Mini project report on

DELIVERY MANAGEMENT SYSTEM

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering

UE20CS301 – DBMS Project

Submitted by:

Surabhi Gudla	**PES2UG20CS919**

Under the guidance of

Prof. Nivedita Kasturi

Assistant Professor

Designation

PES University

AUG - DEC 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

DELIVERY MANAGEMENT SYSTEM

CERTIFICATE

This is to certify that the mini project entitled

is a bonafide work carried out by

Surabhi Gudla	**PES2UG20CS919**

In partial fulfilment for the completion of fifth semester DBMS Project (UE20CSS301) in the Program of Study -Bachelor of Technology in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period AUG. 2022 – DEC. 2022. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The project has been approved as it satisfies the 5th semester academic requirements in respect of project work.

Signature Prof. Nivedita Kasturi Assistant Professor	

DECLARATION

We hereby declare that the DBMS Project entitled **Online Travel Portal System** has been carried out by us under the guidance of **Prof. Nivedita Kasturi, Assistant Professor** and submitted in partial fulfilment of the course requirements for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester AUG – DEC 2022.

Surabhi Gudla	**PES2UG20CS919**	**Surabhi.G**

ACKNOWLEDGEMENT

I would like to express my gratitude to Prof. Nivedita Kasturi, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE20CS301 - DBMS Project.

I take this opportunity to thank Dr. Sandesh B J, C, Professor,ChairPerson, Department of Computer Science and Engineering, PES University, for all the knowledge and support I have received from the department.

I am deeply grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, PES University for providing to me various opportunities and enlightenment every step of the way. Finally, this DBMS Project could not have been completed without the continual support and encouragement I have received from my family and friends.

ABSTRACT

Delivery management system provides a platform from where a customer can buy their products and get it delivered to any city.

After COVID-19, every small task we do has become online. This has also become a boon for us, and we can order from anywhere and get it delivered not only to our city, but another city too.

TABLE OF CONTENTS

1. INTRODUCTION

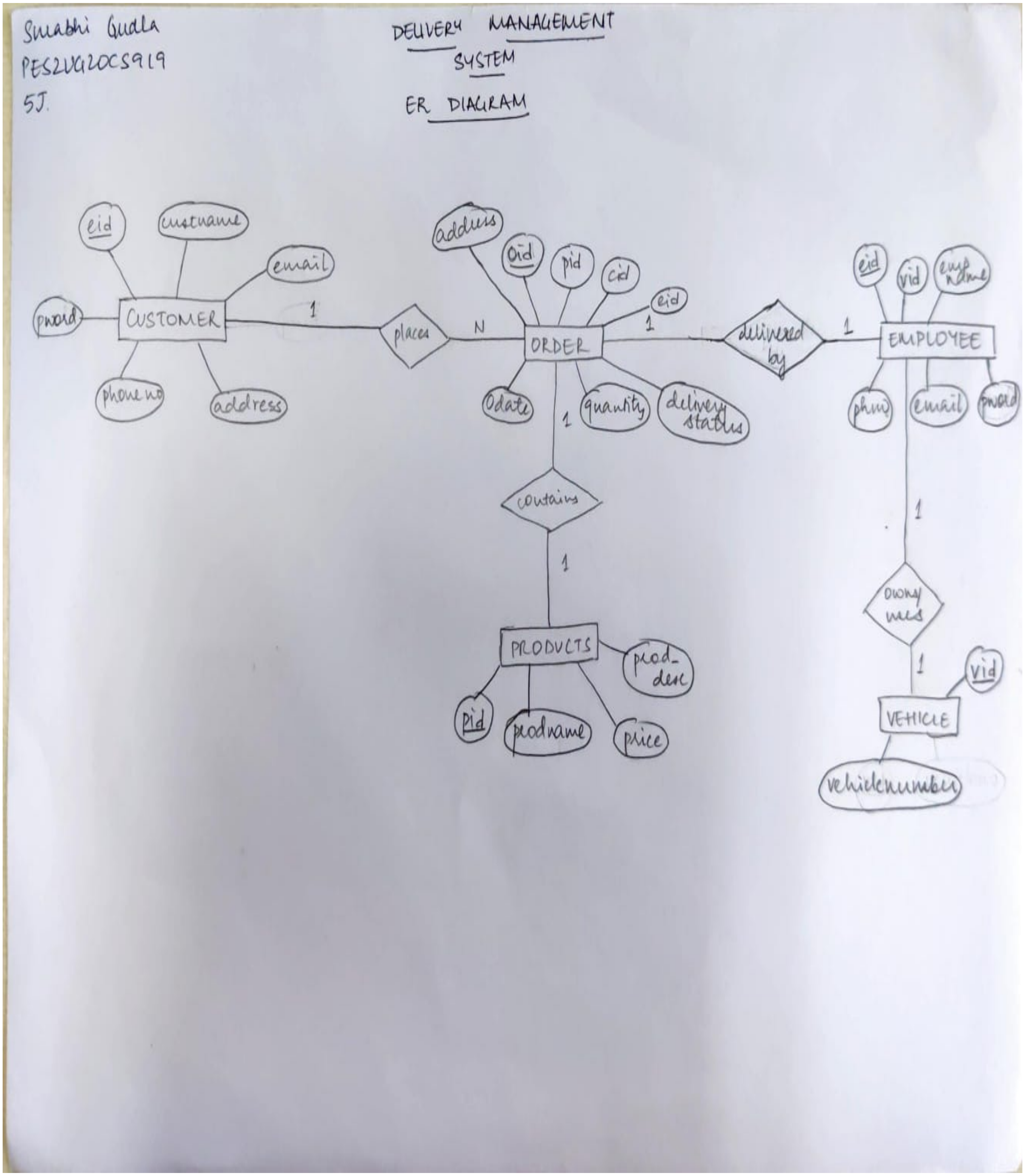
A delivery management system is a tech tool that automates, streamlines and digitalizes the end-to-end logistic processes. Planning, managing, optimizing and executing the final-mile logistics operations aren't possible without delivery management software.

This helps in simplifying the communication between the sender, the company delivering the products and the receiver.

2. PROBLEM DEFINITION

Delivery management system helps to automate and digitize our process of keeping track of the orders being made. Customers living in different cities can order products to their city or to any other city. There are different categories like basic necessities, dairy, snacks, bath and body products, and drinks from where the customer can order what they want. An employee owns a vehicle and is assigned a particular order to complete. The information about delivery status is maintained. If the order has been delivered, it will be initialised to 1 and if not, it is initialized to 0.

3. ER MODEL



Surabhi Gudla
PES2UG20CS919
5J.

DELIVERY MANAGEMENT
SYSTEM

ER DIAGRAM

4. ER TO RELATIONAL MAPPING

4.1 STEPS OF ALGORITHM FOR CHOSEN PROBLEM

Step 1: Mapping of Regular Entity Types

CUSTOMER

cid	custname	email	pword	phno	address

EMPLOYEE

eid	vid	empname	phno	email	pword

PRODUCTS

pid	prodname	prod_desc	price

ORDERS

oid	pid	cid	eid	odate	price	quantity	delivery status	vehicleno	address

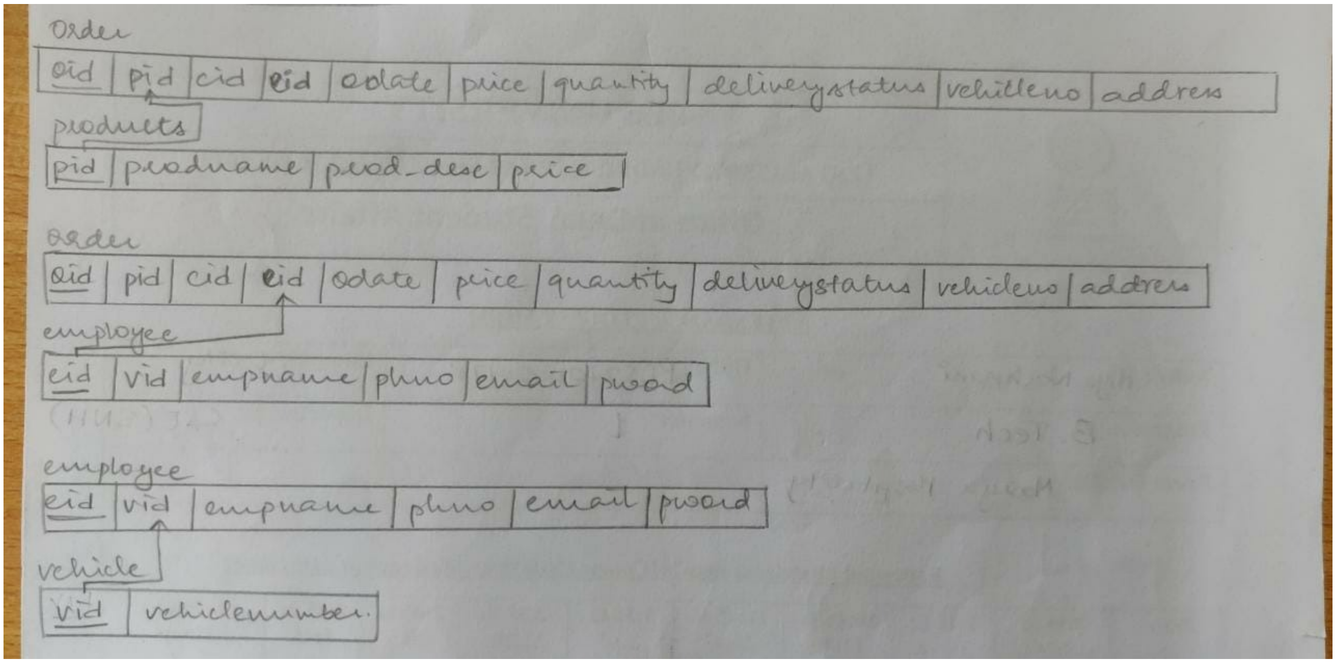
VEHICLE

vid	vehiclenumber

Step 2: Mapping of Weak Entity Types

There are no weak entity types

Step 3: Mapping of Binary 1:1 Relation Types



Order

oid	pid	cid	eid	odate	price	quantity	deliverystatus	vehicleno	address

products

pid	prodname	prod_desc	price

Order

oid	pid	cid	eid	odate	price	quantity	deliverystatus	vehicleno	address

employee

eid	vid	empname	phno	email	pword

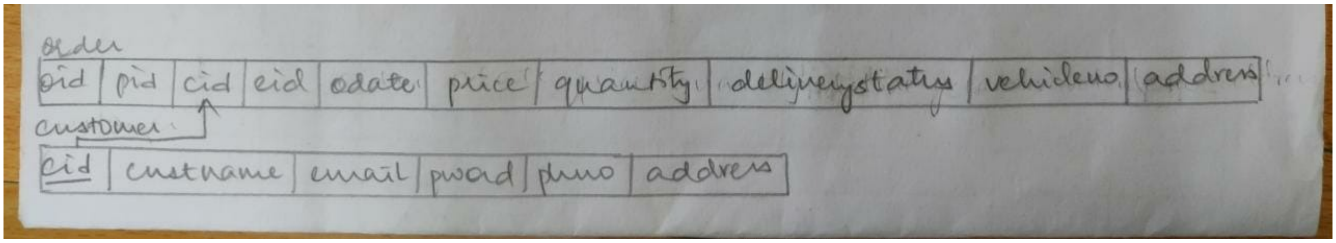
employee

eid	vid	empname	phno	email	pword

vehicle

vid	vehiclenumber.

Step 4: Mapping of Binary 1:N Relationship Types.



Order

oid	pid	cid	eid	odate	price	quantity	deliverystatus	vehicleno	address

customer

cid	custname	email	pword	phno	address

Step 5: Mapping of Binary M:N Relationship Types.

There are no m:n relationships

Step 6: Mapping of Multivalued attributes.

There are no multivalued attributes.

Step 7: Mapping of N-ary Relationship Types.

No n-ary relationships

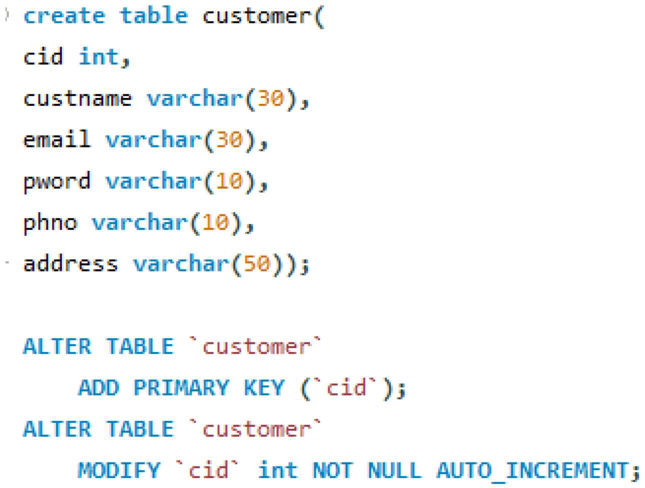
5. DDL STATEMENTS

STATEMENTS WITH SCREEN SHOTS OF THE TABLE CREATION

1. Table customer

```
#1) CUSTOMER TABLE
create table customer(
cid int,
custname varchar(30),
email varchar(30),
pword varchar(10),
phno varchar(10),
address varchar(50));

ALTER TABLE `customer`
        ADD PRIMARY KEY (`cid`);
ALTER TABLE `customer`
```

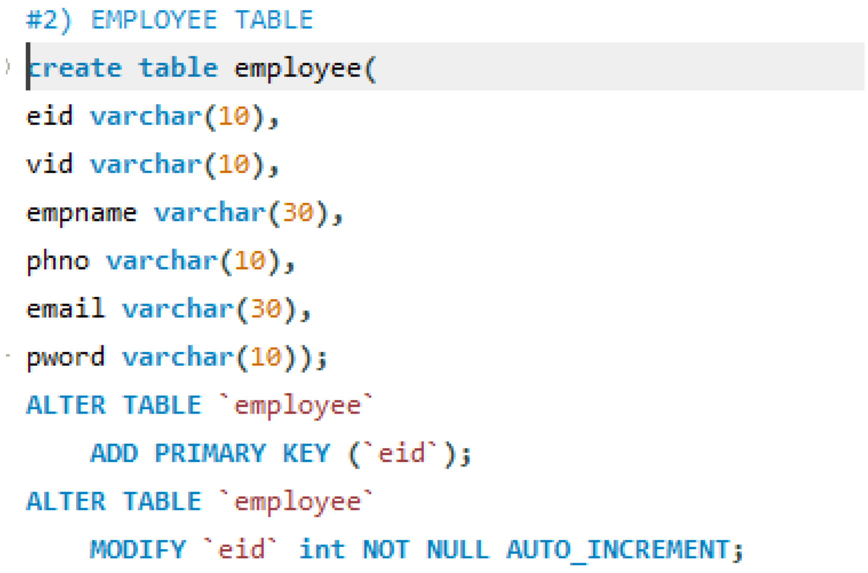


```
create table customer(
cid int,
custname varchar(30),
email varchar(30),
pword varchar(10),
phno varchar(10),
address varchar(50));

ALTER TABLE `customer`
    ADD PRIMARY KEY (`cid`);
ALTER TABLE `customer`
    MODIFY `cid` int NOT NULL AUTO_INCREMENT;
```

2. Table employee

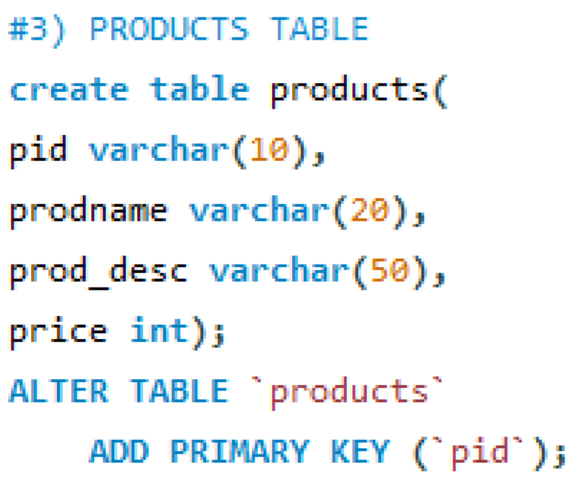
```
create table employee(
eid varchar(10),
vid varchar(10),
empname varchar(30),
phno varchar(10),
email varchar(30),
pword varchar(10));
ALTER TABLE `employee`
        ADD PRIMARY KEY (`eid`);
ALTER TABLE `employee`
        MODIFY `eid` int NOT NULL AUTO_INCREMENT;
```



```
#2) EMPLOYEE TABLE
create table employee(
eid varchar(10),
vid varchar(10),
empname varchar(30),
phno varchar(10),
email varchar(30),
pword varchar(10));
ALTER TABLE `employee`
    ADD PRIMARY KEY (`eid`);
ALTER TABLE `employee`
    MODIFY `eid` int NOT NULL AUTO_INCREMENT;
```

3. Table products

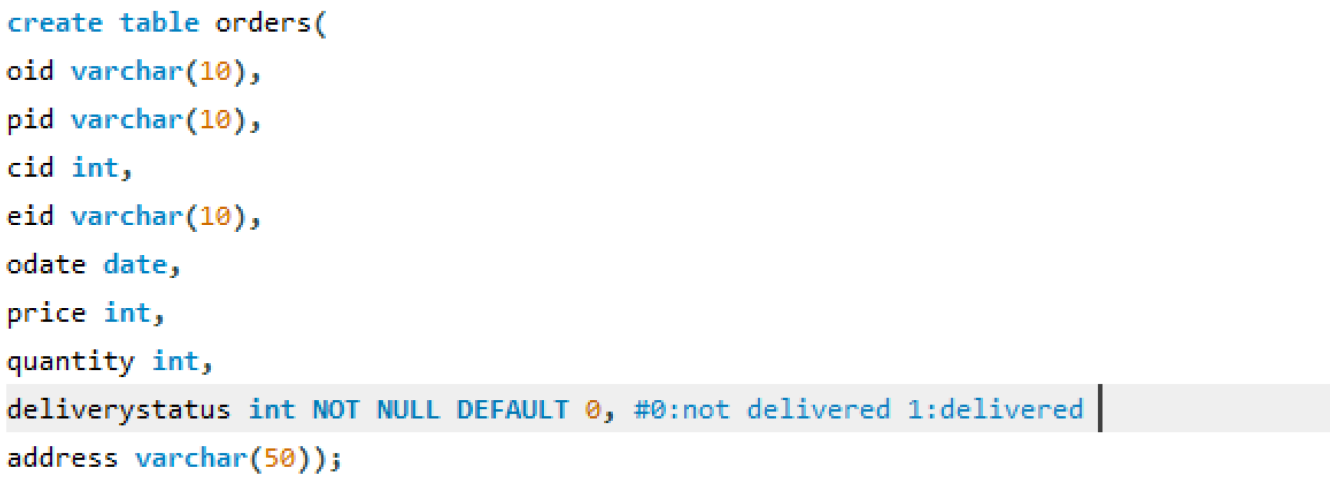
```
create table products(
pid varchar(10),
prodname varchar(20),
prod_desc varchar(50),
price int);
ALTER TABLE `products`
        ADD PRIMARY KEY (`pid`);
```



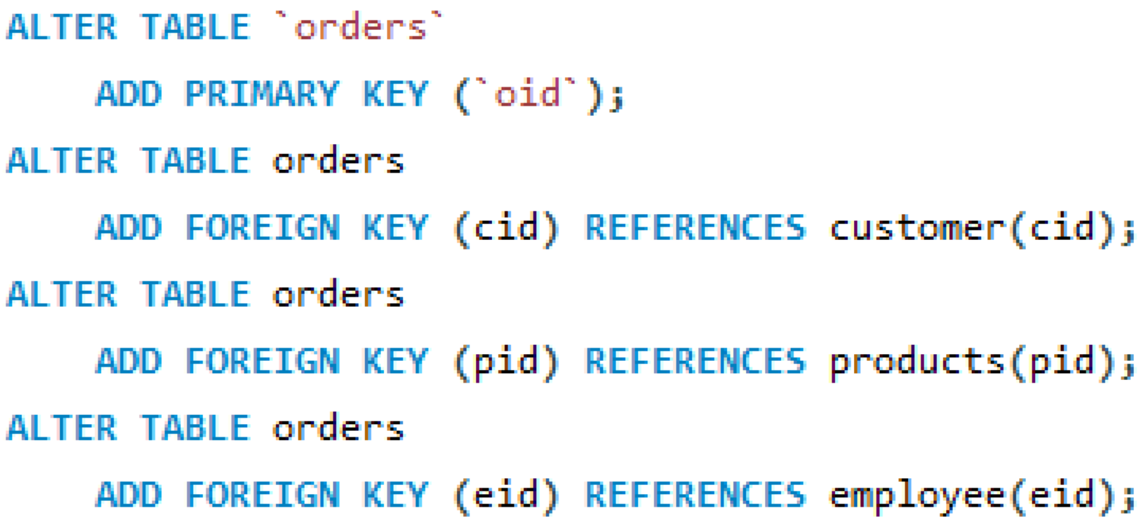
```
#3) PRODUCTS TABLE
create table products(
pid varchar(10),
prodname varchar(20),
prod_desc varchar(50),
price int);
ALTER TABLE `products`
    ADD PRIMARY KEY (`pid`);
```

4. Table orders

```
#4) ORDERS TABLE
create table orders(
oid varchar(10),
pid varchar(10),
cid int,
eid varchar(10),
odate date,
price int,
quantity int,
deliverystatus int NOT NULL DEFAULT 0, #0:not delivered 1:delivered
address varchar(50));
ALTER TABLE `orders`
        ADD PRIMARY KEY (`oid`);
ALTER TABLE orders
        ADD FOREIGN KEY (cid) REFERENCES customer(cid);
ALTER TABLE orders
        ADD FOREIGN KEY (pid) REFERENCES products(pid);
ALTER TABLE orders
        ADD FOREIGN KEY (eid) REFERENCES employee(eid);
```



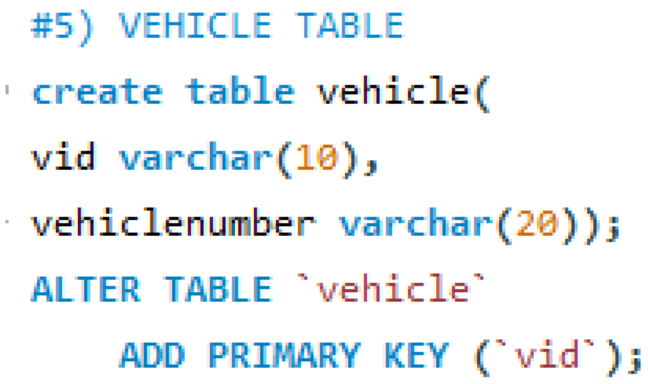
```
create table orders(
oid varchar(10),
pid varchar(10),
cid int,
eid varchar(10),
odate date,
price int,
quantity int,
deliverystatus int NOT NULL DEFAULT 0, #0:not delivered 1:delivered
address varchar(50));
```



```
ALTER TABLE `orders`
     ADD PRIMARY KEY (`oid`);
ALTER TABLE orders
     ADD FOREIGN KEY (cid) REFERENCES customer(cid);
ALTER TABLE orders
     ADD FOREIGN KEY (pid) REFERENCES products(pid);
ALTER TABLE orders
     ADD FOREIGN KEY (eid) REFERENCES employee(eid);
```

5. Table vehicle

```
create table vehicle(
vid varchar(10),
vehiclenumber varchar(20));
ALTER TABLE `vehicle`
     ADD PRIMARY KEY (`vid`);
```



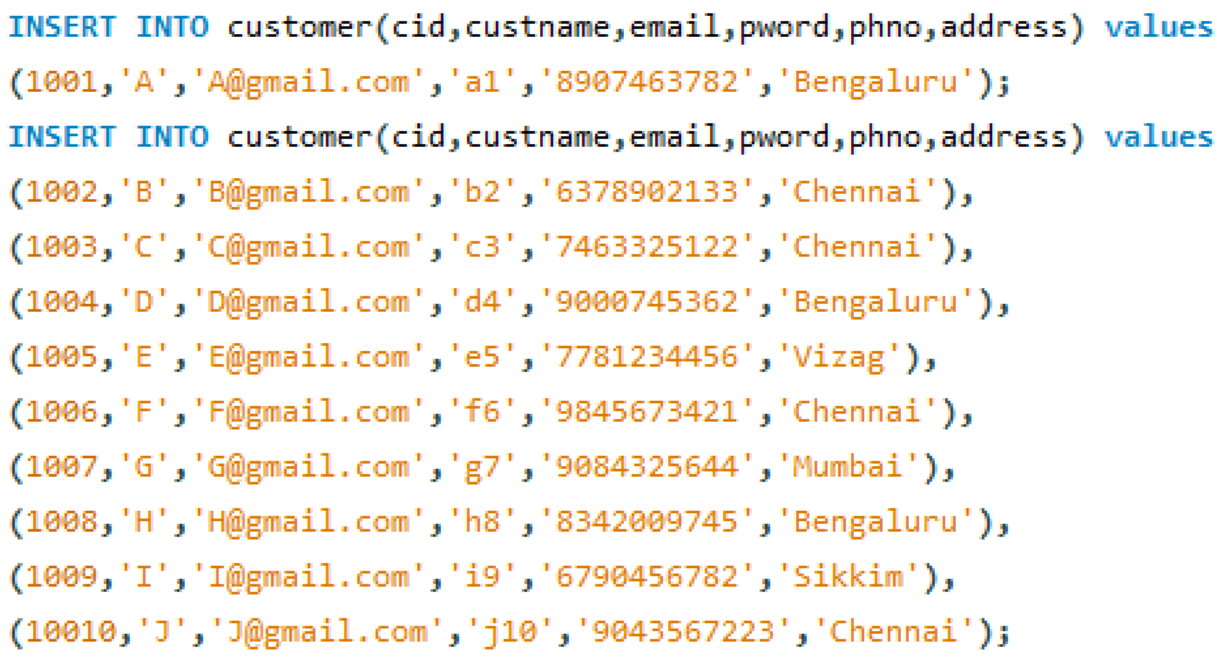
```
#5) VEHICLE TABLE
create table vehicle(
vid varchar(10),
vehiclenumber varchar(20));
ALTER TABLE `vehicle`
    ADD PRIMARY KEY (`vid`);
```

6. DML STATEMENTS

STATEMENTS WITH SCREEN SHOTS OF THE TABLE WITH INSERTED VALUES

1. Table customer

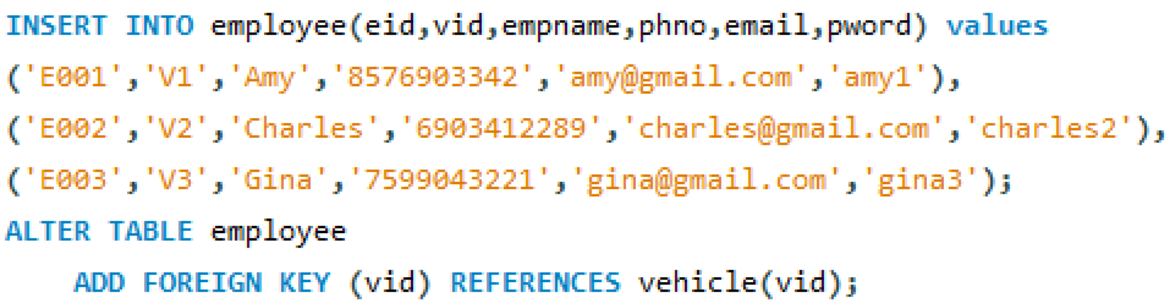
INSERT INTO customer(cid,custname,email,pword,phno,address) values
(1001,'A','A@gmail.com','a1','8907463782','Bengaluru');
INSERT INTO customer(cid,custname,email,pword,phno,address) values
(1002,'B','B@gmail.com','b2','6378902133','Chennai'),
(1003,'C','C@gmail.com','c3','7463325122','Chennai'),
(1004,'D','D@gmail.com','d4','9000745362','Bengaluru'),
(1005,'E','E@gmail.com','e5','7781234456','Vizag'),
(1006,'F','F@gmail.com','f6','9845673421','Chennai'),
(1007,'G','G@gmail.com','g7','9084325644','Mumbai'),
(1008,'H','H@gmail.com','h8','8342009745','Bengaluru'),
(1009,'I','I@gmail.com','i9','6790456782','Sikkim'),
(10010,'J','J@gmail.com','j10','9043567223','Chennai');



```
INSERT INTO customer(cid,custname,email,pword,phno,address) values
(1001,'A','A@gmail.com','a1','8907463782','Bengaluru');
INSERT INTO customer(cid,custname,email,pword,phno,address) values
(1002,'B','B@gmail.com','b2','6378902133','Chennai'),
(1003,'C','C@gmail.com','c3','7463325122','Chennai'),
(1004,'D','D@gmail.com','d4','9000745362','Bengaluru'),
(1005,'E','E@gmail.com','e5','7781234456','Vizag'),
(1006,'F','F@gmail.com','f6','9845673421','Chennai'),
(1007,'G','G@gmail.com','g7','9084325644','Mumbai'),
(1008,'H','H@gmail.com','h8','8342009745','Bengaluru'),
(1009,'I','I@gmail.com','i9','6790456782','Sikkim'),
(10010,'J','J@gmail.com','j10','9043567223','Chennai');
```

2. **Table employee**

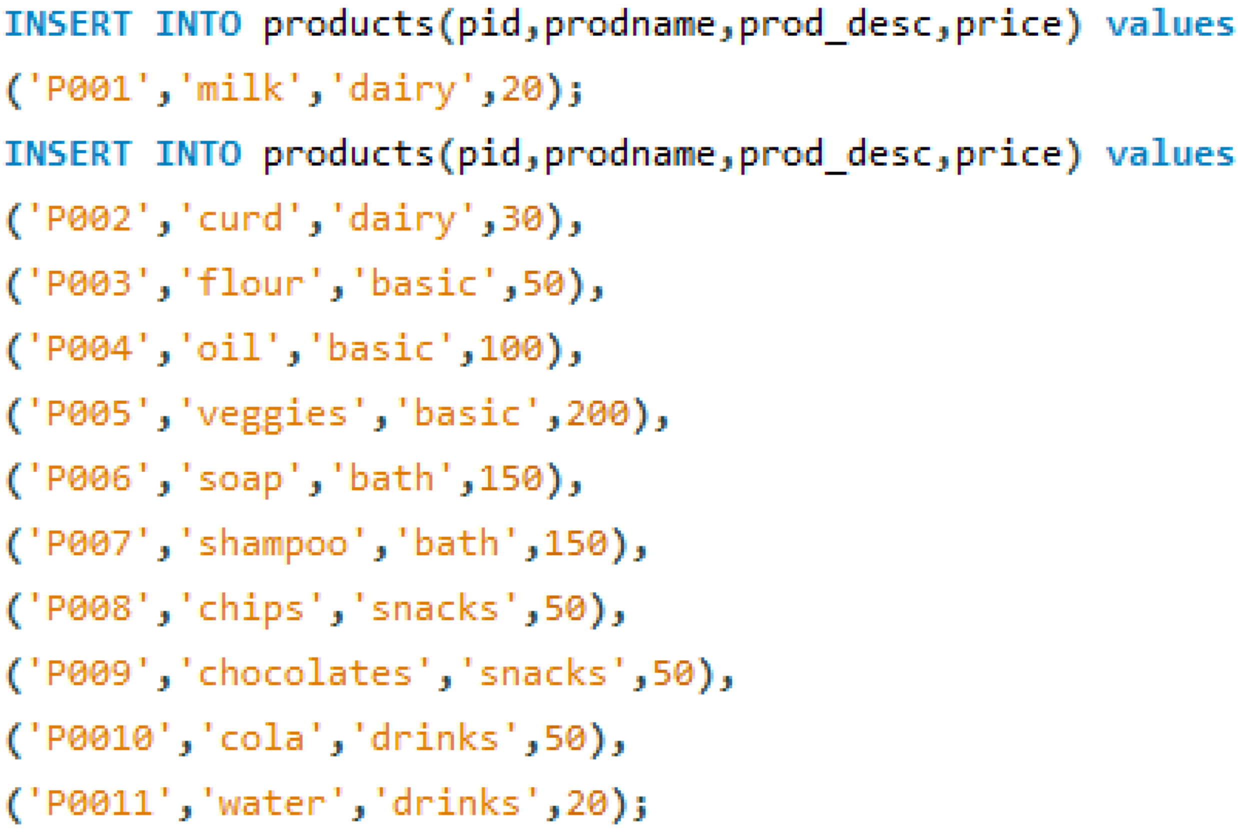
```
INSERT INTO employee(eid,vid,empname,phno,email,pword) values
('E001','V1','Amy','8576903342','amy@gmail.com','amy1'),
('E002','V2','Charles','6903412289','charles@gmail.com','charles2'),
('E003','V3','Gina','7599043221','gina@gmail.com','gina3');
ALTER TABLE employee
        ADD FOREIGN KEY (vid) REFERENCES vehicle(vid);
```



```
INSERT INTO employee(eid,vid,empname,phno,email,pword) values
('E001','V1','Amy','8576903342','amy@gmail.com','amy1'),
('E002','V2','Charles','6903412289','charles@gmail.com','charles2'),
('E003','V3','Gina','7599043221','gina@gmail.com','gina3');
ALTER TABLE employee
    ADD FOREIGN KEY (vid) REFERENCES vehicle(vid);
```

3. **Table products**

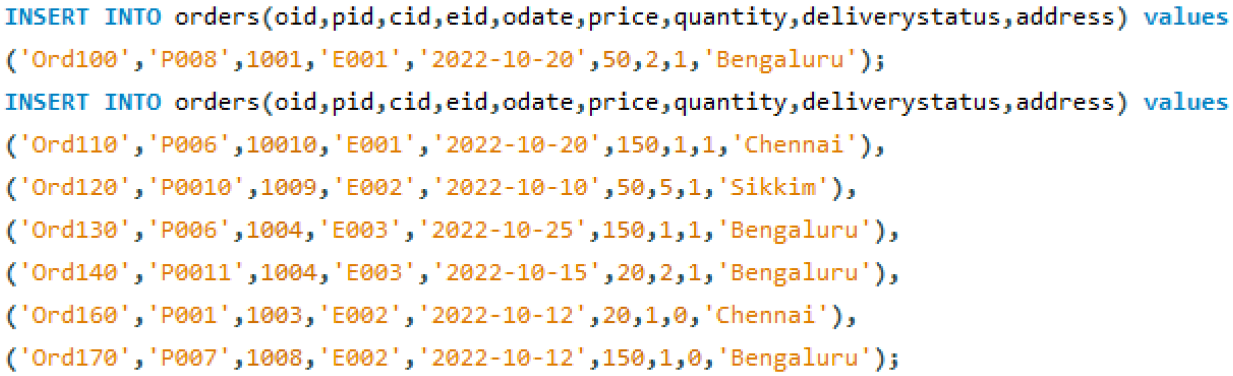
```
INSERT INTO products(pid,prodname,prod_desc,price) values
('P001','milk','dairy',20);
INSERT INTO products(pid,prodname,prod_desc,price) values
('P002','curd','dairy',30),
('P003','flour','basic',50),
('P004','oil','basic',100),
('P005','veggies','basic',200),
('P006','soap','bath',150),
('P007','shampoo','bath',150),
('P008','chips','snacks',50),
('P009','chocolates','snacks',50),
('P0010','cola','drinks',50),
('P0011','water','drinks',20);
```



```
INSERT INTO products(pid,prodname,prod_desc,price) values
('P001','milk','dairy',20);
INSERT INTO products(pid,prodname,prod_desc,price) values
('P002','curd','dairy',30),
('P003','flour','basic',50),
('P004','oil','basic',100),
('P005','veggies','basic',200),
('P006','soap','bath',150),
('P007','shampoo','bath',150),
('P008','chips','snacks',50),
('P009','chocolates','snacks',50),
('P0010','cola','drinks',50),
('P0011','water','drinks',20);
```

4. Table orders

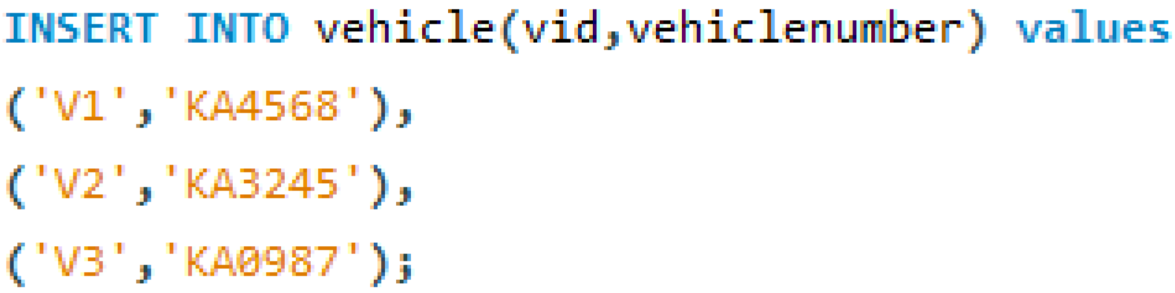
```
INSERT INTO orders(oid,pid,cid,eid,odate,price,quantity,deliverystatus,address) values
('Ord100','P008',1001,'E001','2022-10-20',50,2,1,'Bengaluru'),
('Ord110','P006',10010,'E001','2022-10-20',150,1,1,'Chennai'),
('Ord120','P0010',1009,'E002','2022-10-10',50,5,1,'Sikkim'),
('Ord130','P006',1004,'E003','2022-10-25',150,1,1,'Bengaluru'),
('Ord140','P0011',1004,'E003','2022-10-15',20,2,1,'Bengaluru'),
('Ord160','P001',1003,'E002','2022-10-12',20,1,0,'Chennai'),
('Ord170','P007',1008,'E002','2022-10-12',150,1,0,'Bengaluru');
```



```
INSERT INTO orders(oid,pid,cid,eid,odate,price,quantity,deliverystatus,address) values
('Ord100','P008',1001,'E001','2022-10-20',50,2,1,'Bengaluru');
INSERT INTO orders(oid,pid,cid,eid,odate,price,quantity,deliverystatus,address) values
('Ord110','P006',10010,'E001','2022-10-20',150,1,1,'Chennai'),
('Ord120','P0010',1009,'E002','2022-10-10',50,5,1,'Sikkim'),
('Ord130','P006',1004,'E003','2022-10-25',150,1,1,'Bengaluru'),
('Ord140','P0011',1004,'E003','2022-10-15',20,2,1,'Bengaluru'),
('Ord160','P001',1003,'E002','2022-10-12',20,1,0,'Chennai'),
('Ord170','P007',1008,'E002','2022-10-12',150,1,0,'Bengaluru');
```

5. Table vehicle

```
INSERT INTO vehicle(vid,vehiclenumber) values
('V1','KA4568'),
('V2','KA3245'),
('V3','KA0987');
```



```
INSERT INTO vehicle(vid,vehiclenumber) values
('V1','KA4568'),
('V2','KA3245'),
('V3','KA0987');
```

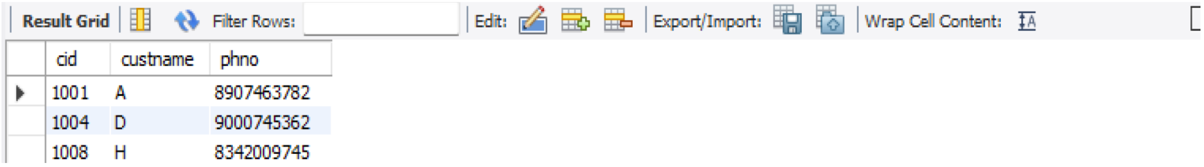
7. QUERIES

a. One Nested Query and one Correlated query

Retrieve the customer id, customer name and the phone number from customer table where the address is ‘Bengaluru’

Code:

```
select cid,custname,phno
from customer
where address='Bengaluru';
```

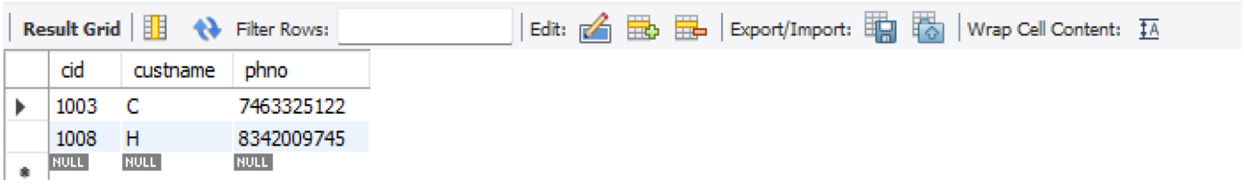


cid	custname	phno
1001	A	8907463782
1004	D	9000745362
1008	H	8342009745

Retrieve the customer id, customer name and the phone number whose delivery status is 0, that is not delivered

Code:

```
 select C.cid,C.custname,C.phno
from customer as C
where cid in (select cid from orders
                                   where deliverystatus = 0);
```



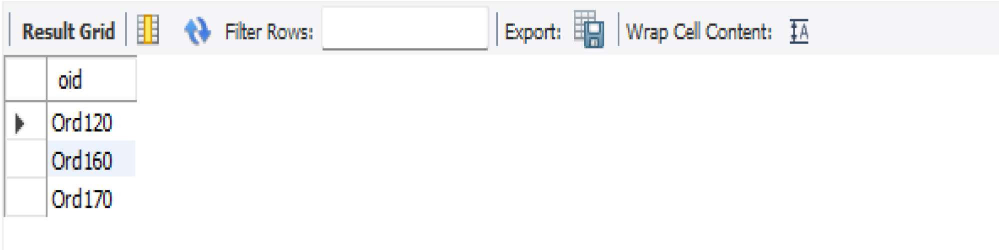
cid	custname	phno
1003	C	7463325122
1008	H	8342009745
NULL	NULL	NULL

b. Two queries for SET Operations

Retrieve the order id of orders placed on either of the dates 10/10/2022 or 12/10/2022

Code:

```
Select oid from orders where (odate='2022-10-10')
UNION
select oid from orders where (odate='2022-10-12');
```

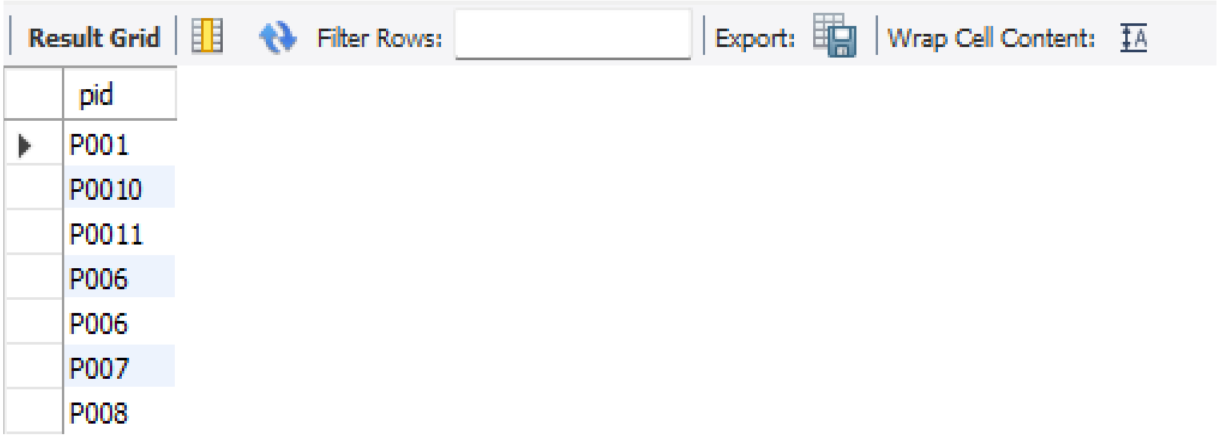


oid
Ord120
Ord160
Ord170

Retrieve the product id present in both tables orders and products

Code:

```
select pid from orders
where EXISTS
(select pid from products);
```



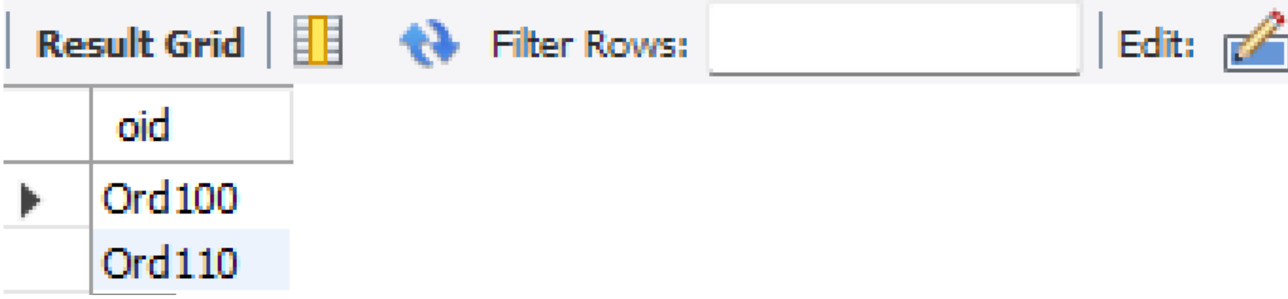
pid
P001
P0010
P0011
P006
P006
P007
P008

c. Query for Aggregate functions with group by clause, use HavingClause.

Count of the number of orders employee id ‘E001’ has

Code:

```
select oid
from orders
where eid = 'E001';
```



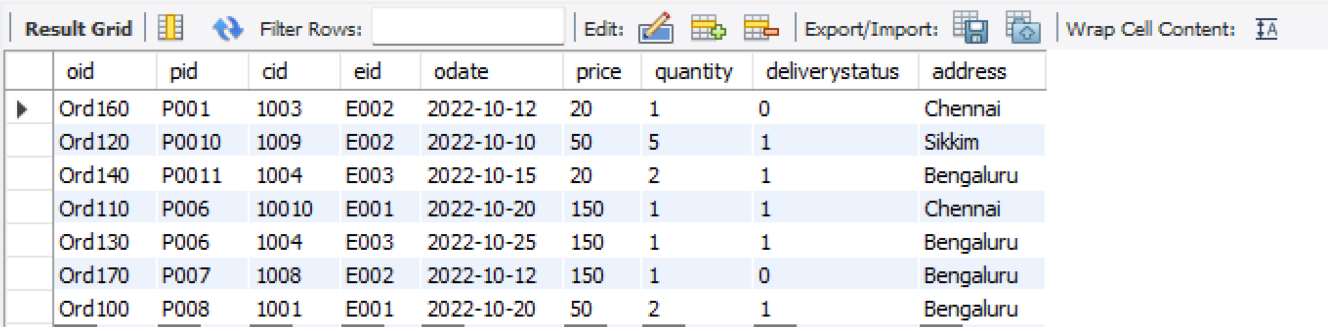
oid
Ord100
Ord110

d. Two Simple Queries with order by clause.

Sort the orders table in ascending order by product id

Code:

```
SELECT * FROM orders
ORDER BY pid ASC;
```

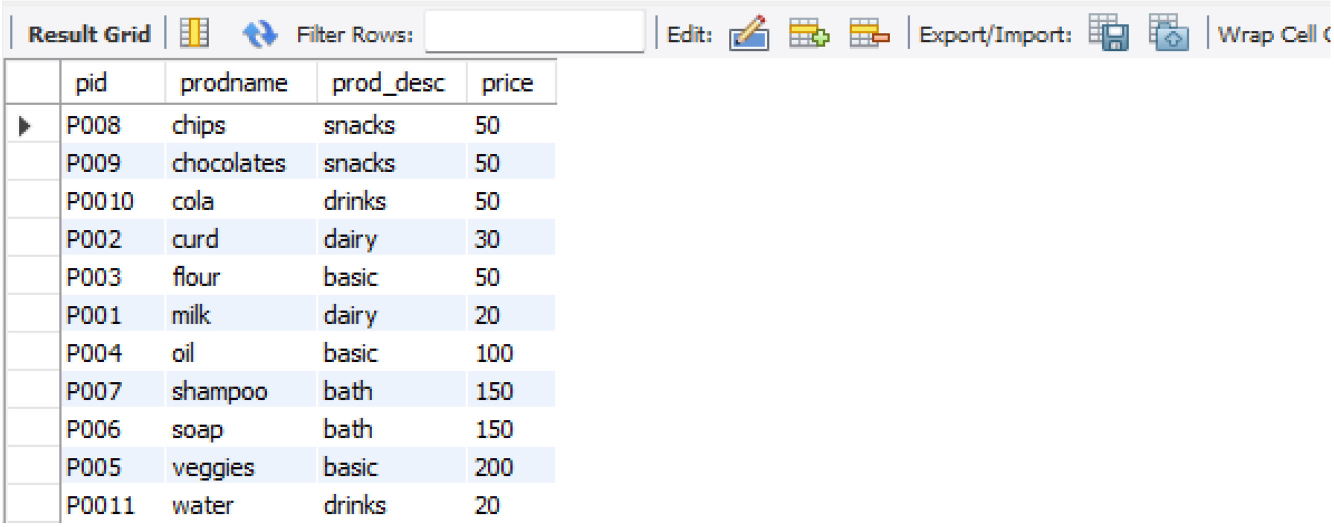


oid	pid	cid	eid	odate	price	quantity	deliverystatus	address
Ord160	P001	1003	E002	2022-10-12	20	1	0	Chennai
Ord120	P0010	1009	E002	2022-10-10	50	5	1	Sikkim
Ord140	P0011	1004	E003	2022-10-15	20	2	1	Bengaluru
Ord110	P006	10010	E001	2022-10-20	150	1	1	Chennai
Ord130	P006	1004	E003	2022-10-25	150	1	1	Bengaluru
Ord170	P007	1008	E002	2022-10-12	150	1	0	Bengaluru
Ord100	P008	1001	E001	2022-10-20	50	2	1	Bengaluru

Sort the products table by prodname

Code:

```
SELECT * FROM products
ORDER BY prodname;
```



pid	prodname	prod_desc	price
P008	chips	snacks	50
P009	chocolates	snacks	50
P0010	cola	drinks	50
P002	curd	dairy	30
P003	flour	basic	50
P001	milk	dairy	20
P004	oil	basic	100
P007	shampoo	bath	150
P006	soap	bath	150
P005	veggies	basic	200
P0011	water	drinks	20

8. STORED PROCEDURES, FUNCTIONS AND TRIGGERS

8.1 STORED PROCEDURES OR FUNCTIONS

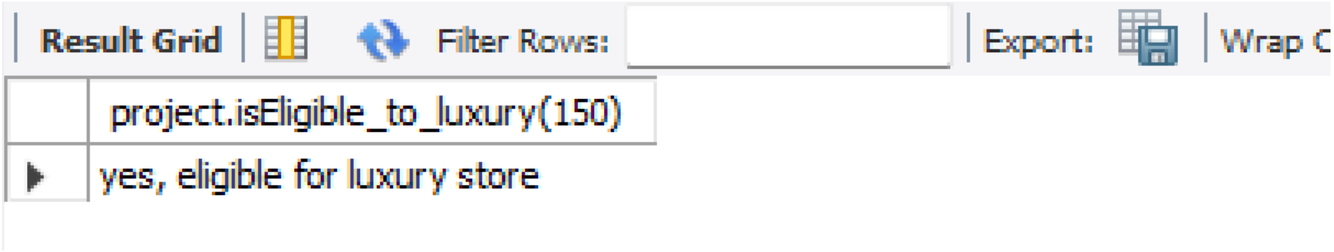
Stored procedure:

To check if a product belongs to the luxury class or no.

Any product whose price is more than Rs.100 is considered to be a luxury product.

```
    CREATE DEFINER=`root`@`localhost` FUNCTION
`isEligible_to_luxury`(price INTEGER
    ) RETURNS varchar(200) CHARSET utf8mb4
      DETERMINISTIC
    BEGIN
    IF price>100 THEN
    RETURN ("yes, eligible for luxury store");
    ELSE
    RETURN ("No, not eligible for luxury store");
    END IF;
    END
```

```
select project.isEligible_to_luxury(150);
```



	project.isEligible_to_luxury(150)
▶	yes, eligible for luxury store

8.2 TRIGGERS

Create a trigger to update the order table if the deliverystatus is null.

```
delimiter //
CREATE TRIGGER status_update BEFORE INSERT ON orders
FOR EACH ROW
BEGIN
    IF NEW.deliverystatus = NULL THEN
        SET NEW.deliverystatus = 0;
END
delimiter ;
```

9. FRONT END DEVELOPEMNT

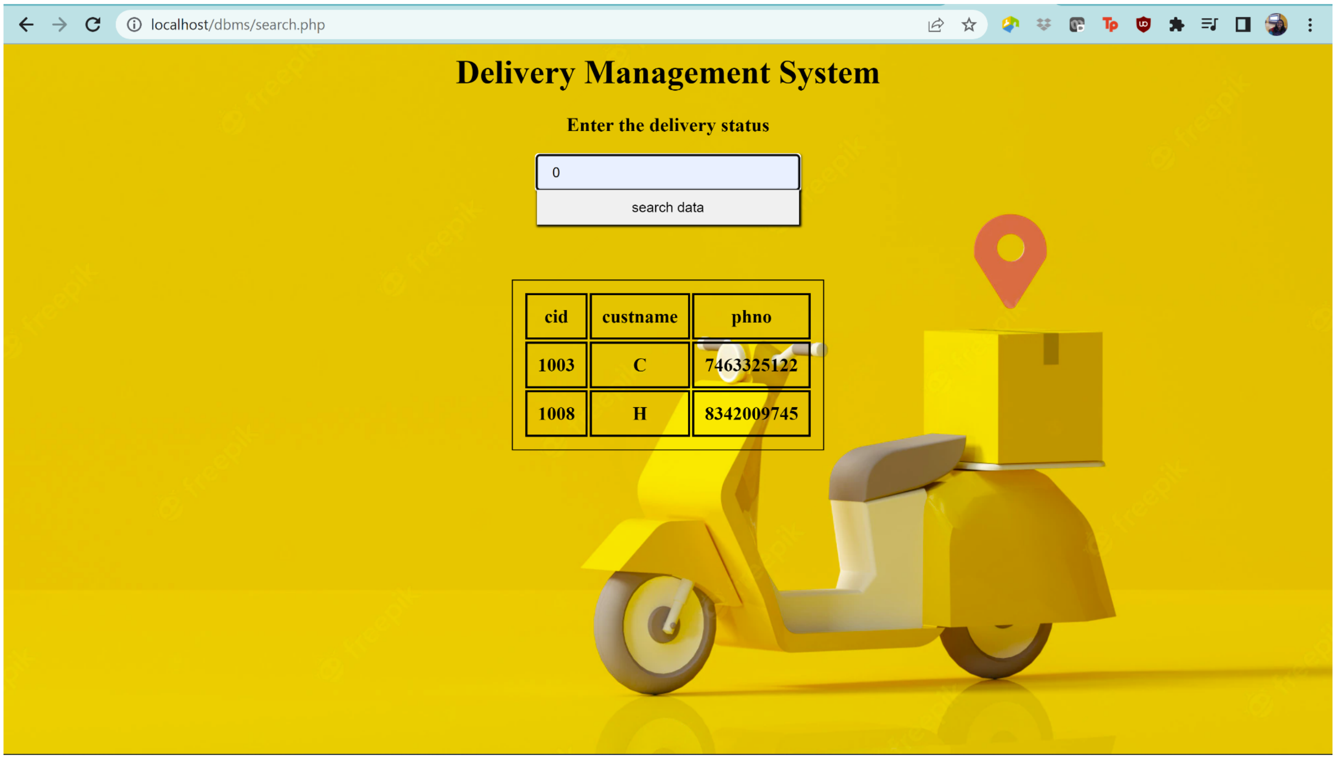
Developing frontend for the query:(for correlated query)

Retrieve the customer id, customer name and the phone number whose delivery status is 0, that is not delivered

Code:

```
 select C.cid,C.custname,C.phno
from customer as C
where cid in (select cid from orders
                                where deliverystatus = 0);
```

For delivery status: 0 (not delivered)



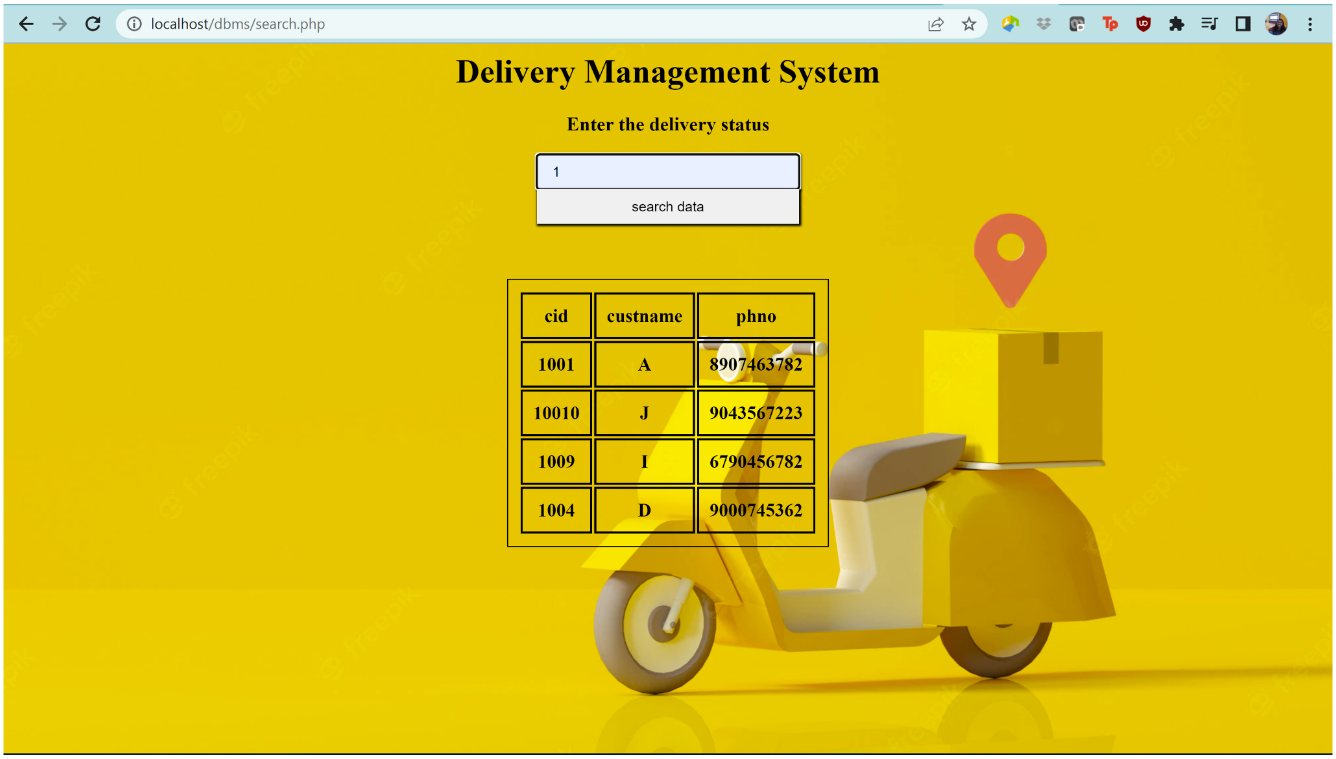
Delivery Management System

Enter the delivery status

0

cid	custname	phno
1003	C	7463325122
1008	H	8342009745

For delivery status: 1 (delivered)



Delivery Management System

Enter the delivery status

1

cid	custname	phno
1001	A	8907463782
10010	J	9043567223
1009	I	6790456782
1004	D	9000745362