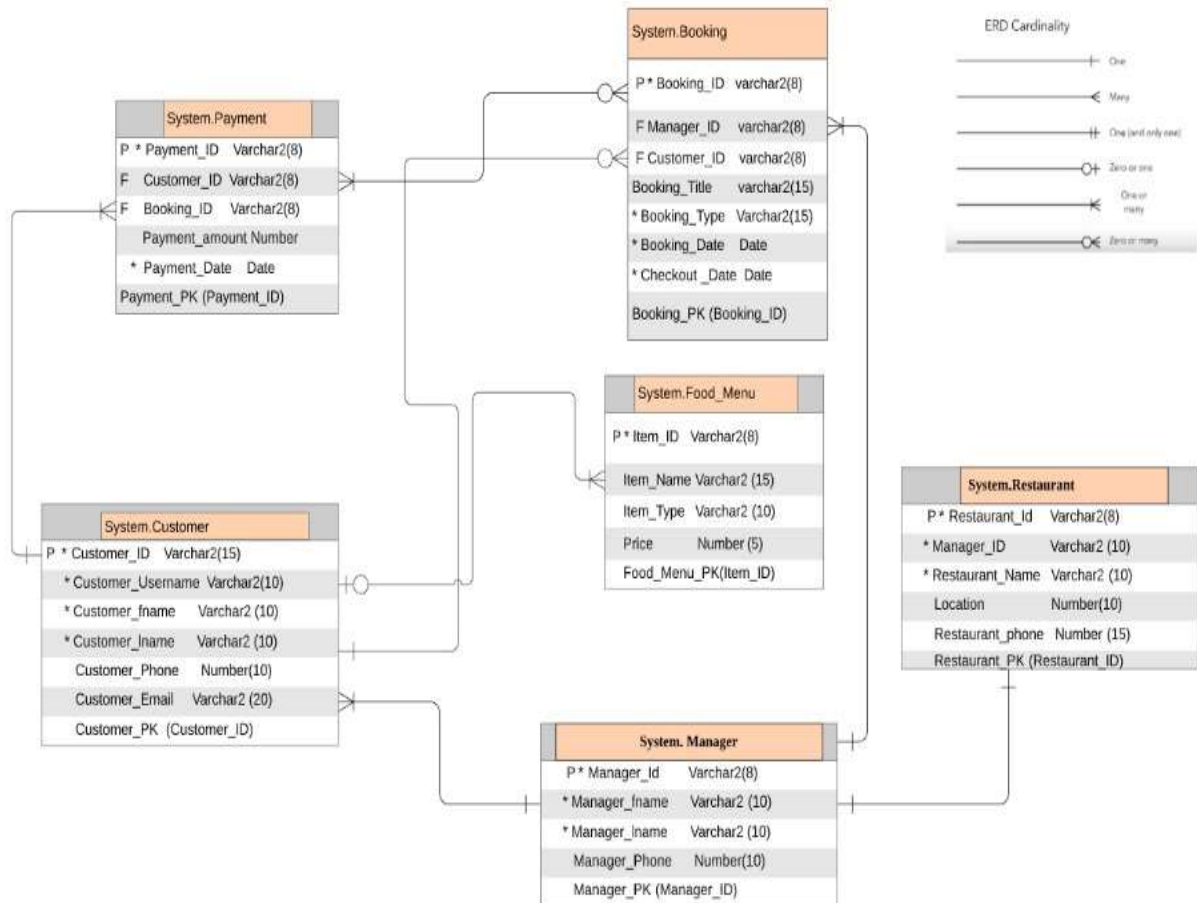
Our team is going to make a system and its title is "Restaurant Management System". In this system, there will be 6 tables. Namely Customer, Manager, Restaurant, Payment, Food_Items and Booking. These tables will have relations and cardinality among themselves. Which is given in the Entity Relation (ER) Diagram below. Basically, we are going to create a system where there will be many restaurants under our supervision in different locations. The menu is similar for all of them. The customer can book his/her order and there will be Eat In and Take Away options. We will work as the bridge between the customer and restaurants. Something similar to what the caterers do. In this project we will populate the tables mentioned above with many relevant values and try to execute various queries and PL sql functions, procedure and cursor. In the later part we will try to make a web application that can interact with the database.

2.0 CONCEPTUAL DATABASE DESIGN

Entity-Relationship Model

Entity Relationship Diagram Example (UML Notation)

naskisami | December 15, 2019



3.0 DATA DICTIONARY FOR ER DIAGRAM

Data Dictionary

Customer

Attribute Name	Data Type	Size	NULL
Customer_Id(PK)	VARCHAR2	8	NO
Customer_Username	VARCHAR2	15	NO
Customer_fname	VARCHAR2	15	NO
Customer_lname	VARCHAR2	10	NO
Customer_Phone	NUMBER	15	YES
Customer_Email	VARCHAR2	20	YES
Item_ID	VARCHAR2	8	NO

Restaurant

Attribute Name	Data Type	Size	NULL
Restaurant_Id(PK)	NUMBER	8	NO
Manager_Id(FK)	NUMBER	8	NO
Restaurant_Name	VARCHAR2	15	NO
Location	VARCHAR2	15	YES
Restaurant_Phone	NUMBER		YES

Booking

Attribute Name	Data Type	Size	NULL
Booking_Id(PK)	NUMBER	8	NO
Manager_Id(FK)	NUMBER	8	NO
Customer_Id(FK)	VARCHAR2	8	NO
Booking_Title	VARCHAR2	15	YES
Booking_Type	VARCHAR2	15	NO
Booking_Date	DATE		NO
Checkout_Date	DATE		NO

Manager

Attribute Name	Data Type	Size	NULL
Manager_Id(PK)	NUMBER	8	NO
Manager_fname	VARCHAR2	10	NO
Manager_lname	VARCHAR2	10	NO
Manager_Phone	NUMBER	15	YES

Payment

Attribute Name	Data Type	Size	NULL
Payment_Id(PK)	NUMBER	8	NO
Customer_Id(FK)	VARCHAR2	8	NO
Booking_Id(FK)	NUMBER	8	NO
Payment_Amount	NUMBER		NO
Payment_Date	DATE		NO

Food_Menu

Attribute Name	Data Type	Size	NULL
Item_Id(PK)	VARCHAR2	8	NO
Item_Name	VARCHAR2	15	YES
Item_Type	VARCHAR2	10	YES
Price	NUMBER	4	YES

User Creation:

SQL> conn

Enter user-name: SYSTEM

Enter password:

Connected.

SQL> conn sys as sysdba;

Enter password:

Connected.

SQL> conn

Enter user-name: hr

Enter password:

ERROR:

ORA-28000: the account is locked

Warning: You are no longer connected to ORACLE.

```
SQL> alter user hr identified by hr account unlock;
```

SP2-0640: Not connected

```
SQL> conn sys as sysdba
```

Enter password:

Connected.

```
SQL> create user project identified by project;
```

User created.

```
SQL> grant dba to assignment1;
```

Grant succeeded.

```
SQL> commit;
```

Commit complete.

4.0 PHYSICAL DATABASE DESIGN

Create and drop table

```
drop table Booking cascade constraints;
```

```
drop table Customer cascade constraints;
```

```
drop table Restaurant cascade constraints;  
drop table Manager cascade constraints;  
drop table Payment cascade constraints;  
drop table Food_Menu cascade constraints;
```

```
create table Customer  
(Customer_Id varchar2(8),  
Customer_Username varchar2(15) not null,  
Customer_fname varchar2(15) not null,  
Customer_lname varchar2(10) not null,  
Customer_Phone number(15),  
Customer_Email varchar2(20),  
Item_ID varchar2(8),  
primary key(Customer_Id),  
foreign key(Item_ID) references Food_Menu(Item_ID));
```

```
create table Restaurant  
(Restaurant_Id number(8),  
Manager_Id number(8),  
Restaurant_Name varchar2(15) not null,  
Location varchar2(15),  
Restaurant_Phone number,  
primary key (Restaurant_Id));
```

```
create table Booking  
(Booking_Id number(8),  
Manager_Id number(8),
```


Customer_Id varchar2(8),
Booking_Title varchar2(15),
Booking_Type varchar2(15) not null,
Booking_Date date not null,
Checkout_Date date not null,
primary key(Booking_Id),
foreign key(Manager_Id) references Manager(Manager_Id),
foreign key(Customer_Id) references Customer(Customer_Id));

create table Manager
(Manager_Id number(8) ,
Manager_fname varchar2(10) not null,
Manager_lname varchar2(10) not null,
Manager_Phone number(15),
primary key(Manager_Id));

create table Payment
(Payment_Id number(8),
Customer_Id varchar2(8),
Booking_Id number(8),
Payment_Amount number,
Payment_Date date not null,
primary key (Payment_Id),
foreign key (Customer_Id) references Customer(Customer_Id),
foreign key (Booking_Id) references Booking(Booking_Id));

```
create table Food_Menu
(Item_ID varchar2(8) not null,
Item_Name varchar2(15) null,
Item_Type varchar2(10),
Price number(4),
primary key (Item_ID));
```

5.0 SCRIPT TO POPULATE TABLE

```
insert into customer values(166777,'Mokles','Md','Sarif',0176322,'jizzi@gmail.com','I02');
insert into customer values(162324,'Zalali','Akash','Ahmad',0276623,'kiz86i@gmail.com','I05');
insert into customer values(162352,'Jib','Nasik','Sami',0176232,'nasikii@gmail.com','I09');
insert into customer values(167865,'Fel','Zian','Ali',0176644,'afiquezian@gmail.com','I10');
insert into customer values(186542,'Saap','Janwar','Sani',0276452,'janwar@gmail.com','I12');
insert into customer values(153766,'Bolod','Tawsif','Mia',0136754,'chor@gmail.com','I12');
insert into customer values(163322,'Makkhi','Amir','Saki',0143652,'sadia@gmail.com','I12');
insert into customer values(122233,'Gadha','Meem','Dota',0123392,'maal@gmail.com','I02');
insert into customer values(164322,'humby','Gerave','Rai',0156652,'ji232ii@gmail.com','I01');
insert into customer values(163221,'Iot','Md','Sarif',0164905,'jiuuh@gmail.com','I05');
```

```
insert into manager values(126777,'Rojaro','shen',112755);
insert into manager values(136625,'Rojaro','Alim',113725);
insert into manager values(146729,'Robert','Smith',236955);
insert into manager values(128762,'Micle','Ubala',446555);
```

```
insert into manager values(134331,'Eka','kappor',114325);
insert into manager values(114231,'Baal','Saad',142455);
insert into manager values(104532,'Hero','Alom',134229);
insert into manager values(136988,'Joya','Ahsan',129325);
insert into manager values(122558,'Abul','Hayat',543200);
insert into manager values(119577,'Murad','Takla',985321);
insert into manager values(164332,'Sanai','Takla',114235);
insert into manager values(124556,'Joya','Ubala',114325);
insert into manager values(744342,'Mujib','Ali',117625);
```

```
insert into booking values(156331,126777,163221,'Indian','Eatin',to_date('25-DEC-19'),to_date('28-DEC-19'));
insert into booking values(166666,114231,164322,'Western','Take Away',to_date('25-DEC-19'),to_date('27-DEC-19'));
insert into booking values(152341,744342,122233,'Malay','Eatin',to_date('22-NOV-19'),to_date('24-NOV-19'));
insert into booking values(296651,164332,163322,'Western','Take Away',to_date('02-DEC-19'),to_date('08-DEC-19'));
insert into booking values(656331,122558,153766,'Indian','Take Away',to_date('24-DEC-19'),to_date('26-DEC-19'));
insert into booking values(199331,128762,186542,'Malay','Eatin',to_date('24-DEC-19'),to_date('25-DEC-19'));
insert into booking values(185331,744342,167865,'Western','Eatin',to_date('02-DEC-17'),to_date('08-DEC-17'));
insert into booking values(136391,126777,162352,'Indian','Take Away',to_date('29-NOV-18'),to_date('30-NOV-18'));
insert into booking values(198331,122558,162324,'Malay','Eatin',to_date('26-NOV-17'),to_date('29-NOV-17'));
insert into booking values(189766,124556,166777,'Western','Take Away',to_date('27-FEB-18'),to_date('01-MAR-18'));
```

insert into booking values(876623,136625,166777,'Indian','Eatin',to_date('28-AUG-18'),to_date('02-SEP-18'));

insert into Restaurant values(987755,126777,'Restaurant Taj','Gombak',655422);
insert into Restaurant values(988772,136625,'Restaurant Mihh','Idaman',655222);
insert into Restaurant values(163265,146729,'Marks','Selangor',152443);
insert into Restaurant values(875542,128762,'Plaza UU','Gombak',652222);
insert into Restaurant values(692553,134331,'Yumi Ila','Idaman',324229);
insert into Restaurant values(765430,114231,'Restaurant Juyy','Idaman',298554);
insert into Restaurant values(987654,104532,'Restaurant Maya','Langkawi',652422);
insert into Restaurant values(987751,136988,'Restaurant Syl','Sylhet',651232);
insert into Restaurant values(982555,122558,'Restaurant Taj','Sylhet',611422);
insert into Restaurant values(895442,124556,'HT Mango','Langkawi',987722);
insert into Restaurant values(111111,744342,'Restaurant Miz','Sylhet',265122);

insert into payment values(654423,166777,156331,3150,to_date('02-MAR-19'));
insert into payment values(654421,162324,166666,1950,to_date('21-MAR-19'));
insert into payment values(621123,167865,152341,null,to_date('25-JAN-19'));
insert into payment values(554423,162352,296651,8400,to_date('28-FEB-19'));
insert into payment values(123123,186542,656331,null,to_date('25-FEB-19'));
insert into payment values(786524,153766,199331,850,to_date('27-FEB-19'));
insert into payment values(907699,163322,185331,8500,to_date('25-APR-17'));
insert into payment values(623313,122233,136391,1300,to_date('30-JUL-18'));
insert into payment values(234567,164322,198331,null,to_date('01-APR-71'));

```
insert into Food_Menu values('I01','Pizza','western', 30);
insert into Food_Menu values('I02','Burger','western', 10);
insert into Food_Menu values('I03','Pasta','western', 15);
insert into Food_Menu values('I04','Shwarma','western', 5);
insert into Food_Menu values('I05','Nasi Ayam','malay', 6);
insert into Food_Menu values('I06','Nasi Lemak','malay', 7);
insert into Food_Menu values('I07','Nasi Goreng Pataya','malay', 6);
insert into Food_Menu values('I08','Ayam kunit','malay', 5);
insert into Food_Menu values('I09','Roti Canai','indian', 8);
insert into Food_Menu values('I10','Tanduri','indian', 6);
insert into Food_Menu values('I11','Biryani','indian', 10);
insert into Food_Menu values('I12','Kacchi','indian', 12);
```

6.0 QUERY, ADD, REMOVE, INSERT & MODIFY DATA

Value insert in customer table

```
SQL> insert into customer
values(166221,'Mojid','Orick','Ori',0164105,'oori@gmail.com','I09');

1 row created.
```

Normal Select Statement to see the details of customers

```
SQL> select * from customer;
```

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			
164322 humby	Gerave	Rai	156652	
ji232ii@gmail.com	I01			
162324 Zalali	Akash	Ahmad	276623	
kiz86i@gmail.com	I05			
162352 Jib	Nasik	Sami	176232	
nasikii@gmail.com	I09			

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			
167865 Fel	Zian	Ali	176644	
afiquezian@gmail.com	I10			
186542 Saap	Janwar	Sani	276452	
janwar@gmail.com	I12			
153766 Bolod	Tawsif	Mia	136754	
chor@gmail.com	I12			

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			
163322 Makkhi	Amir	Saki	143652	
sadia@gmail.com	I12			
122233 Gadha	Meem	Dota	123392	

maal@gmail.com I02

163221 Iot Md Sarif 164905
jiuuh@gmail.com I05

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
-----	-----	-----	-----	-----
CUSTOMER_EMAIL	ITEM_ID			
-----	-----			
166777 Mokles	Md	Sarif	176322	
jizzi@gmail.com	I02			
166221 Mojid	Orick	Ori	164105	
oori@gmail.com	I09			

11 rows selected.

Select Statement to find people with certain first name

```
SQL> select customer_id, customer_fname, customer_lname from customer where  
customer_fname='Akash';
```

CUSTOMER	CUSTOMER_FNAME	CUSTOMER_L
-----	-----	-----
162324 Akash	Ahmad	

Finding Restaurants from same location using Join

```
SQL> select restaurant_id, restaurant.manager_id, restaurant_name, location  
2 from restaurant, manager  
3 where location='Sylhet' AND restaurant.manager_id=  
manager.manager_id;
```

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION
-----	-----	-----	-----
987751	136988	Restaurant Syl	Sylhet
982555	122558	Restaurant Taj	Sylhet
111111	744342	Restaurant Miz	Sylhet

Finding the bill of a certain person using Join (Condition: Name= Md Sarif)

```
SQL> select customer.customer_id,customer_fname,customer_lname,booking_id,
2 booking_date,checkout_date,customer.Item_id,Item_name,Item_type,
3 price from customer, booking,food_menu
4 Where customer.customer_id= booking.customer_id AND
5 customer.Item_ID=food_menu.item_Id AND customer_fname='Md' AND
6 customer_lname='Sarif';
```

CUSTOMER	CUSTOMER_FNAME	CUSTOMER_L	BOOKING_ID	BOOKING_D	CHECKOUT_	ITEM_ID
163221	Md	Sarif	156331	25-DEC-19	28-DEC-19	I05
Nasi Ayam		malay	6			
166777	Md	Sarif	189766	27-FEB-18	01-MAR-18	I02
Burger		western	10			
166777	Md	Sarif	876623	28-AUG-18	02-SEP-18	I02
Burger		western	10			

Finding the Restaurant name and location of a manager's Restaurant using subquery

```
SQL> select restaurant_name, location,restaurant_phone as contact from
restaurant
2 where manager_id=(select manager_id from manager
3 where manager_fname='Mujib' AND manager_lname='Ali');
```

RESTAURANT_NAME	LOCATION	CONTACT
Restaurant Miz	Sylhet	265122

Update a table to change a value

```
SQL> select * from customer;
```

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
164322	humby	Gerave	Rai	156652
ji232ii@gmail.com		I01		

162324	Zalali	Akash	Ahmad	276623
kiz86i@gmail.com	I05			

162352	Jib	Nasik	Sami	176232
nasikii@gmail.com	I09			

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
-----	-----	-----	-----	-----

CUSTOMER_EMAIL	ITEM_ID			
-----	-----			

167865	Fel	Zian	Ali	176644
afiquezian@gmail.com	I10			

186542	Saap	Janwar	Sani	276452
janwar@gmail.com	I12			

153766	Bolod	Tawsif	Mia	136754
chor@gmail.com	I12			

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
-----	-----	-----	-----	-----

CUSTOMER_EMAIL	ITEM_ID			
-----	-----			

163322	Makkhi	Amir	Saki	143652
sadia@gmail.com	I12			

122233	Gadha	Meem	Dota	123392
maal@gmail.com	I02			

163221	Iot	Md	Sarif	164905
jiuuh@gmail.com	I05			

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
-----	-----	-----	-----	-----

CUSTOMER_EMAIL	ITEM_ID			
-----	-----			

166777	Mokles	Md	Sarif	176322
jizzi@gmail.com	I02			

166221	Mojid	Orick	Ori	164105
oori@gmail.com	I09			

11 rows selected.

```
SQL> update customer
  2  set customer_username='Borodhon'
  3  where customer_fname='Zian'AND customer_lname='Ali';
```

1 row updated.

```
SQL> select * from customer;
```

CUSTOMER	CUSTOMER_USERNAME	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			

164322	humby	Gerave	Rai	156652
ji232ii@gmail.com	I01			

162324	Zalali	Akash	Ahmad	276623
kiz86i@gmail.com	I05			

162352	Jib	Nasik	Sami	176232
nasikii@gmail.com	I09			

CUSTOMER	CUSTOMER_USERNAME	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			

167865	Borodhon	Zian	Ali	176644
afiquezian@gmail.com	I10			

186542	Saap	Janwar	Sani	276452
janwar@gmail.com	I12			

153766	Bolod	Tawsif	Mia	136754
chor@gmail.com	I12			

CUSTOMER	CUSTOMER_USERNAME	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			

163322	Makkhi	Amir	Saki	143652
sadia@gmail.com	I12			

122233	Gadha	Meem	Dota	123392
maal@gmail.com	I02			

163221	Iot	Md	Sarif	164905
jiuuh@gmail.com	I05			

CUSTOMER	CUSTOMER_USERNAME	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE
CUSTOMER_EMAIL	ITEM_ID			

166777	Mokles	Md	Sarif	176322
jizzi@gmail.com		I02		
166221	Mojid	Orick	Ori	164105
oori@gmail.com		I09		

11 rows selected.

Delete values from a table

SQL> select * from payment;

PAYMENT_ID	CUSTOMER	BOOKING_ID	PAYMENT_AMOUNT	PAYMENT_D
654423	166777	156331	3150	02-MAR-19
654421	162324	166666	1950	21-MAR-19
621123	167865	152341		25-JAN-19
554423	162352	296651	8400	28-FEB-19
123123	186542	656331		25-FEB-19
786524	153766	199331	850	27-FEB-19
907699	163322	185331	8500	25-APR-17
623313	122233	136391	1300	30-JUL-18
234567	164322	198331		01-APR-71

9 rows selected.

SQL> delete from payment
2 where payment_amount >= 8000;

2 rows deleted.

SQL> select * from payment;

PAYMENT_ID	CUSTOMER_ID	BOOKING_ID	PAYMENT_AMOUNT	PAYMENT_D
621123	157865	152341		25-JAN-19
123123	176542	656331		25-FEB-19
786524	143766	199331	850	27-FEB-19
234567	154322	198331		01-APR-71
654423	156777	156331	3150	02-MAR-19
654421	152324	166666	1950	21-MAR-19
623313	112233	136391	1300	30-JUL-18

7 rows selected.

Creating a view

```
SQL> create view manager_count
  2  as select restaurant_id, manager_id, restaurant_name
  3  from restaurant
  4  order by manager_id DESC;
```

View created.

Join a table with a view to find people with First Name Rojaro

```
SQL> select * from (select
m.manager_id,manager_fname,manager_lname,Restaurant_name
  2      from manager_count l,manager m
  3      where m.manager_id=l.manager_id AND manager_fname='Rojaro');
```

MANAGER_ID	MANAGER_FN	MANAGER_LN	RESTAURANT_NAME
136625	Rojaro	Alim	Restaurant Mihh
126777	Rojaro	shen	Restaurant Taj

PROCEDURES

1.Sometimes customers change the checkout date of the Restaurant. Using this procedure the user can update the checkout date of a customer.

```
create or replace procedure call(ID booking,booking_ID%TYPE,c_customer_id
booking.customer_id%TYPE,newdate date) is
BEGIN
```

```
    update booking
    set booking_date=newdate
    where booking_id=ID and customer_id=c_customer_id;
```

```
    DBMS_OUTPUT.PUT_LINE('The data has updated');
END;
/
```

--ANONYMOUS BLOCK

--=====

ACCEPT ID PROMPT 'Enter the booking ID: '

ACCEPT c_customer_id PROMPT 'Enter the Customer ID: '

ACCEPT newdate PROMPT 'Enter the new booking date: '

DECLARE

v_origdate booking.booking_date%type;

BEGIN

```
select booking_date
into v_origdate
from booking
where booking_id = '&ID'
and customer_id = '&c_customer_id';
```

```
call ('&ID', '&c_customer_id', '&newdate');
DBMS_OUTPUT.PUT_LINE('Data updated ');
END;
/
```

Booking table before the updated checkout date

BOOKING_ID	MANAGER_ID	CUSTOMER	BOOKING_TITLE	BOOKING_TYPE	BOOKING_D
CHECKOUT_					
206651	164332	163322	Western	Take Away	02-DEC-19
656221	122558	153766	Indian	Take Away	24-DEC-19
199331	128762	186542	Malay	Eatin	24-DEC-19
BOOKING_ID MANAGER_ID CUSTOMER BOOKING_TITLE BOOKING_TYPE BOOKING_D					
CHECKOUT_					
185331	744342	167865	Western	Eatin	02-DEC-17
136391	126777	162352	Indian	Take Away	29-NOV-18
198331	122558	162324	Malay	Eatin	26-NOV-17
BOOKING_ID MANAGER_ID CUSTOMER BOOKING_TITLE BOOKING_TYPE BOOKING_D					
CHECKOUT_					
189766	124556	166777	Western	Take Away	27-FEB-18
876623	136625	166777	Indian	Eatin	28-AUG-18

11 rows selected.

7.0 Procedure Execution

```
SQL>
SQL> @"C:\Users\Md. Afique Amin Zian\Desktop\procedure1.txt"
SP2-0734: unknown command beginning "1.Sometime..." - rest of line ignored.

Procedure created.

Enter the booking ID: 189766
Enter the Customer ID: 166777
Enter the new booking date: 28-FEB-18
old 9: where booking_id = '&ID'
new 9: where booking_id = '189766'
old 10: and customer_id = '&c_customer_id';
new 10: and customer_id = '166777';
old 12: call ('&ID', '&c_customer_id', '&newdate');
new 12: call ('189766', '166777', '28-FEB-18');

PL/SQL procedure successfully completed.
```

Booking table after the Procedure

BOOKING_ID	MANAGER_ID	CUSTOMER	BOOKING_TITLE	BOOKING_TYPE	BOOKING_D
185331	744342	167865	Western	Eatin	02-DEC-17
136391	126777	162352	Indian	Take Away	29-NOV-18
198331	122558	162324	Malay	Eatin	26-NOV-17
189766	124556	166777	Western	Take Away	28-FEB-18
876623	136625	166777	Indian	Eatin	28-AUG-18

11 rows selected.

2.This Procedure is used to see the details of a customer

```
create or replace procedure call2(c_customer_ID
customer.customer_id%TYPE,c_customer_username
customer.customer_username%TYPE) IS
fname varchar2(20);
lname varchar2(20);
phone number;
email varchar2(20);

    BEGIN
        select customer_fname,customer_lname,customer_phone,customer_email
into fname,lname,phone,email
        from customer
        where c_customer_id=customer_id
        AND c_customer_username=customer_username;

        dbms_output.put_line('First Name: '||fname);
        dbms_output.put_line('Last Name: '||lname);
        dbms_output.put_line('Phone Number: '||phone);
        dbms_output.put_line('Email: '||email);
    end;
/

--ANONYMOUS BLOCK
=====

ACCEPT custID PROMPT 'Enter the customer ID: '

ACCEPT cust_username PROMPT 'Enter the Customer username: '

begin
call2('&custID','&cust_username');

dbms_output.put_line('The data is successfully entered. ');
end;
/
```

Procedure Execution

```
SQL> set serveroutput on
SQL> @"D:\Semester 4\Database programming\Project\procedure\procedure2.txt"

Procedure created.

Enter the customer ID: 112233
Enter the Customer username: Izumi
old 2: call2('&custID','&cust_username');
new 2: call2('112233','Izumi');
First Name: Meepo
Last Name: Dota
Phone Number: 122392
Email: meepo@gmail.com
The data is successfully entered.

PL/SQL procedure successfully completed.
```

3.Using this procedure the user can insert new values in the Restaurant table

```
-----  
create or replace procedure newval  
(h_Restaurant_id Restaurant.Restaurant_id%TYPE,  
 h_manager_id Restaurant.manager_id%TYPE,  
 h_Restaurant_name Restaurant.Restaurant_name%TYPE,  
 h_location Restaurant.location%TYPE,  
 h_Restaurant_phone Restaurant.Restaurant_phone%TYPE  
) IS  
BEGIN  
INSERT INTO Restaurant  
(Restaurant.Restaurant_id,Restaurant.manager_id,Restaurant.Restaurant_name  
,Restaurant.location,Restaurant.Restaurant_phone)  
VALUES  
(h_Restaurant_id,h_manager_id,h_Restaurant_name,h_location,h_Restaurant_ph  
one);  
DBMS_OUTPUT.PUT_LINE('Data successfully added');  
END;  
/  
--ANONYMOUS BLOCK  
-----  
ACCEPT h_Restaurant_id prompt 'ENTER RESTAURANT ID: '  
ACCEPT h_manager_id prompt 'ENTER MANAGER ID: '  
ACCEPT h_Restaurant_name prompt 'ENTER RESTAURANT NAME: '  
ACCEPT h_location prompt 'ENTER LOCATION: '  
ACCEPT h_Restaurant_phone prompt 'ENTER RESTAURANT PHONE NUMBER: '  
  
BEGIN  
newval('&h_Restaurant_id','&h_manager_id','&h_Restaurant_name','&h_locatio  
n','&h_Restaurant_phone');  
END;  
/  
-----
```

Restaurant table before Entering a new values


```
SQL> select * from restaurant;
```

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION	RESTAURANT_PHONE
987755	126777	Restaurant Taj	Gombak	655422
988772	136625	Restaurant Mihh	Idaman	655222
163265	146729	Marks	Selangor	152443
875542	128762	Plaza UU	Gombak	652222
692553	134331	Yumi Ila	Idaman	324229
765430	114231	Restaurant Juyy	Idaman	298554
987654	104532	Restaurant Maya	Langkawi	652422
987751	136988	Restaurant Syl	Sylhet	651232
982555	122558	Restaurant Taj	Sylhet	611422
895442	124556	HT Mango	Langkawi	987722
111111	744342	Restaurant Miz	Sylhet	265122

```
11 rows selected.
```

Procedure Execution

```
SQL> set serveroutput on
SQL> @"C:\Users\Md. Afique Amin Zian\Desktop\procedure3.txt"
```

```
Procedure created.
```

```
ENTER RESTAURANT ID: 111222
ENTER MANAGER ID: 126777
ENTER RESTAURANT NAME: Restarant Kubur
ENTER LOCATION: Kualalumpur
ENTER RESTAURANT PHONE NUMBER: 123456
old 2: newval('&h_Restaurant_id','&h_manager_id','&h_Restaurant_name','&h_location','&h_Restaurant_phone');
new 2: newval('111222','126777','Restarant Kubur','Kualalumpur','123456');
Data successfully added
```

```
PL/SQL procedure successfully completed.
```

Restaurant Table after the execution of the procedure

```
SQL> select * from restaurant;
```

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION	RESTAURANT_PHONE
987755	126777	Restaurant Taj	Gombak	655422
988772	136625	Restaurant Mihh	Idaman	655222
163265	146729	Marks	Selangor	152443
875542	128762	Plaza UU	Gombak	652222
692553	134331	Yumi Ila	Idaman	324229
765430	114231	Restaurant Juyy	Idaman	298554
987654	104532	Restaurant Maya	Langkawi	652422
987751	136988	Restaurant Syl	Sylhet	651232
982555	122558	Restaurant Taj	Sylhet	611422
895442	124556	HT Mango	Langkawi	987722
111111	744342	Restaurant Miz	Sylhet	265122

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION	RESTAURANT_PHONE
111222	126777	Restarant Kubur	Kualalumpur	123456

```
12 rows selected.
```

```
SQL>
```

8.0 FUNCTIONS

1. Using this Function the user can see the bill of a certain customer. For the bill if the Restaurant is situated in Gombak, the bill will be multiplied by 1.4. If the Restaurant is situated in Idaman, the bill will be multiplied by 1.3. if the Restaurant is situated in Selangor, the bill will be multiplied by 1.6. if the Restaurant is situated in Sylhet, the bill

will be multiplied by 1.8. if the Restaurant is situated in Langkawi, the bill will be multiplied by 2.

For the types of the food, the rent of the take away western food are 500 rm and the Indian and malay food 250 . If the food is eaten the price will be added by 10 and for the take away the price will be added by 15. This function will check the booking date and the checkout date of the customer and calculate the bill.

```
-----  
CREATE OR REPLACE FUNCTION citydisc  
(p_id in number)  
return number  
is  
price number;  
p_type varchar2(12);  
p_title varchar2(12);  
p_bookingdate date;  
p_checkoutdate date;  
p_location varchar(12);  
c_number number;  
  
BEGIN  
select booking_type,booking_title,checkout_date,booking_date,location into  
p_type,p_title,p_checkoutdate,p_bookingdate,p_location from booking join Restaurant on  
booking.manager_id=Restaurant.manager_id  
AND (p_id=booking_id);  
  
if p_location='Gombak' then c_number:=1.4;  
elsif p_location='Idaman' then c_number:=1.3;  
elsif p_location='Selangor' then c_number:=1.6;  
elsif p_location='Sylhet' then c_number:=1.8;  
elsif p_location='Langkawi' then c_number:=2;  
end if;  
  
if p_type='Eatin' AND p_title='Western' then price :=(10+500)*(p_checkoutdate-  
p_bookingdate)*c_number;  
elsif p_type='Take Away' AND p_title='Western' then price:=(15+500)*(p_checkoutdate-  
p_bookingdate)*c_number;  
elsif p_type='Eatin' AND p_title='Indian ' then price:=(10+250)*(p_checkoutdate-  
p_bookingdate)*c_number;  
elsif p_type='Take Away' AND p_title='indian' then price:=(15+250)*(p_checkoutdate-  
p_bookingdate)*c_number;  
elsif p_type='Take Away' AND p_title='Malay' then price:=(15+250)*(p_checkoutdate-  
p_bookingdate)*c_number;  
elsif p_type='Eatin' AND p_title='Malay' then price:=(10+250)*(p_checkoutdate-  
p_bookingdate)*c_number;
```

```

end if;

return(price);
end;
/

--ANONYMOUS BLOCK
--=====

ACCEPT p_id prompt 'Enter Booking ID:'
declare
p_id number;

BEGIN
p_id:=&p_id;

DBMS_OUTPUT.put_line
('Total Bill is : RM'||citydisc(p_id));
END;
/

```

Function Execution

```

SQL> set serveroutput on
SQL> @"C:\Users\Md. Afique Amin Zian\Desktop\baaalssaaal.txt"

Function created.

Enter Booking ID:166666
old 5: p_id:=&p_id;
new 5: p_id:=166666;
Total Bill is : RM1339

PL/SQL procedure successfully completed.

```

2. Using this function the user will provide discount. If the User pays the before the checkout date he/she will get 20% discount. If the user pays the bill before 5 days after

checkout, he/she will get 10% discount. If the user pays the bill after 5 days, he/she will get 5% discount

```
-----
CREATE OR REPLACE FUNCTION discount1
(c_id in number)
return number
is
discount number;
p_checkoutdate date;
p_paymentdate date;

BEGIN
select checkout_date,payment_date into p_checkoutdate,p_paymentdate from booking join
payment on payment.booking_id=booking.booking_id
AND (c_id=payment.booking_id);

if (p_paymentdate-p_checkoutdate)<0 then discount:=.2;
elsif (p_paymentdate-p_checkoutdate)>=0 AND (p_paymentdate-p_checkoutdate)<5 then
discount:=.1;
elsif (p_paymentdate-p_checkoutdate)>5 then discount:=.05;
end if;

return(discount);
end;
/

--ANONYMOUS BLOCK
--=====

ACCEPT c_id prompt 'Enter Booking ID:'
declare
c_id number;

BEGIN
c_id:=&c_id;

DBMS_OUTPUT.put_line
('Total Discount is '||discount1(c_id));
END;
/
```

Function Execution

```
SQL>
SQL> @"C:\Users\Md. Afique Amin Zian\Desktop\baaalssaaal.txt"

Function created.

Enter Booking ID:199331
old   5: c_id=&c_id;
new   5: c_id=199331;
Total Discount is .2

PL/SQL procedure successfully completed.
```

9.0 CURSOR

Implicit Cursor

Using Cursor to delete Restaurants of certain location from Restaurant table

```
Set serveroutput on
BEGIN
DELETE FROM Restaurant
where location='Sylhet';

IF SQL%NOTFOUND THEN
    DBMS_OUTPUT.PUT_LINE('No maching record');
ELSIF SQL%FOUND then DBMS_OUTPUT.PUT_LINE('DATA Successfully deleted');
END IF;
```

```
END;  
/
```

Restaurant table before cursor execution

```
SQL> select * from restaurant ;
```

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION	RESTAURANT_PHONE
163265	156729	Marks	Selangor	152443
987654	154532	Restaurant Maya	Langkawi	652422
987751	156988	Restaurant Syl	Sylhet	651232
982555	192558	Restaurant Taj	Sylhet	611422
895442	169577	HT Mango	Langkawi	987722
111111	164332	Restaurant Miz	Sylhet	265122

Cursor Execution

```
SQL> Set serveroutput on  
SQL> BEGIN  
2  DELETE FROM Restaurant  
3  where location='Sylhet';  
4  
5  IF SQL%NOTFOUND THEN  
6  DBMS_OUTPUT.PUT_LINE('No maching record');  
7  ELSIF SQL%FOUND then DBMS_OUTPUT.PUT_LINE('DATA Successfully deleted');  
8  END IF;  
9  END;  
10 /  
DATA Successfully deleted  
  
PL/SQL procedure successfully completed.  
  
SQL> select * from restaurant;
```

RESTAURANT_ID	MANAGER_ID	RESTAURANT_NAME	LOCATION	RESTAURANT_PHONE
163265	156729	Marks	Selangor	152443
987654	154532	Restaurant Maya	Langkawi	652422
895442	169577	HT Mango	Langkawi	987722

Explicit Cursor

Using cursor to see the customer id, first name and last name of the the first five customers from customer table

Declare

```
c_customer_id customer.customer_id%TYPE;  
c_customer_FNAME customer.customer_fname%TYPE;  
c_customer_LNAME customer.customer_lname%TYPE;
```

```
        CURSOR customer_cursor IS  
        select customer_id,customer_fname,customer_lname  
        from customer;  
BEGIN
```

```

        OPEN customer_cursor;
    LOOP
        FETCH customer_cursor into
c_customer_id,c_customer_FNAME,c_customer_LNAME;
        EXIT WHEN customer_cursor%ROWCOUNT >5 OR
                customer_cursor%NOTFOUND;

        DBMS_OUTPUT.PUT_LINE
        (to_char(c_customer_id)||      '||c_customer_FNAME||'      '||c_customer_LNAME);
    END LOOP;
CLOSE customer_cursor;
END;
/

```

Cursor Execution

```

SQL> @"D:\Semester 4\Database programming\Project\Cursors\Explicit Cursor.txt"
156777      Mehran      Mujib
152324      Akkas      Ali
152352      Nasik      Sami
157865      Zian      Ali
176542      Sanwar      Sani

```

Explicit cursor application

Using cursor to see the customer id, customer username of the customers from customer table using %notfound and application of loop

```

set serveroutput on
set verify off
Declare
cursor C1 IS select Customer_Id, Customer_username from customer;
vc_Id customer.Customer_Id%type;
vc_Uname customer.Customer_Username%type;
begin
open C1;
loop
fetch C1 into vc_Id, vc_Uname;

```



```

exit when C1% notfound;
Dbms_output.put_line (vc_Id ||'-----'|| vc_Uname);
end loop;
close C1;
end;
/

```

```

SQL> select * from customer
2 ;

```

CUSTOMER	CUSTOMER_USERNA	CUSTOMER_FNAME	CUSTOMER_L	CUSTOMER_PHONE	CUSTOMER_EMAIL
156777	Mke	Mehran	Mujib	176652	jizzii@gmail.com
152324	Zkammi	Akkas	Ali	176623	jiz66i@gmail.com
152352	Jibbas	Nasik	Sami	176232	nasikii@gmail.com
157865	Mzb	Zian	Ali	176654	zian@gmail.com
176542	Saah	Sanwar	Sani	176452	sanwar@gmail.com
143766	Iuuy	Tawsif	Mia	176754	choy@gmail.com
153322	MKiaa	Amith	Sadi	123652	sadi@gmail.com
112233	Izumi	Meepo	Dota	122392	meepo@gmail.com
154322	huuye	Gerault	Ravia	176652	ji232ii@gmail.com
153221	Ioe	Mehran	Mujib	164900	jiuuh@gmail.com
132156	Ilam	Gerault	Mujib	1122329	6iwzii@gmail.com

```

11 rows selected.

```

Result after the execution

```

SQL> @"D:\Semester 4\Database programming\Project\Cursors\Explicit Cursor.txt"
156777      Mehran      Mujib
152324      Akkas       Ali
152352      Nasik       Sami
157865      Zian        Ali
176542      Sanwar      Sani

```

10.0 Web Application:

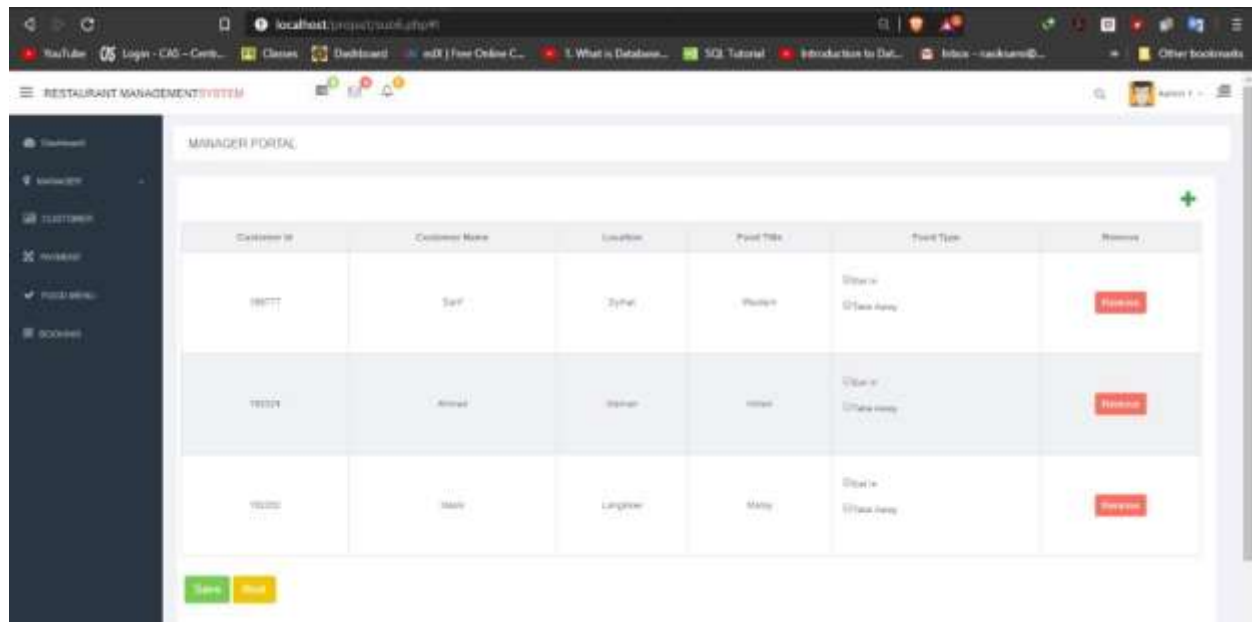
We try to do it in three phases with three tools.

We used HTML,CS, JAVASCRIPT, BOOTSTRAP template,PHP and XAMPP for this prototype.

It can insert the data, Update the data and can delete from the Database.

Here we have showed the functionality of the manager portal where the manager can add the details of a customer and his orders and he can also remove certain customer from this list . On the left side panel we have given the list of the six tables which is included in our database.

1. Using Boostrap and Php Tools:



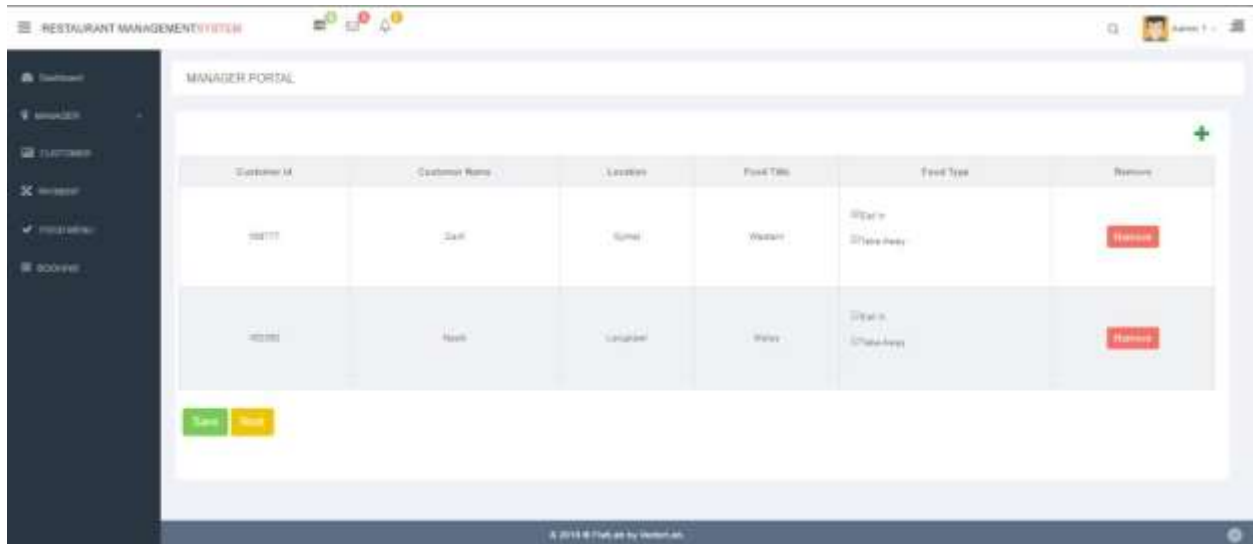
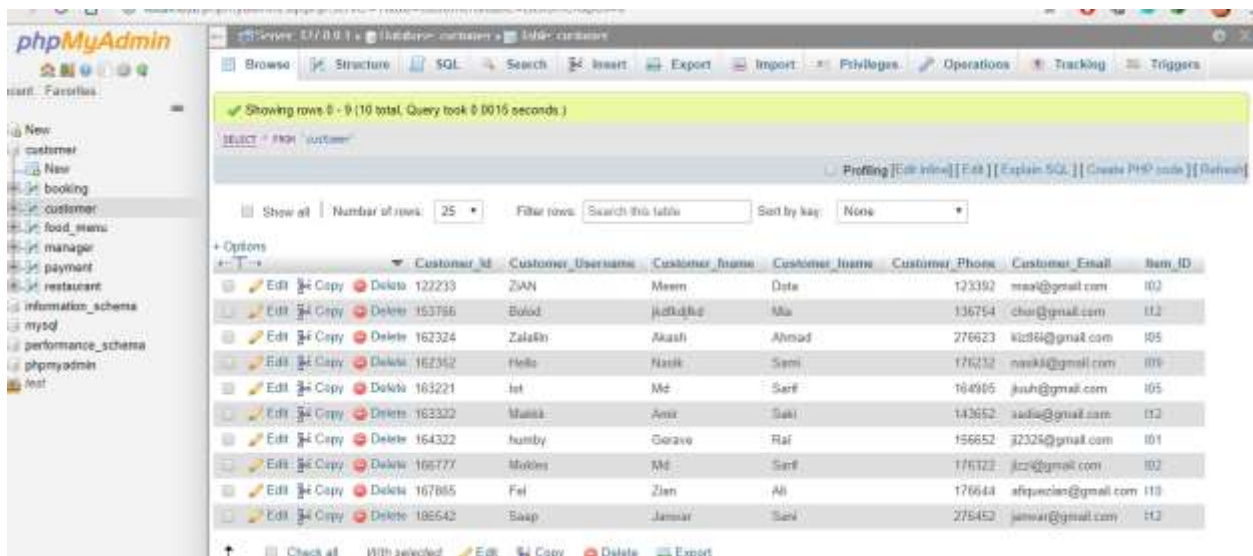


Figure: Using Bootstrap and website

2. Using Xampp And PhpMyAdmin(With Java Script css and PHP codes):



Welcome to Customer Information

Record

CUSTOMER_ID ▾ Type to Search

Search

CUSTOMER_ID	UserName	Fname	Lname	Phone	Email	Item_ID	Option
122233	ZiAN	Meem	Dota	123392	maal@gmail.com	I02	Delete / Edit
153766	Bolod	jkdfkdjfkf	Mia	136754	choe@gmail.com	I12	Delete / Edit
162324	Zalalin	Akash	Ahmad	276623	kiz86i@gmail.com	I05	Delete / Edit
162352	Hello	Nasik	Sami	176232	nasaki@gmail.com	I09	Delete / Edit
163221	Iot	Md	Sarif	164905	jjuuh@gmail.com	I05	Delete / Edit
163322	Makkk	Amir	Saki	143652	sadia@gmail.com	I12	Delete / Edit
164322	humby	Gerave	Rai	156652	ji232u@gmail.com	I01	Delete / Edit
166777	Mokies	Md	Sarif	176322	jizzi@gmail.com	I02	Delete / Edit
167865	Fel	Zian	Ali	176644	afiquezian@gmail.com	I10	Delete / Edit
186542	Snap	Janwar	Sani	276452	janwar@gmail.com	I12	Delete / Edit

Welcome to Add Form



CUSTOMER_ID:

UserName:

Fname:

Lname:

Phone:

Email:

ITEM_ID:

Add

Clear

```

connect.php      index.php      searchandhome.php      Edit_Form.php
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Edit Form</title>

<link rel="stylesheet" type="text/css" href="Style/Edit_Form.css" />

</head>

<body background="Images/MyBackground.png" bgcolor="#FFCC99">
<div class="topbar"><h1 style="color:#FFF"><center>Edit Form</center></h1></div>
<center>
<div class="box" ">
<?php
$Id = '';
if( isset( $_GET['txtid'])) {
$Id = $_GET['txtid'];
}
include ("connect.php");
$sql = "select * from Customer where Customer_Id=".$Id;
$h= mysql_query($conn,$l);
if($h=mysql_fetch_array($h))
{
<table><form method="post" action="">
<tr>
<th>CUSTOMER ID</th>

```

Figure: Using Xampp and Phpmyadmin

3. We try make the web app using caspio also.

All Data pages store

Name	Data Source	Authentication	Deployed	Last Modified
add new customer	Customer		Enabled	
Booking Web Form	Booking		Enabled	29 Dec 2019 05:37 AM
Hotel Web Form	Hotel		Enabled	29 Dec 2019 05:39 AM
Manager Web Form	Manager	Demo-Admin Login	Enabled	29 Dec 2019 05:35 AM
Payment Web Form	Payment		Enabled	29 Dec 2019 05:41 AM

Website for our restaurant



All submission will be stored in our database tables

For an example we are adding a booking information from customers

Restaurent

HOME BOOKING CUSTOMERS HOTELS PAYMENTS MANAGERS

booking id *	<input type="text" value="1626667"/>
manager_id *	<input type="text" value="126373"/>
customer id *	<input type="text" value="123456"/>
booking tittle	<input type="text" value="take away"/>
booking type	<input type="text" value="indian"/>
booking date	<input type="text" value="30/12/2019"/>
checkout date	<input type="text" value="31/12/2019"/>

POWERED BY weebly

And now it's already added into booking table

caspio BRIDGE

Home Messages Account Tools Help Search carful Upgrade Account

hotel management

Overview

DataPages

Tables

Views

Authentications

Connections

Styles

Localizations

Tasks

Tables > Booking

Datasheet Table Design Triggered Actions

New Refresh Download Find Replace Filter Visual Automation Delete Delete All

14 Records 5 of 5

	booking_id	Manager_id	customer_id	booking_title	booking_type	booking_date	checkout_date
	156331	156777	152324	Single	Premium	12/25/2019	12/28/2019
	166666	156625	156777	Double	Normal	12/25/2019	12/27/2019
	152341	156729	152352	Single	Premium	12/22/2019	12/24/2019
	15262	30474	4647587	eating	western	12/30/2019	12/31/2019
	153668	153372	123456	take away	indian	12/30/2019	12/31/2019

Live Help

This page authorized by managers and only authorized managers can inter in this pag

HOME BOOKING CUSTOMERS HOTELS PAYMENTS MANAGERS

Managers

Using the credentials below, sign in as an administrator to manage the directory.

Manage Physicians

Email

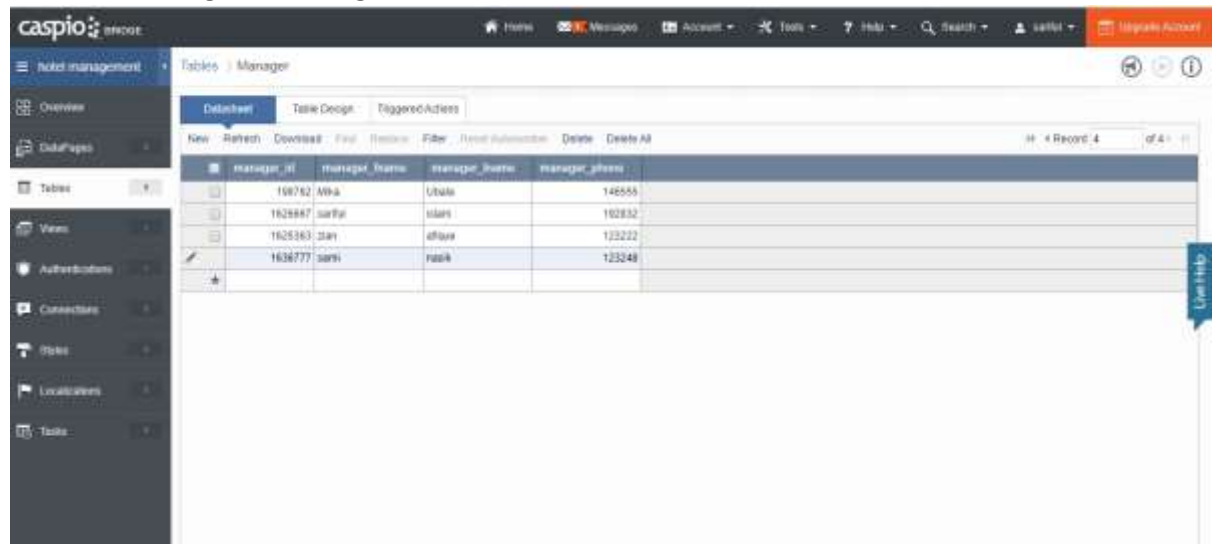
Password

LOGIN

POWERED BY weebly

This is the table for managers who can inter manager webpages

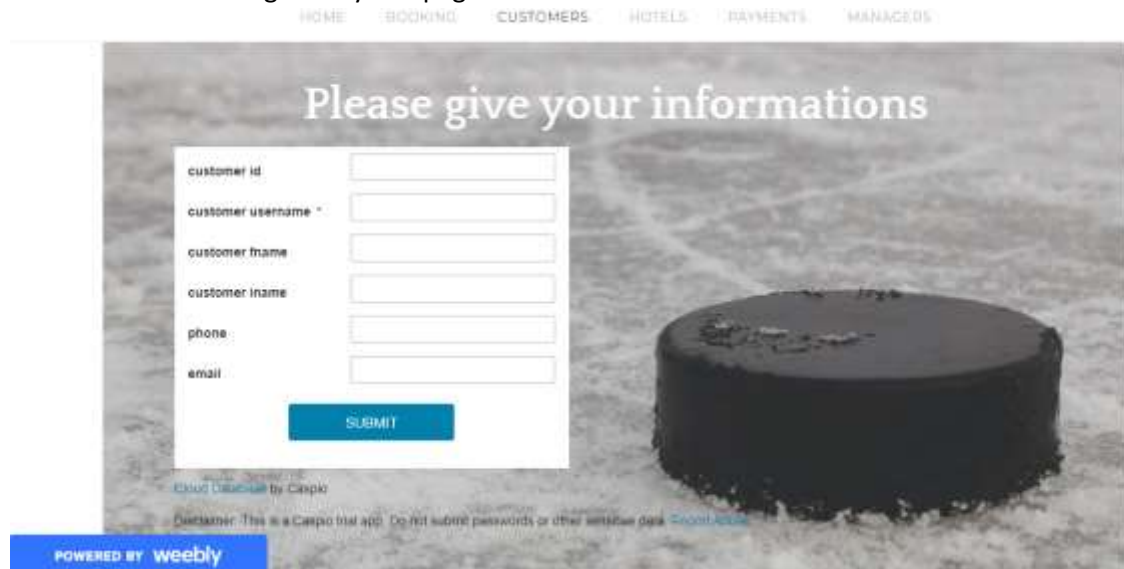
This table manages all manager's informations



The screenshot shows the Caspio interface for a table named 'Manager'. The table has four columns: 'manager_id', 'manager_firstname', 'manager_lastname', and 'manager_phone'. There are four records in the table.

manager_id	manager_firstname	manager_lastname	manager_phone
160762	Anna	Utah	146555
162887	Suraj	slas	162832
162583	Alan	offore	123222
163777	Sarah	nasik	123248

New customer can register by this page



The screenshot shows a registration form with the following fields: 'customer id', 'customer username *', 'customer fname', 'customer lname', 'phone', and 'email'. A 'SUBMIT' button is at the bottom. The form is set against a background image of a chocolate cake. At the bottom, there is a 'POWERED BY weebly' logo and a disclaimer: 'Disclaimer: This is a Caspio trial app. Do not submit passwords or other sensitive data. Report Abuse'.

Customer can choose their hotel by this page and all information will be stored in database

HOME BOOKING CUSTOMERS HOTELS PAYMENTS MANAGERS

Choose your favorite Hotels

hotel id *

manager id

hotel name

location

hotel phone

SUBMIT

Figure: Using Caspio

References:

1. https://learn.org/articles/What_Does_a_Database_Programming_Professional_Do.html
2. https://www.google.com/search?q=database+programming+with+sql+midterm+exam&rlz=1C1CHZL_enMY753MY753&oq=database+programming+&aqs=chrome.2.69i57j0l4j69i61j69i60j69i61.8684j0j7&sourceid=chrome&ie=UTF-8
3. https://www.tutorialspoint.com/oracle_terminal_online.php
4. <https://www.capterra.com/restaurant-management-software/>

.....END.....