



## **SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

### **CSE2004 – Database Management System**

**Fall Sem 2019-2020**

#### **Project**

#### **Team Members:**

1. *SAHARSH PODDAR 18BCE0266*
2. *SAYAK MAJI 18BCE0261*

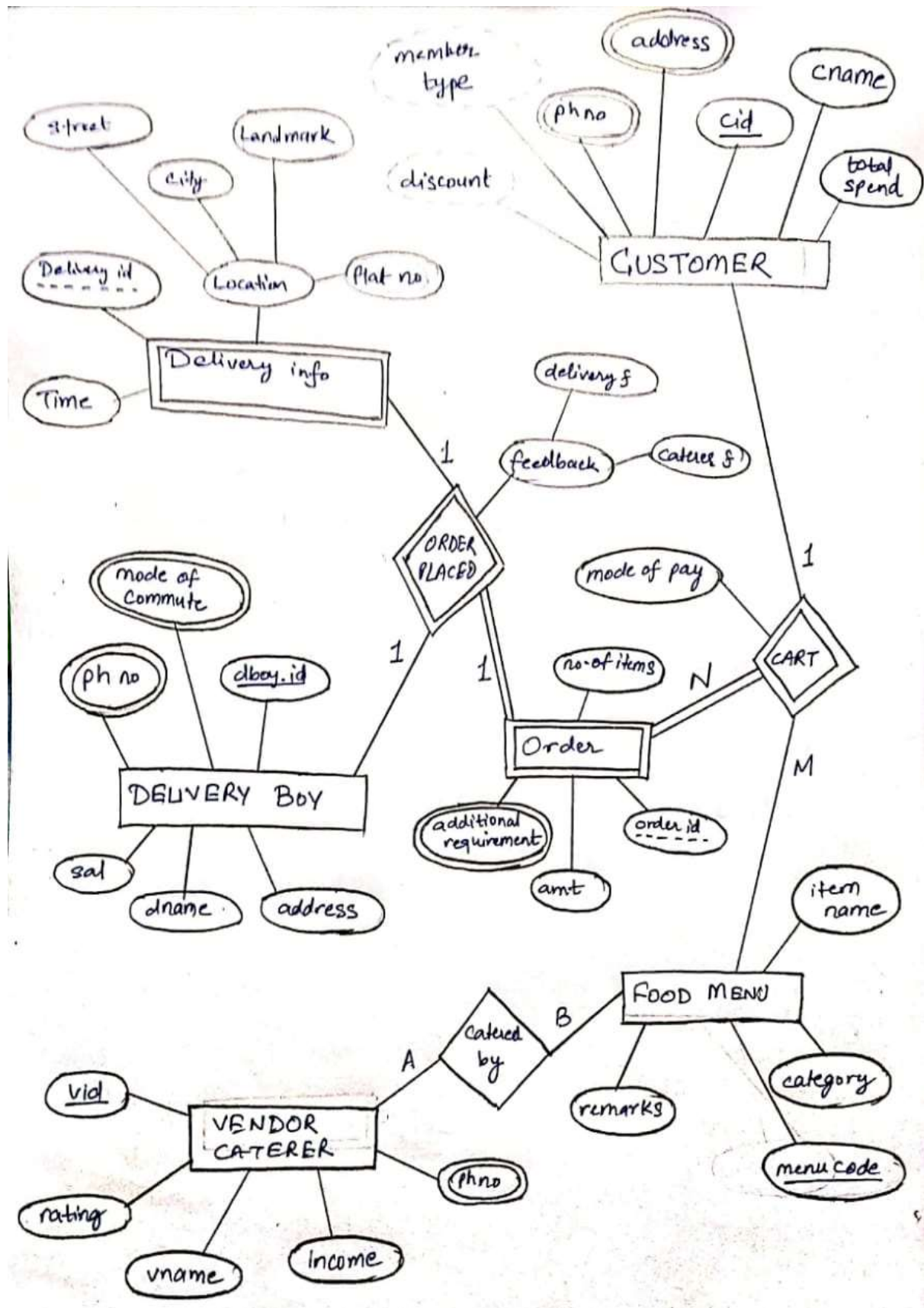
#### **Real world problem:**

Designing a database for food start-up consisting of street vendors and delivery person at the sellers' end and customers. The database would be required to maintain the records of the sale, delivery details, cart of the customer and the food menu for different caterers.

## **Requirements:**

1. The application provides login as a customer and a caterer.
2. There is an option for sign up for new customers to be able to use the services and place orders.
3. The application should allow the customer to view the food menu of different caterers.
4. The customer can use filtered search to view the menu based on specific categorisation(s) such as veg/non-veg, caterer name, price range, bestsellers/recommended etc.
5. The customer also has a facility to provide any additional requirements he/she wants to be fulfilled. (e.g. extra cheese, extra spicy, less spicy etc.)
6. Customer can provide feedback on the food quality (caterer feedback) as well as feedback regarding the delivery.
7. The customer can opt to make the payment via online transaction, debit/credit card or choose COD(cash on delivery).
8. The net amount of the customer's invoice incorporates discount based upon the membership type of the customer. The membership type depends on the total expenditure made by the customer till date.
9. The caterer has the options to make modifications in the food menu.
10. The application is a user-friendly interface which interacts with the database to yield desired results.

## ER Diagram:



## **ER Relational Model:**

### *Strong entities:*

customer(cid, cname, tot\_spend)

food\_menu(menu\_code, item\_name, category, remarks)

vendor(vid, vname, income, rating)

dboy(dboy\_id, dname, sal, address)

### *Weak entities:*

orders(no\_of\_items, amount, order\_id, cid, dboy\_id, menu\_code)

dinfo(city, street, lanmark, flat\_no, delivery\_id, order\_id, dboy\_id)

### *Multi-valued attributes:*

customer\_phno(cid, phno)

vendor\_phno(vid, phno)

dboy\_phno(dboy\_id, phno)

customer\_address(cid, address)

mode\_of\_commute(dboy\_id, moc)

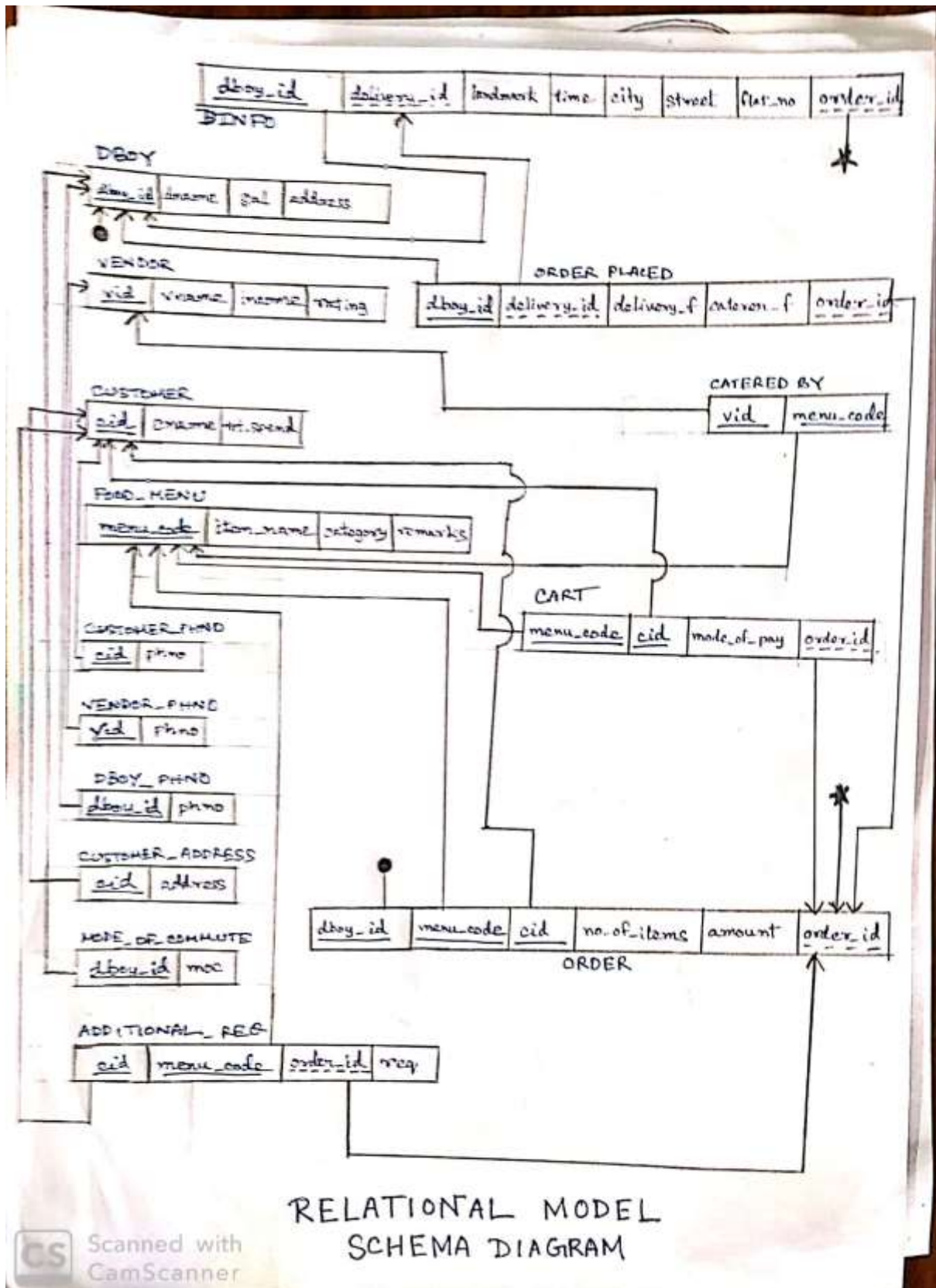
additional\_req(order\_id, cid, menu\_code, requirements)

### *Multi-nary relations:*

cart(mode\_of\_pay, cid, menu\_code, order\_id)

order\_placed(caterer\_f, delivery\_f, dboy\_id, delivery\_id, order\_id)

catered\_by(vid, menu\_code)

Schema Diagram:

## Sample Code:

```
SQL> create table customer(cid number(5),cname varchar(15),tot_spend number(7));  
Table created.
```

```
SQL> create table fmenu(mcode number(5),iname varchar(15),cat char(1),remarks varchar(15));  
Table created.  
SQL> create table vendor(vid number(5),vname varchar(15),income number(7),rating number(2));  
Table created.  
SQL> create table dboy(dboy_id number(5),dname varchar(15),sal number(7),address varchar(15));  
Table created.
```

```
SQL> alter table cust add constraint pk_cid primary key(cid);  
Table altered.  
SQL> alter table fmenu add constraint pk_mcode primary key(mcode);  
Table altered.  
SQL> alter table vendor add constraint pk_vid primary key(vid);  
Table altered.  
SQL> alter table dboy add constraint pk_dboy_id primary key(dboy_id);  
Table altered.
```

```
SQL> alter table cust modify (cname varchar(15) constraint nn_cname not null);  
Table altered.  
SQL> alter table fmenu modify (iname varchar(15) constraint nn_iname not null);  
Table altered.  
SQL> alter table vendor modify income number(7) default 5000;  
Table altered.  
SQL> alter table dboy modify sal number(7) default 2000;  
Table altered.
```

```
SQL> desc cust;
Name                                     Null?      Type
-----
CID                                     NOT NULL   NUMBER(5)
CNAME                                  NOT NULL   VARCHAR2(15)
TOT_SPEND                              NULL       NUMBER(7)

SQL> desc fmenu;
Name                                     Null?      Type
-----
MCODE                                  NOT NULL   NUMBER(5)
INAME                                  NOT NULL   VARCHAR2(15)
CAT                                    NULL       CHAR(1)
REMARKS                               NULL       VARCHAR2(15)

SQL> desc vendor;
Name                                     Null?      Type
-----
VID                                    NOT NULL   NUMBER(5)
VNAME                                  NULL       VARCHAR2(15)
INCOME                                NULL       NUMBER(7)
RATING                                NULL       NUMBER(2)

SQL> desc dboy;
Name                                     Null?      Type
-----
DBOY_ID                               NOT NULL   NUMBER(5)
DNAME                                  NULL       VARCHAR2(15)
SAL                                    NULL       NUMBER(7)
ADDRESS                               NULL       VARCHAR2(15)
```

```
SQL> alter table cust_ph add cid number(5);
Table altered.

SQL> alter table cust_ph add constraint fk_cid_cust_ph foreign key(cid) references cust(cid) on delete cascade;
Table altered.

SQL> alter table vendor_ph add vid number(5);
Table altered.

SQL> alter table vendor_ph add constraint fk_vid_vendor_ph foreign key(vid) references vendor(vid) on delete cascade;
Table altered.

SQL> alter table dboy_ph add dboy_id number(5);
Table altered.

SQL> alter table dboy_ph add constraint fk_dboy_id_dboy_ph foreign key(dboy_id) references dboy(dboy_id) on delete cascade;
Table altered.
```

```
SQL> alter table cust_addr add cid number(5);
Table altered.
```



```
SQL> alter table cust_addr add constraint fk_cid_cust_addr foreign key(cid) references cust(cid) on delete cascade;
Table altered.

SQL> alter table mode_of_comm add dboy_id number(5);
Table altered.

SQL> alter table mode_of_comm add constraint fk_dboy_id_mode_of_comm foreign key(dboy_id) references dboy(dboy_id) on delete cascade;
Table altered.
```

```
SQL> alter table cust_ph add constraint pk_cid_cust_ph primary key(cid);
Table altered.

SQL> alter table vendor_ph add constraint pk_vid_vendor_ph primary key(vid);
Table altered.

SQL> alter table dboy_ph add constraint pk_dboy_id_dboy_ph primary key(dboy_id);
Table altered.

SQL> alter table cust_addr add constraint pk_cid_cust_addr primary key(cid);
Table altered.

SQL> alter table mode_of_comm add constraint pk_dboy_id_mode_of_comm primary key(dboy_id);
Table altered.
```

```
SQL> desc cust_ph;
Name                               Null?      Type
-----
CPHNO                              VARCHAR2(10)
CID                                NOT NULL   NUMBER(5)

SQL> desc vendor_ph;
Name                               Null?      Type
-----
VPHNO                              VARCHAR2(10)
VID                                NOT NULL   NUMBER(5)

SQL> desc dboy_ph;
Name                               Null?      Type
-----
DPHNO                              VARCHAR2(10)
DBOY_ID                            NOT NULL   NUMBER(5)

SQL> desc cust_addr;
Name                               Null?      Type
-----
ADDRESS                            VARCHAR2(15)
CID                                NOT NULL   NUMBER(5)

SQL> desc mode_of_comm;
Name                               Null?      Type
-----
MOC                                VARCHAR2(15)
DBOY_ID                            NOT NULL   NUMBER(5)
```



```

SQL> alter table orders add order_id number(5);
Table altered.

SQL> alter table orders add cid number(5);
Table altered.

SQL> alter table orders add mcode number(5);
Table altered.

SQL> alter table orders add dboy_id number(5);
Table altered.

SQL> alter table orders add constraint fk_dboy_id_orders foreign key(dboy_id) references dboy(dboy_id) on delete cascade;
Table altered.

SQL> alter table orders add constraint fk_cid_orders foreign key(cid) references cust(cid) on delete cascade;
Table altered.

SQL> alter table orders add constraint fk_mcode_orders foreign key(mcode) references fmenu(mcode) on delete cascade;
Table altered.

SQL> alter table orders add constraint pk_order_id primary key(order_id);
Table altered.

```

```

SQL> alter table dinfo add order_id number(5);
Table altered.

SQL> alter table dinfo add dboy_id number(5);
Table altered.

SQL> alter table dinfo add del_id number(5);
Table altered.

SQL> alter table dinfo add constraint fk_dboy_id_dinfo foreign key(dboy_id) references dboy(dboy_id) on delete cascade;
Table altered.

SQL> alter table dinfo add constraint fk_order_id_dinfo foreign key(order_id) references orders(order_id) on delete cascade;
Table altered.

SQL> alter table dinfo add constraint pk_del_id primary key(del_id);
Table altered.

```

```

SQL> desc dinfo;

```

Name	Null?	Type
CITY		VARCHAR2(15)
STREET		VARCHAR2(15)
LANDMARK		VARCHAR2(15)
FLAT_NO		VARCHAR2(5)
ORDER_ID		NUMBER(5)
DBOY_ID		NUMBER(5)
DEL_ID	NOT NULL	NUMBER(5)

```
SQL> create table add_req(req varchar(15),cid number(5),mcode number(5),order_id number(5));
Table created.

SQL> alter table add_req add constraint fk_order_id_add_req foreign key(order_id) references orders(order_id) on delete cascade;
Table altered.

SQL> alter table add_req add constraint fk_mcode_add_req foreign key(mcode) references fmenu(mcode) on delete cascade;
Table altered.

SQL> alter table add_req add constraint fk_cid_add_req foreign key(cid) references cust(cid) on delete cascade;
Table altered.
```

```
SQL> alter table add_req add constraint pk_cid_add_req primary key(cid);
```

Table altered.

```
SQL> desc add_req;
```

Name	Null?	Type
REQ		VARCHAR2(15)
CID	NOT NULL	NUMBER(5)
MCODE		NUMBER(5)
ORDER_ID		NUMBER(5)

```
SQL> alter table add_req drop constraint pk_cid_add_req;
```

Table altered.

```
SQL> desc add_req;
```

Name	Null?	Type
REQ		VARCHAR2(15)
CID		NUMBER(5)
MCODE		NUMBER(5)
ORDER_ID		NUMBER(5)

```
SQL> alter table add_req add constraint pk_order_id_add_req primary key(order_id);
```

Table altered.

```
SQL> create table cart(mop varchar(15),cid number(5),mcode number(5),order_id number(5));
```

Table created.

```
SQL> alter table cart add constraint fk_mcode_cart foreign key(mcode) references fmenu(mcode) on delete cascade;
```

Table altered.

```
SQL> alter table cart add constraint fk_cid_cart foreign key(cid) references cust(cid) on delete cascade;
```

Table altered.

```
SQL> alter table cart add constraint fk_order_id_cart foreign key(order_id) references orders(order_id) on delete cascade;
```

Table altered.

```
SQL> alter table cart add constraint pk_order_id_cart primary key(order_id);
Table altered.

SQL> desc cart;

```

Name	Null?	Type
MOP		VARCHAR2(15)
CID		NUMBER(5)
MCODE		NUMBER(5)
ORDER_ID	NOT NULL	NUMBER(5)

```
SQL> create table catered_by(vid number(5),mcode number(5));
Table created.

SQL> alter table catered_by add constraint fk_mcode_catered_by foreign key(mcode) references fmenu(mcode) on delete cascade;
Table altered.

SQL> alter table catered_by add constraint fk_vid_catered_by foreign key(vid) references vendor(vid) on delete cascade;
Table altered.

SQL> alter table catered_by add constraint pk_vid_catered_by primary key(vid);
Table altered.

SQL> desc catered_by;

```

Name	Null?	Type
VID	NOT NULL	NUMBER(5)
MCODE		NUMBER(5)

```
SQL> create table order_placed(caterer_f varchar(15),del_f varchar(15),dboy_id number(5),del_id number(5),order_id number(5));
Table created.

SQL> alter table order_placed add constraint fk_order_id_order_placed foreign key(order_id) references orders(order_id) on delete cascade;
Table altered.

SQL> alter table order_placed add constraint fk_dboy_id_order_placed foreign key(dboy_id) references dboy(dboy_id) on delete cascade;
Table altered.

SQL> alter table order_placed add constraint fk_del_id_order_placed foreign key(del_id) references dinfo(del_id) on delete cascade;
Table altered.

SQL> alter table order_placed add constraint pk_del_id_catered_by primary key(del_id);
Table altered.

SQL> desc order_placed;

```

Name	Null?	Type
CATERER_F		VARCHAR2(15)
DEL_F		VARCHAR2(15)
DBOY_ID		NUMBER(5)
DEL_ID	NOT NULL	NUMBER(5)
ORDER_ID		NUMBER(5)

```
SQL> commit;
Commit complete.
```

```
SQL> desc cust;
Name                                         Null?    Type
-----
CID                                           NOT NULL NUMBER(5)
CNAME                                         NOT NULL VARCHAR2(15)
TOT_SPEND                                     NUMBER(7)

SQL> desc fmenu;
Name                                         Null?    Type
-----
MCODE                                         NOT NULL NUMBER(5)
INAME                                         NOT NULL VARCHAR2(15)
CAT                                           CHAR(1)
REMARKS                                       VARCHAR2(15)

SQL> desc vendor;
Name                                         Null?    Type
-----
VID                                           NOT NULL NUMBER(5)
VNAME                                         VARCHAR2(15)
INCOME                                       NUMBER(7)
RATING                                       NUMBER(2)

SQL> desc dboy;
Name                                         Null?    Type
-----
DBOY_ID                                       NOT NULL NUMBER(5)
DNAME                                         VARCHAR2(15)
SAL                                           NUMBER(7)
ADDRESS                                       VARCHAR2(15)
```

```
SQL> desc orders;
Name                                         Null?    Type
-----
NO_ITEMS                                     NUMBER(3)
AMT                                           NUMBER(5)
ORDER_ID                                       NOT NULL NUMBER(5)
CID                                           NUMBER(5)
MCODE                                         NUMBER(5)
DBOY_ID                                       NUMBER(5)

SQL> desc dinfo;
Name                                         Null?    Type
-----
CITY                                           VARCHAR2(15)
STREET                                         VARCHAR2(15)
LANDMARK                                       VARCHAR2(15)
FLAT_NO                                       VARCHAR2(5)
ORDER_ID                                       NUMBER(5)
DBOY_ID                                       NUMBER(5)
DEL_ID                                       NOT NULL NUMBER(5)
```

```
SQL> desc cart;
```

Name	Null?	Type
MOP		VARCHAR2(15)
CID		NUMBER(5)
MCODE		NUMBER(5)
ORDER_ID	NOT NULL	NUMBER(5)

```
SQL> desc order_placed;
```

Name	Null?	Type
CATERER_F		VARCHAR2(15)
DEL_F		VARCHAR2(15)
DBOY_ID		NUMBER(5)
DEL_ID	NOT NULL	NUMBER(5)
ORDER_ID		NUMBER(5)

```
SQL> desc catered_by;
```

Name	Null?	Type
VID	NOT NULL	NUMBER(5)
MCODE		NUMBER(5)

```
SQL> desc cust_ph;
```

Name	Null?	Type
CPHNO		VARCHAR2(10)
CID	NOT NULL	NUMBER(5)

```
SQL> desc vendor_ph;
```

Name	Null?	Type
VPHNO		VARCHAR2(10)
VID	NOT NULL	NUMBER(5)

```
SQL> desc dboy_ph;
```

Name	Null?	Type
DPHNO		VARCHAR2(10)
DBOY_ID	NOT NULL	NUMBER(5)

```
SQL> desc cust_addr;
```

Name	Null?	Type
ADDRESS		VARCHAR2(15)
CID	NOT NULL	NUMBER(5)

```
SQL> desc add_req;
```

Name	Null?	Type
REQ		VARCHAR2(15)
CID		NUMBER(5)
MCODE		NUMBER(5)
ORDER_ID	NOT NULL	NUMBER(5)

```
SQL> desc mode_of_comm;
```

Name	Null?	Type
MOC		VARCHAR2(15)
DBOY_ID	NOT NULL	NUMBER(5)

**Front-end:**

```

.....Welcome.....

What do you identify yourself as?....

1.Customer
2.Vendor
3.Delivery boy
4.Exit

Enter your choice:
1
Username:safal
Enter id:3
('safal',)
What do you want to do?....

1.Edit details
2.View menu
3.Access cart
4.Search filter
5.Feedback
6.Exit

Enter your choice:
2
(9, 'paneer', 'v', 'spicy')
(10, 'gulab jamun', 'v', 'sweet')
>>> |

```

```

.....Welcome.....

What do you identify yourself as?....

1.Customer
2.Vendor
3.Delivery boy
4.Exit

Enter your choice:
2
Username:pallu
Enter id:5
('pallu',)
What do you want to do?....

1.Add in menu
2.Delete from menu
3.Update in menu
4.Edit details
5.Exit

Enter your choice:
1
Enter food item name:16
Enter category- veg or non-veg(v/n):v
Enter remarks(spicy/extra cheesy...):cheesy
>>> |

```



```

.....Welcome.....

What do you identify yourself as?....

1.Customer
2.Vendor
3.Delivery boy
4.Exit

Enter your choice:
1
Username:safal
Enter id:3
('safal',)
What do you want to do?....

1.Edit details
2.View menu
3.Access cart
4.Search filter
5.Feedback
6.Exit

Enter your choice:
4
You can do filtered search based on veg/non-veg(v/n):....
v
(9, 'paneer', 'v', 'spicy')
(10, 'gulab jamun', 'v', 'sweet')
(51, 'l6', 'v', 'cheesy')
,

.....Welcome.....
.....

What do you identify yourself as?....

1.Customer
2.Vendor
3.Delivery boy
4.Exit

Enter your choice:
1
Username:ramlal
Enter id:4
What do you want to do?....

1.Edit details
2.View menu
3.Access cart
4.Search filter
5.Feedback
6.Exit

Enter your choice:
2
(9, 'paneer', 'v', 'spicy')
(10, 'gulab jamun', 'v', 'sweet')
(51, 'l6', 'v', 'cheesy')
,

```