**BDM PROJECT**

**SUBMITTED BY –**

**SAGAR VASAIKAR (188009852)**

**TARANG HARIA (190004326)**

**SAGAR GUL ADVANI (188009815)**

**PART 1**

**Tables –**

Customer(Cust\_id,Cust\_name,Cust\_age,Cust\_ph,Cust\_type,Cust\_pref)

Guest(Cust\_id,Guest\_name)

Rooms(Room\_id,Room\_type,Floor,Amenities,Room\_price)

Events(Events\_id,Cust\_id,Room\_id,Event\_name,Events\_type,No\_of\_part)

Booking(Book\_id,Cust\_id,Room\_id,Book\_type,Book\_desc,Date)

Payments(Pay\_id,Cust\_id,Book\_id,Pay\_amount,Pay\_date,Pay\_type)

**Assumptions and Constraints** **–**

Cust\_id is UNIQUE,NOT NULL and PRIMARY KEY in the Customer table.

Cust\_type can be primary or non-primary where the primary customers are responsible for the bill and non-primary customers are the dependents.

Cust\_pref is the preference of the room type that the customer wants.

Cust\_id in the guest table is the foreign key of Customer table which shows us who all dependents came along with the primary customer.

Room\_id is the Room number which will be UNIQUE,NOT NULL and it is the PRIMARY KEY of the Rooms table.

Room\_type can be single,double,king,queen,banquet or lawn.

Amenities can be spa,gym,parking,restaurant,room service,games lounge.

Room\_price depends on the seasons and the discounts if there are any.

Events\_id is UNIQUE,NOT NULL and it is the PRIMARY KEY of the Events table.

Events\_type can be corporate,birthday,wedding,babyshower.

No\_of\_part depends on the event and how many people are attending.

Cust\_id and Room\_id are FOREIGN KEYS in the Events table.

Book\_id is UNIQUE,NOT NULL and PRIMARY KEY of the Booking table.

Book\_type is the mode of booking which can be on spot or online.

Book\_desc can be additional notes mentioned by the Customer.

Cust\_id and Room\_id are the FOREIGN KEYS in the Booking table.

Pay\_id is UNIQUE,NOT NULL and PRIMARY KEY of the Payments table.

Pay\_type can be Debit,Credit or Online.

Cust\_id and Book\_id are the FOREIGN KEYS in the Payments table.

***ER DIAGRAM IS ATTACHED IN THE ZIP FILE.***

**PART 2 AND 3:**

**QUERIES FOR CREATING THE DATABASE AND SCREENSHOTS OF THE CREATED DATABASES:**

CREATE TABLE `sts`.`Customer`(

`Cust\_id` int PRIMARY KEY NOT NULL,

`Cust\_name` varchar(45),

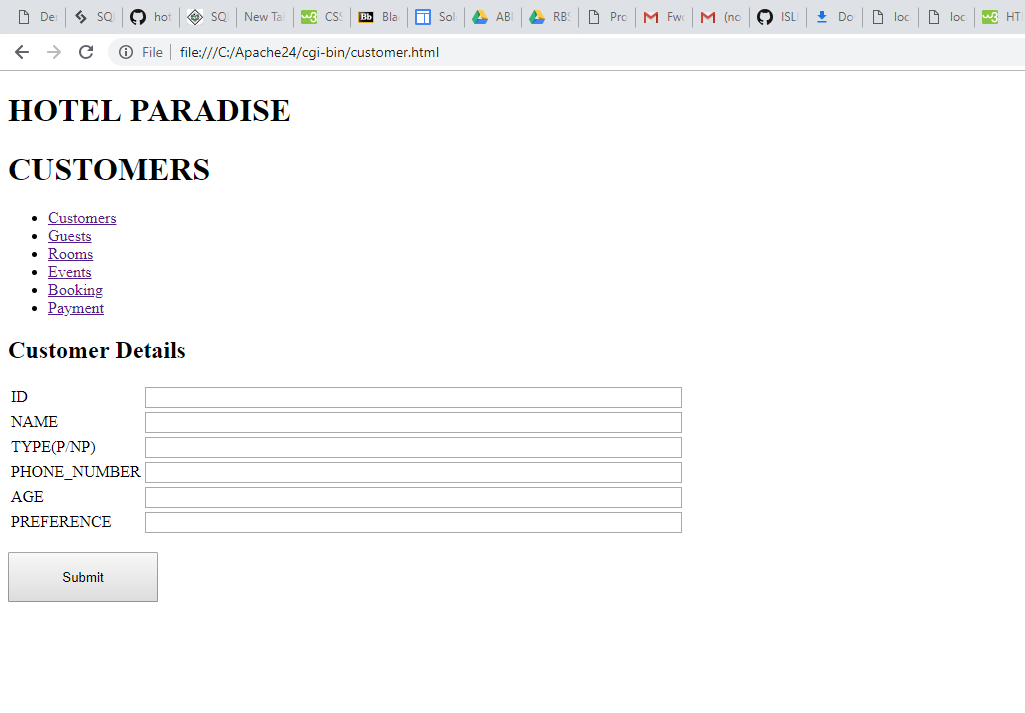
`Cust\_type` varchar(45),

`Cust\_ph`varchar(10) CHECK (length(Cust\_ph=10)),

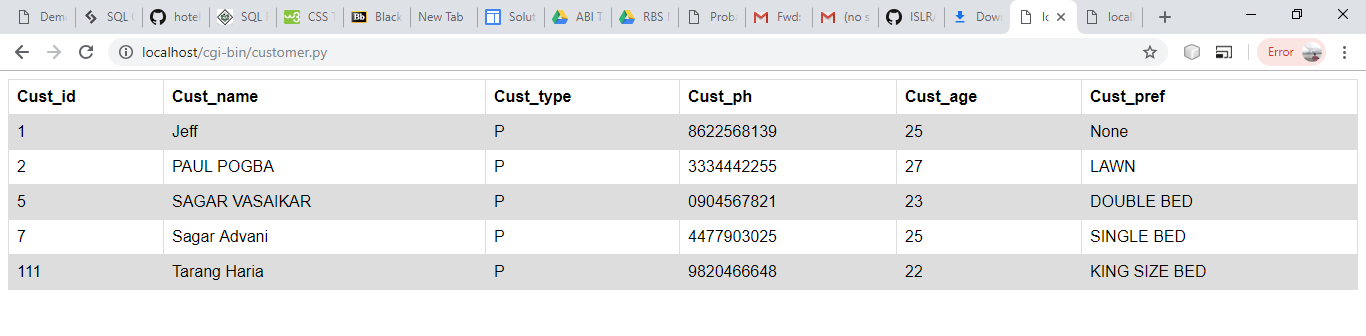
`Cust\_age` int,

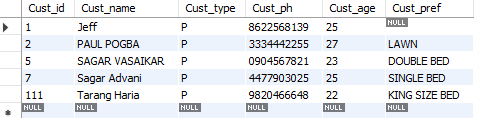
`Cust\_pref`varchar(45));

This is how our Customer page looks :



Below is the screenshot of our customer database created on browser:



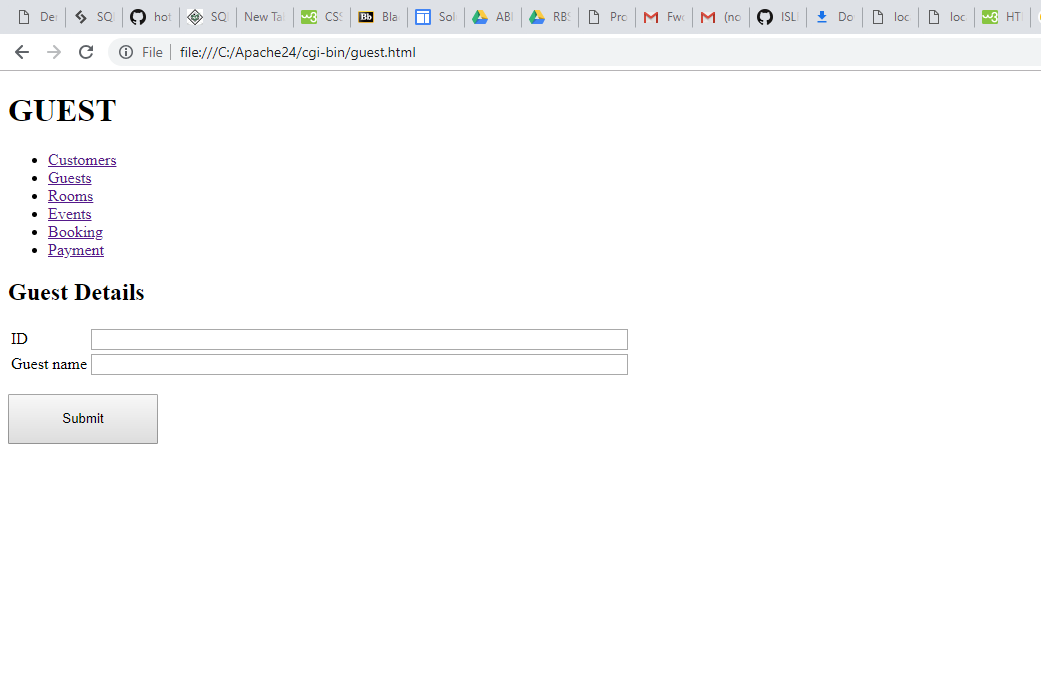


CREATE TABLE `sts`.`Guest`(

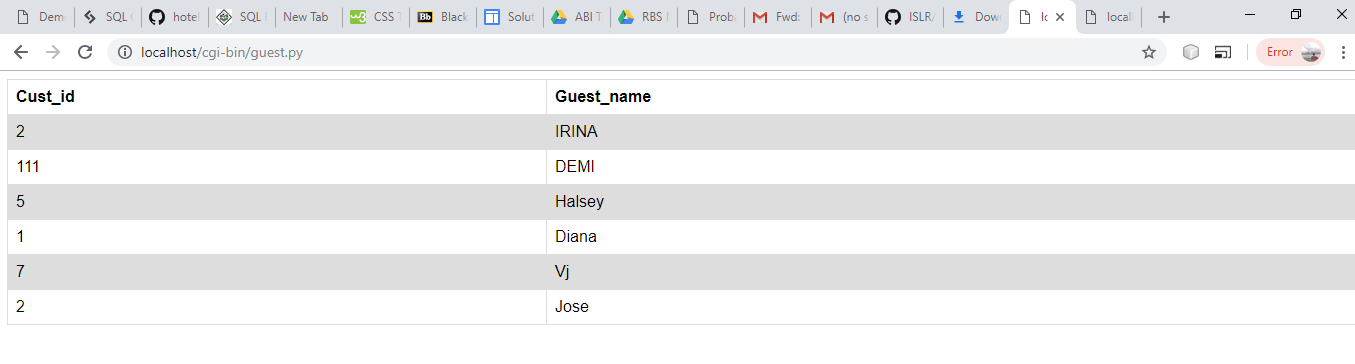
`Cust\_id` int references Customer ,

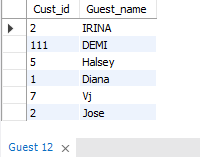
`Guest\_name` varchar(45) NOT NULL);

This is how our Guest page looks :



Below is the screenshot of our guest database created on browser:





CREATE TABLE `sts`.`Rooms`(

`Floor` int,

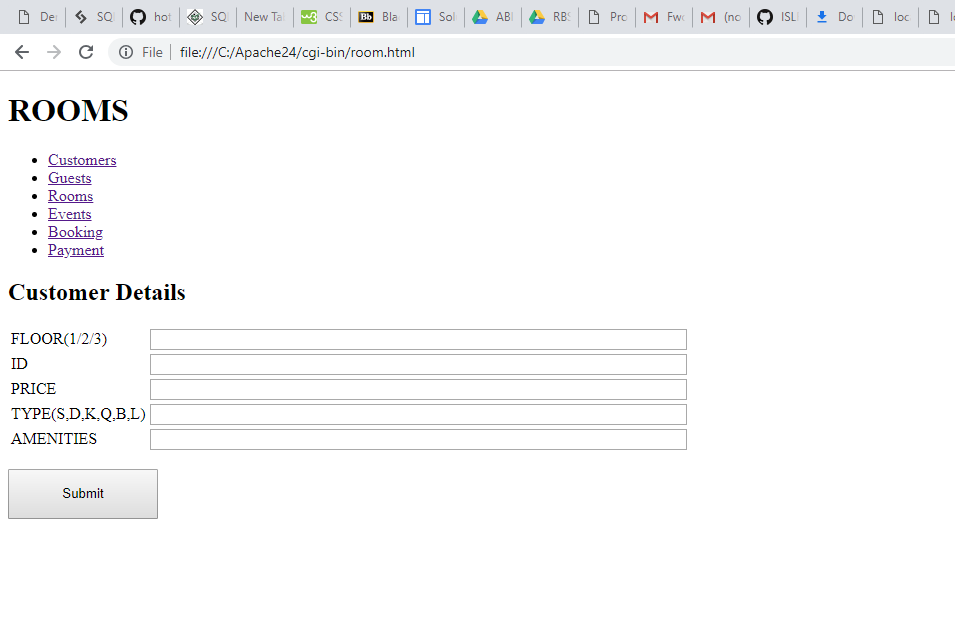
`Room\_id` int PRIMARY KEY UNIQUE NOT NULL,

`Room\_price` int ,

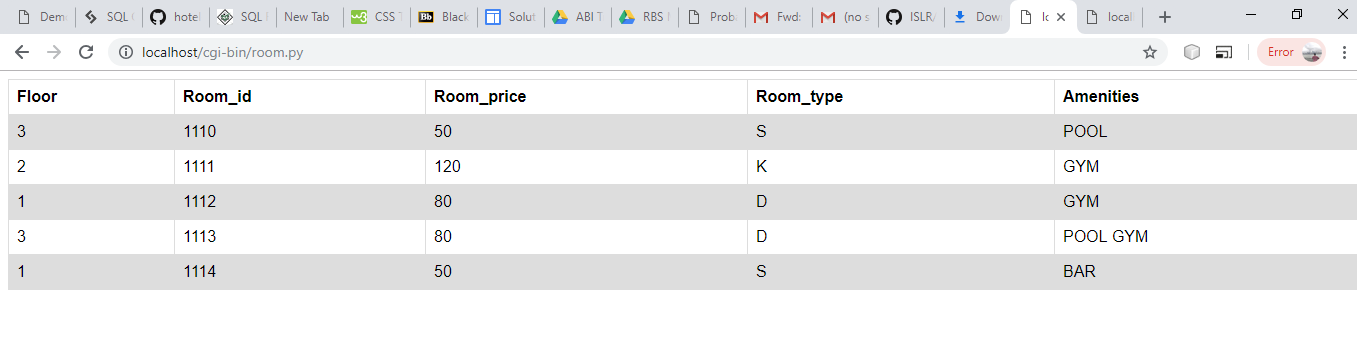
`Room\_type` varchar(45),

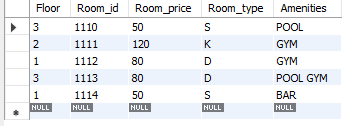
`Amenities` varchar(45));

This is how our Room page looks :



Below is the screenshot of our Room database created on browser:





CREATE TABLE `sts`.`Events`(

`Event\_type`varchar(45),

`Event\_name`varchar(45),

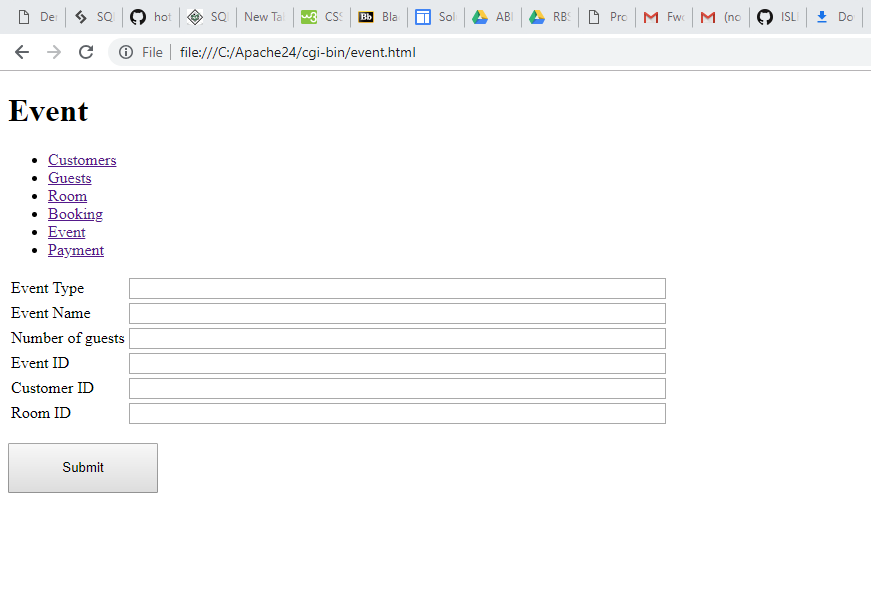
`No\_of\_part`int,

`Event\_id` int UNIQUE NOT NULL PRIMARY KEY ,

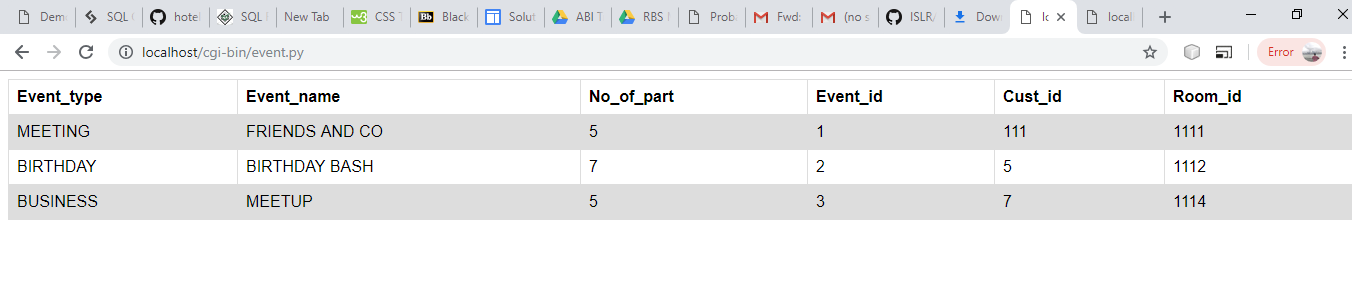
`Cust\_id` int references Customer ,

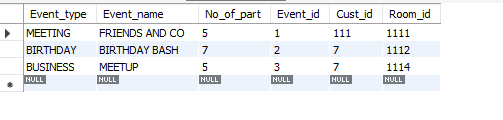
`Room\_id` int references Rooms);

This is how our Event page looks:



Below is the screenshot of our Event database created on browser:





CREATE TABLE `sts`.`Booking`(

`Book\_desc` varchar(45),

`Book\_id` int Primary key not null,

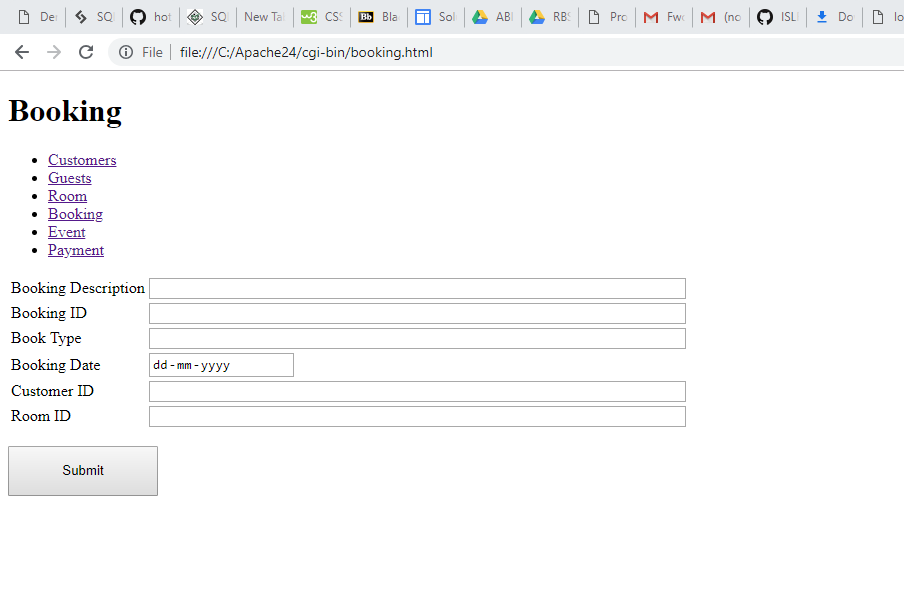
`Book\_type` varchar(45),

`Date` date,

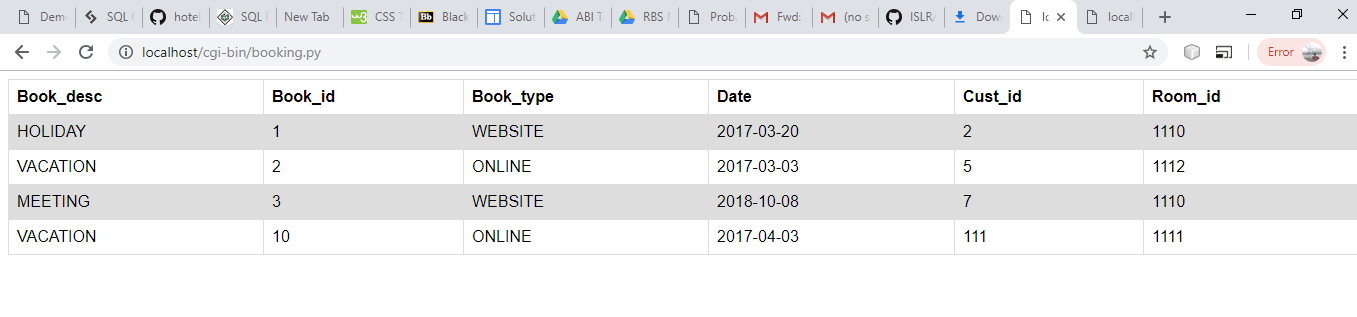
`Cust\_id` int references Customer,

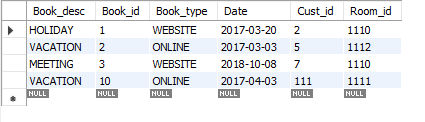
`Room\_id` int references Rooms);

This is how our Booking page looks :



Below is the screenshot of our Booking database created on browser:





CREATE TABLE `sts`.`Payment`(

`Pay\_type` varchar(45),

`Cust\_id`int references Customer,

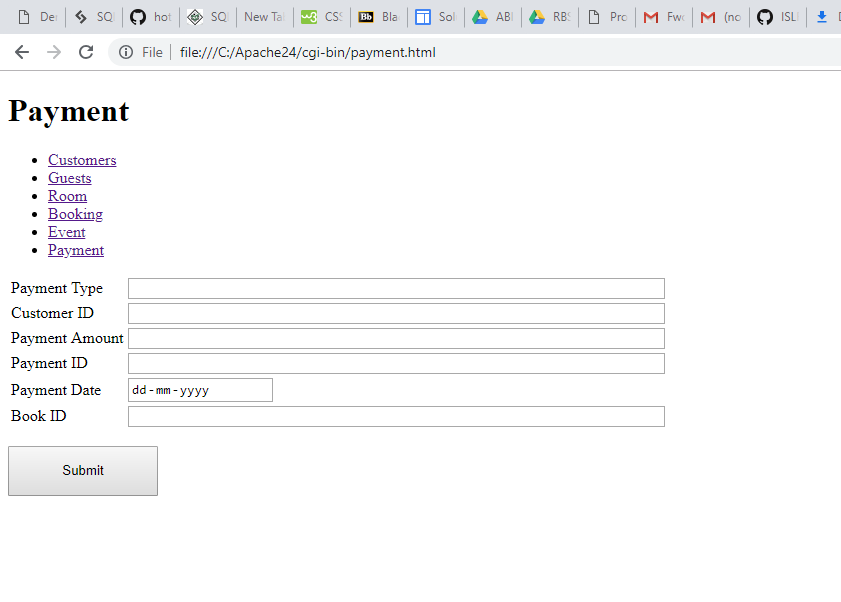
`Pay\_amount`int,

`Pay\_id` int primary key ,

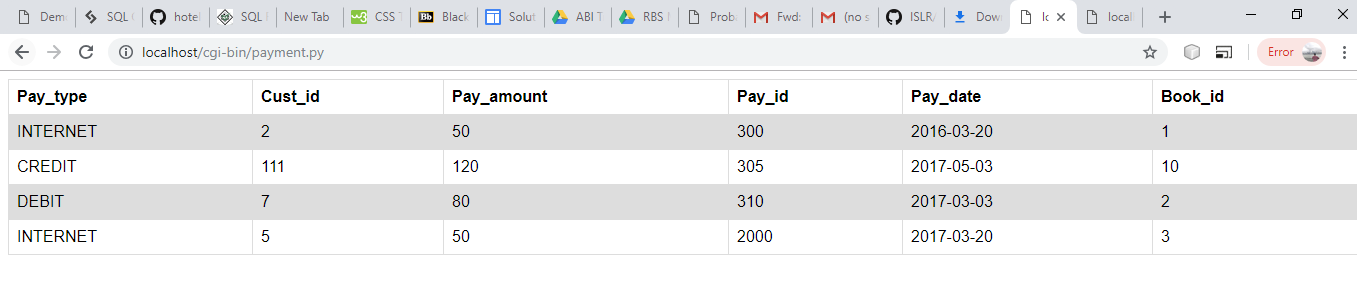
`Pay\_date` date,

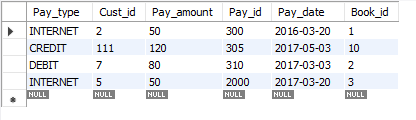
`Book\_id` int references Booking);

This is how our Payment page looks :

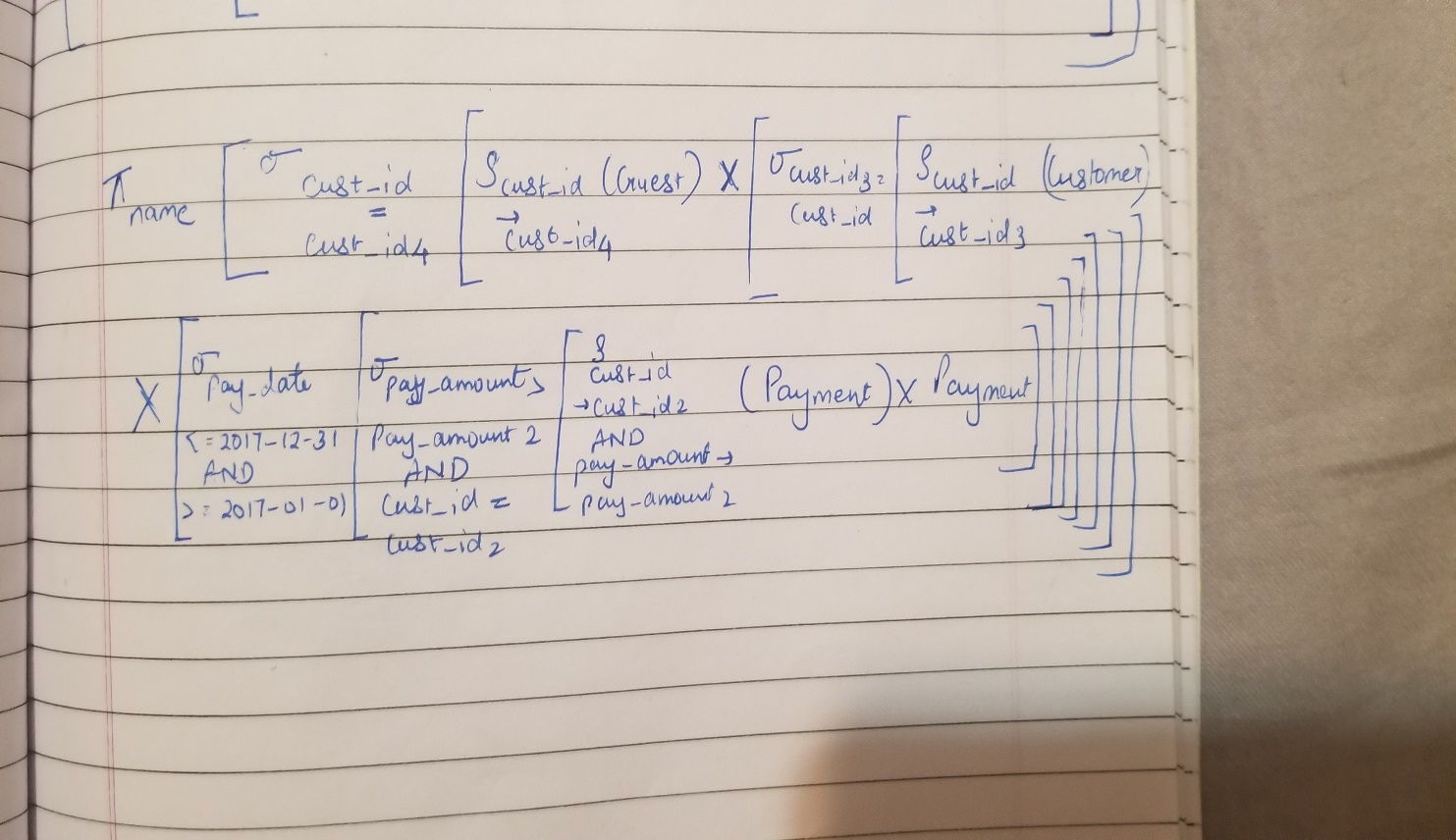


Below is the screenshot of our Payment database created on browser:





**Relational Algebra and Query –**

****

select distinct c.Cust\_name from Customer c,Payment p,Guest g

where

p.Pay\_amount in (select MAX(p2.Pay\_amount) as MAX from Payment p2

where

p2.Pay\_date between '2017-01-01' and '2017-12-31'

and

p.Cust\_id = c.Cust\_id)

and

g.Cust\_id = c.Cust\_id;

