



RESTAURANT DATABASE MANAGEMENT

BY :- HITESHREE PANCHAL 27

RIDHHI PANDYA 28

NEHA PAREPALLI 30

ANUJA PATEL 31

DESCRIPTION

- THIS RESTAURANT MANAGEMENT SYSTEM IS FOR A SMALL-SCALE RESTAURANT WITH ONLY ONE LOCATION.
- THE RESTAURANT HAS THE RECORD OF EACH EMPLOYEE WORKING IN THE RESTAURANT:
- THE EMPLOYEES HAVE DIFFERENT ROLE SUCH AS MANAGER, WAITER, CHEF AND EMPLOYEE. EMPLOYEE'S HIRE-DATE IS ALSO MENTIONED.
- THE DATABASE MUST KEEP TRACK OF THE MENU USED BY THE RESTAURANT:
- EACH ITEM HAS AN ITEM-NO. , NAME, PRICE OF THE ITEMS ,THE AMOUNT OF QUANTITY ORDERED BY EACH CUSTOMER.
- THE RESTAURANT TO KEEP A RECORD OF ITS CUSTOMERS:
- EVERY CUSTOMER HAS AN ID, NAME, ADDRESS, PHONE NUMBER. WE KEEP TRACK OF THE CUSTOMER'S ORDER.
- THE RESTAURANT HAS AN INVENTORY WHICH OVERSEES THE ALL REQUIREMENT RAW MATERIALS:
- WE KEEP TRACK OF THE ID, NAME AND QUANTITY-LEFT AND LAST-DATE-OF-ENTRY OF EACH RAW MATERIALS AVAILABLE IN THE INVENTORY.
- THE RESTAURANT NEEDS TO KEEP A RECORD OF ALL THE BILLS: EVERY CUSTOMER MUST PAY BILLS AFTER HIS/HER MEAL. EACH CUSTOMER IS GIVEN AN OPTION TO PAY THE BILL WITH DIFFERENT METHODS.

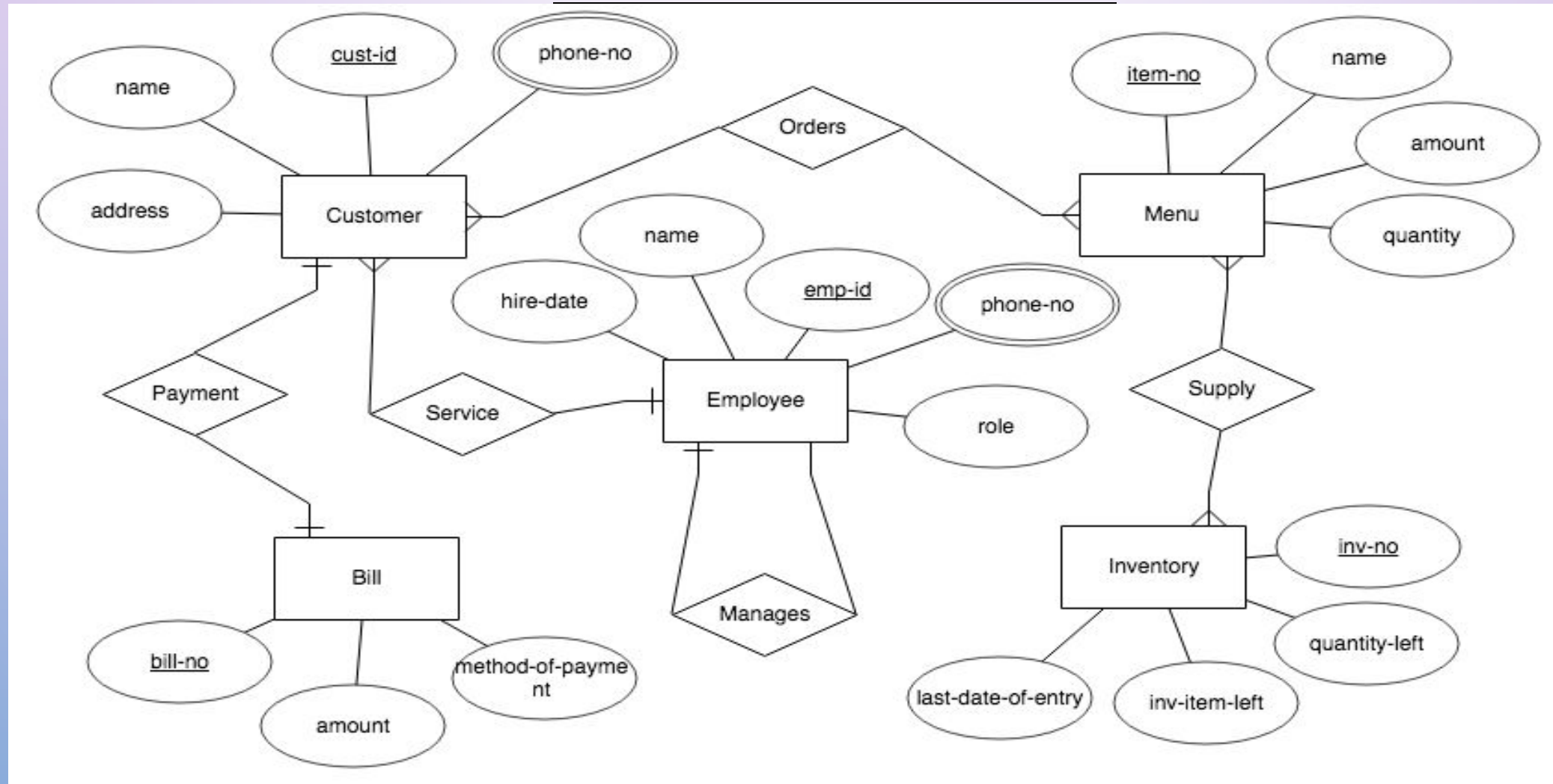
ENTITY

- EMPLOYEE (EMP-ID, NAME ,PHONE-NO ,ROLE, HIRE-DATE)
- BILL (BILL-NO ,AMOUNT , METHOD-OF-PAYMENT)
- CUSTOMER (CUSTOMER-ID, NAME ,_PHN-NO , ADDRESS)
- MENU (ITEM-NO,NAME , PRICE,QUANTITY)
- INVENTORY (INV-NO , QUANTITY-LEFT , INV-ITEM-NAME , LAST-DATE-OF-ENTRY)

RELATIONS :

- MANAGES :- 1:N MANGER WITH EMPLOYEE ; BOTH PARTIAL
- SERVICE :- 1:N CUSTOMER WITH EMPLOYEE; CUSTOMER TOTAL, EMPLOYEE PARTIAL
- PAYMENT :- 1:1 CUSTOMER WITH BILL ; BILL TOTAL ,CUSTOMER PARTIAL
- ORDERS :- M:N MENU WITH CUSTOMER ;MENU PARTIAL, CUSTOMER TOTAL
- SUPPLY :- M:N MENU WITH INVENTORY ;BOTH PARTIAL

ER DIAGRAM



RELATIONAL MODEL

- EMPLOYEE (EMP-ID, NAME ,PHONE-NO ,ROLE, HIRE- DATE,MANAGER-ID*)
- BILL (BILL-NO , AMOUNT , METHOD-OF-PAYMENT , CUSTOMER-ID*)
- CUSTOMER (CUSTOMER-ID, NAME ,_PHN-NO , ADDRESS , EMP-ID*)
- MENU (ITEM-NO, NAME , AMOUNT ,QUANTITY)
- INVENTORY (INV-NO , QUANTITY-LEFT , INV-ITEM-NAME , LAST-DATE-OF-ENTRY)
- ORDERS (ITEM-NO* , CUST-ID*)
- SUPPLY (ITEM-NO* , INV-NO*)

CODE DDL

```
drop table employee;  
drop table bill;  
drop table customer;  
drop table menu;  
drop table inventory;  
drop table orders;  
drop table supply;
```

```
create table employee(  
  emp_id numeric(20) primary  
  key,  
  emp_name varchar(50) not  
  null,  
  phn_no numeric(20),  
  hire_date date not null,  
  role varchar(50) not null,  
  manager_id nvarchar(20),  
  index(emp_id),  
  constraint fk_man_id foreign  
  key(emp_id) references  
  employee(emp_id));
```

```
create table bill(  
  bill_no numeric(20) primary  
  key,  
  amount numeric(50),  
  method_of_payment  
  varchar(30) not null,  
  customer_id numeric(20),  
  index(customer_id),  
  constraint fk_custom_id foreign  
  key(customer_id) references  
  customer(customer_id));
```

```
create table customer(  
customer_id numeric(20) primary key,  
customer_name varchar(50) not null,  
phn_no numeric(20),  
address varchar(80) not null,  
emp_id numeric(20),  
index(emp_id),  
constraint fk_employ_id foreign  
key(emp_id) references  
employee(emp_id));
```

```
create table menu(  
item_no numeric(30) primary key,  
item_name varchar(50) not null,  
amount numeric(50,2) not null,  
quantity numeric(50));
```

```
create table inventory(  
inv_no numeric(50) primary key,  
quantity_left numeric(50) not null,  
inv_item_name varchar(50) not null,  
last_date_of_entry date);
```

```
create table orders(  
item_no numeric(30),  
customer_id numeric(20),  
index(item_no),  
index(customer_id),  
constraint pk_orders primary  
key(item_no,customer_id),  
constraint fk_iteem_no foreign  
key(item_no) references  
menu(item_no) on delete cascade,  
constraint fk_cusmer_id foreign  
key(customer_id) references  
customer(customer_id));
```

```
create table supply(  
item_no numeric(30),  
inv_no numeric(50),  
index(item_no),  
index(inv_no),  
constraint pk_supply primary  
key(item_no,inv_no),  
constraint fk_itemmm_no foreign  
key(item_no) references  
menu(item_no) on delete cascade,  
constraint fk_inv_no foreign  
key(inv_no) references  
inventory(inv_no));
```


CODE DML

```
insert into bill values (5, 929,  
'google_pay', 6160390);  
insert into bill values (2, 941, 'debit',  
6734826);  
insert into bill values (7, 681,  
'google_pay', 7238390);  
insert into bill values (4, 387,  
'google_pay', 6003358);  
insert into bill values (3, 216, 'cash',  
5816967);  
insert into bill values (10, 638, 'credit',  
5989537);  
insert into bill values (8, 369, 'paytm',  
6615632);  
insert into bill values (9, 762, 'debit',  
6500977);
```

```
insert into bill values (1, 269, 'cash',  
6645127);  
insert into bill values (6, 185, 'credit',  
6310256);  
  
insert into customer values (6160390,  
'Eve', 9902055540,  
'E-20_Swami_Road_Virar', 5980172);  
insert into customer values (6734826,  
'Avijit', 9903028079,  
'E-65_Bhatia_Road_Dahisar',  
7156878);  
insert into customer values (7238390,  
'Santosh', 9949380158,  
'F-328_Vajira_Andheri', 6704120);  
insert into customer values (6003358,  
'Gautam', 9938208823,  
'E-65_Bhatia_Road_Dahisar',  
5762513);
```

```
insert into customer values (5816967,  
'Shubham', 9941434328,  
'A-32_Yam_Road_Dadar', 5846757);  
insert into customer values (5989537,  
'Deepak', 9940917415,  
'F-328_Vajira_Andheri', 6404075);  
insert into customer values (6615632,  
'Jill', 9950235956,  
'A-520_Balaji_road_Bandra',  
7150565);  
insert into customer values (6500977,  
'Rakesh', 9920769189,  
'A-57_Ambadi_road_Vasai',  
6474086);  
insert into customer values (6645127,  
'Jack', 9893392469,  
'B-43_Navghar_Road_Parle',  
5683223);  
insert into customer values (6310256,
```

insert into employee values (5980172, 'Rachit', 5453372364, '2019/10/20', 'cashier', 'null');
insert into employee values (7156878, 'Anuja', 6345414720, '2020/03/01', 'manager', 7156878);
insert into employee values (6704120, 'Neha', 2011340305, '2019/06/25', 'waiter', 'null');
insert into employee values (5762513, 'Rishi', 6273820597, '2019/12/23', 'employee', 'null');
insert into employee values (5846757, 'Vikrant', 2449117256, '2019/04/05', 'chef', 'null');
insert into employee values (6404075, 'Riddhi', 8559449516, '2019/10/18', 'cashier', 'null');
insert into employee values (7150565, 'Atharva', 3996501490, '2019/04/02', 'manager', 7150565);

insert into employee values (6474086, 'Dhairya', 4464939775, '2019/08/29', 'waiter', 'null');
insert into employee values (5683223, 'Hiteshree', 2233514132, '2019/08/08', 'employee', 'null');
insert into employee values (6604861, 'Diksha', 5849987454, '2020/02/20', 'chef', 'null');

insert into inventory values (4, 35, 'Potato', '2020/03/08');
insert into inventory values (26, 17, 'Bread', '2020/02/06');
insert into inventory values (5, 6, 'Onion', '2020/02/11');
insert into inventory values (10, 7, 'Spinach', '2020/03/06');
insert into inventory values (27, 4, 'Tomato', '2020/02/15');

insert into inventory values (60, 34, 'Cheese', '2020/02/29');
insert into inventory values (11, 24, 'Butter', '2020/01/15');
insert into inventory values (17, 20, 'Baby_Corn', '2020/03/14');
insert into inventory values (14, 15, 'Carrot', '2020/03/18');
insert into inventory values (30, 40, 'Capsicum', '2020/03/18');

insert into menu values (8, 'Dum_Aloo', 809.49, 2);
insert into menu values (19, 'Pizza', 160.95, 5);
insert into menu values (34, 'Pasta', 415.9, 8);
insert into menu values (36, 'Sandwich', 240.27, 1);

```
insert into menu values (15, 'Sizzler',  
542.79, 4);  
insert into menu values (3, 'Manchurian',  
863.21, 3);  
insert into menu values (9, 'Fried_Rice',  
848.65, 4);  
insert into menu values (40,  
'Shahi_Paneer', 343.28, 2);  
insert into menu values (17, 'Burger',  
490.06, 3);  
insert into menu values (20,  
'French_Fries', 844.09, 9);
```

```
insert into orders values (8, 6160390);  
insert into orders values (19,  
6734826);  
insert into orders values (34,  
7238390);  
insert into orders values (36,  
6003358);
```

```
insert into orders values (15,  
5816967);  
insert into orders values (3, 5989537);  
insert into orders values (9, 6615632);  
insert into orders values (40,  
6500977);  
insert into orders values (17,  
6645127);  
insert into orders values (20,  
6310256);
```

```
insert into supply values (8, 4);  
insert into supply values (19, 26);  
insert into supply values (34, 5);  
insert into supply values (36, 10);  
insert into supply values (15, 27);  
insert into supply values (3, 60);  
insert into supply values (9, 11);  
insert into supply values (40, 17);  
insert into supply values (17, 14);  
insert into supply values (20, 30);  
  
commit;
```

TABLES

```
mysql> select * from bill;
```

bill_no	amount	method_of_payment	customer_id
1	269	cash	6645127
2	941	debit	6734826
3	216	cash	5816967
4	387	google_pay	6003358
5	929	google_pay	6160390
6	185	credit	6310256
7	681	google_pay	7238390
8	369	paytm	6615632
9	762	debit	6500977
10	638	credit	5989537

```
10 rows in set (0.00 sec)
```

```
mysql> select * from supply;
```

item_no	inv_no
3	60
8	4
9	11
15	27
17	14
19	26
20	30
34	5
36	10
40	17

```
10 rows in set (0.00 sec)
```



```
mysql> select * from customer;
```

customer_id	customer_name	phn_no	address	emp_id
5816967	Shubham	9941434328	A-32_Yam_Road_Dadar	5846757
5989537	Deepak	9940917415	F-328_Vajira_Andheri	6404075
6003358	Gautam	9938208823	E-65_Bhatia_Road_Dahisar	5762513
6160390	Eve	9902055540	E-20_Swami_Road_Virar	5980172
6310256	Eve	9950906733	A-32_Yam_Road_Dadar	6604861
6500977	Rakesh	9920769189	A-57_Ambadi_road_Vasai	6474086
6615632	Jill	9950235956	A-520_Balaji_road_Bandra	7150565
6645127	Jack	9893392469	B-43_Navghar_Road_Parle	5683223
6734826	Avijit	9903028079	E-65_Bhatia_Road_Dahisar	7156878
7238390	Santosh	9949380158	F-328_Vajira_Andheri	6704120

10 rows in set (0.00 sec)

```
mysql> select * from menu;
```

item_no	item_name	amount	quantity
3	Manchurian	863.21	3
8	Dum_Aloo	809.49	2
9	Fried_Rice	848.65	4
15	Sizzler	542.79	4
17	Burger	490.06	3
19	Pizza	160.95	5
20	French_Fries	844.09	9
34	Pasta	415.90	8
36	Sandwich	240.27	1
40	Shahi_Paneer	343.28	2

10 rows in set (0.00 sec)

```
mysql> select * from employee;
```

emp_id	emp_name	phn_no	hire_date	role	manager_id
5683223	Hiteshree	2233514132	2019-08-08	employee	null
5762513	Rishi	6273820597	2019-12-23	employee	null
5846757	Vikrant	2449117256	2019-04-05	chef	null
5980172	Rachit	5453372364	2019-10-20	cashier	null
6404075	Riddhi	8559449516	2019-10-18	cashier	null
6474086	Dhairya	4464939775	2019-08-29	waiter	null
6604861	Diksha	5849987454	2020-02-20	chef	null
6704120	Neha	2011340305	2019-06-25	waiter	null
7150565	Atharva	3996501490	2019-04-02	manager	7150565
7156878	Anuja	6345414720	2020-03-01	manager	7156878

10 rows in set (0.00 sec)

```
mysql> select * from orders;
```

item_no	customer_id
15	5816967
3	5989537
36	6003358
8	6160390
20	6310256
40	6500977
9	6615632
17	6645127
19	6734826
34	7238390

10 rows in set (0.00 sec)


```
mysql> select * from inventory;
```

inv_no	quantity_left	inv_item_name	last_date_of_entry
4	35	Potato	2020-03-08
5	6	Onion	2020-02-11
10	7	Spinach	2020-03-06
11	24	Butter	2020-01-15
14	15	Carrot	2020-03-18
17	20	Baby_Corn	2020-03-14
26	17	Bread	2020-02-06
27	4	Tomato	2020-02-15
30	40	Capsicum	2020-03-18
60	34	Cheese	2020-02-29

```
10 rows in set (0.00 sec)
```

SQL QUERIES

1) Show the bill details of the customers who has their amount greater than Rs.300

```
mysql> select *  
      -> from bill  
      -> where amount > 300;
```

bill_no	amount	method_of_payment	customer_id
2	941	debit	6734826
4	387	google_pay	6003358
5	929	google_pay	6160390
7	681	google_pay	7238390
8	369	paytm	6615632
9	762	debit	6500977
10	638	credit	5989537

```
7 rows in set (0.04 sec)
```

Change the price of Pizza to Rs 200.

```
mysql> update menu
```

```
-> set amount=200
```

```
-> where item_name="Pizza";
```

```
Query OK, 1 row affected (0.18 sec)
```

```
Rows matched: 1 Changed: 1 Warnings: 0
```

BEFORE

```
mysql> select * from menu;
```

item_no	item_name	amount	quantity
3	Manchurian	863.21	3
8	Dum_Aloo	809.49	2
9	Fried_Rice	848.65	4
15	Sizzler	542.79	4
17	Burger	490.06	3
19	Pizza	160.95	5
20	French_Fries	844.09	9
34	Pasta	415.90	8
36	Sandwich	240.27	1
40	Shahi_Paneer	343.28	2

```
10 rows in set (0.00 sec)
```

AFTER

```
mysql> select * from menu;
```

item_no	item_name	amount	quantity
3	Manchurian	863.21	3
8	Dum_Aloo	809.49	2
9	Fried_Rice	848.65	4
15	Sizzler	542.79	4
17	Burger	490.06	3
19	Pizza	200.00	5
20	French_Fries	844.09	9
34	Pasta	415.90	8
36	Sandwich	240.27	1
40	Shahi_Paneer	343.28	2
64	Pav Bhaji	200.00	2

```
11 rows in set (0.00 sec)
```


Delete the records of Onion from restaurant.

```
mysql> delete from supply
-> where inv_no in
-> (select inv_no
-> from inventory
-> where inv_item_name="Onion");
Query OK, 1 row affected (0.19 sec)
```

```
mysql> delete from inventory
-> where inv_item_name="Onion";
Query OK, 1 row affected (0.16 sec)
```

```
mysql> select * from inventory;
```

inv_no	quantity_left	inv_item_name	last_date_of_entry
4	35	Potato	2020-03-08
10	7	Spinach	2020-03-06
11	24	Butter	2020-01-15
14	15	Carrot	2020-03-18
17	20	Baby_Corn	2020-03-14
26	17	Bread	2020-02-06
27	4	Tomato	2020-02-15
30	40	Capsicum	2020-03-18
60	34	Cheese	2020-02-29

9 rows in set (0.04 sec)

```
mysql> select * from supply;
```

item_no	inv_no
3	60
8	4
9	11
15	27
17	14
19	26
20	30
36	10
40	17

9 rows in set (0.03 sec)

Show all the customers names who has "a" letter in their name.

```
mysql> select customer_name  
-> from customer  
-> where customer_name like '%a%';
```

customer_name
Shubham
Deepak
Gautam
Rakesh
Jack
Avijit
Santosh

7 rows in set (0.05 sec)

CONCLUSION

- FROM THIS PROJECT WE LEARNED HOW TO OPERATE MYSQL.
- HOW TO WRITE THE CODE TO FORM A TABLE AND INPUT DATA WAS ALSO LEARNED.
- WE ATTAINED DEEPER KNOWLEDGE OF DBMS WITH THE HELP OF THIS PROJECT.