DBMS - MINI PROJECT

INVENTORY MANAGEMENT SYSTEM

Submitted By:

Name: Jayasimha S

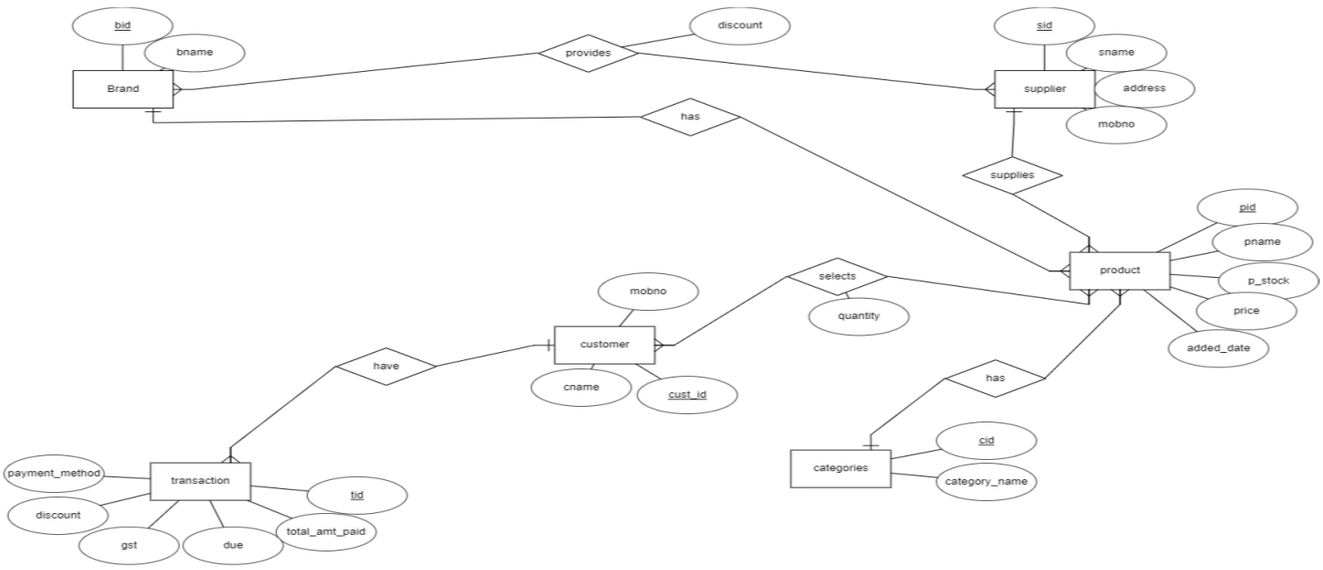
SRN: PES1UG20CS177

V Semester Section = C Section

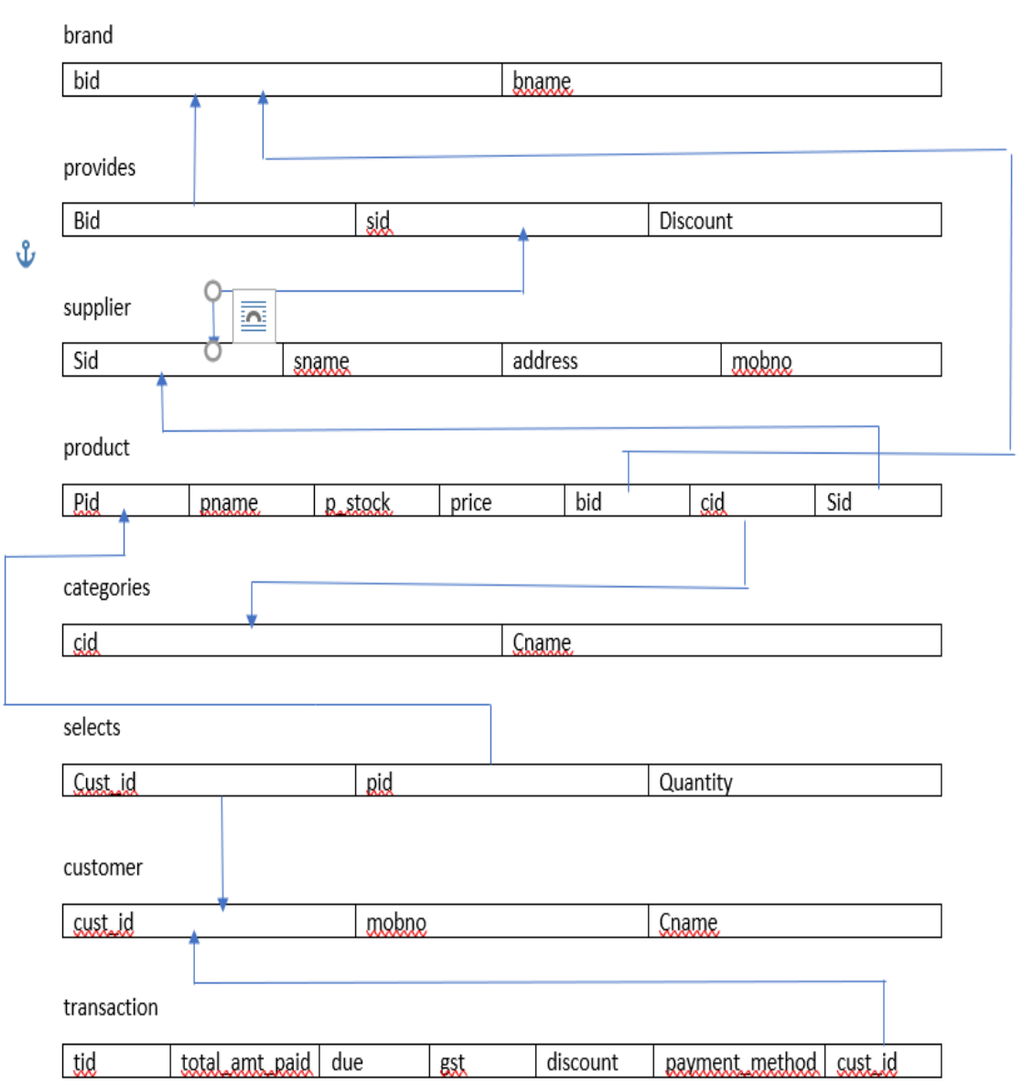
ABSTRACT

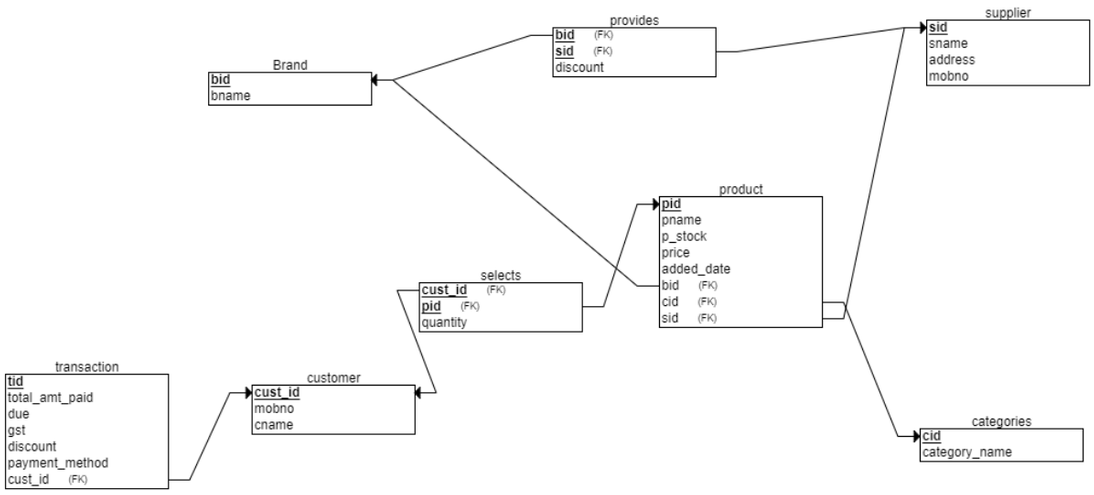
The inventory management system consists of 5 basic entities namely brand, categories, product, supplier, customer, transaction. The product table consists of the product details of all the products of different brands, different categories and from different suppliers whose details are present in the brand, categories and supplier table respectively. The customer table consists the details of all the customers and when the customer selects a product, his selection is recorded in the selects table and the transaction details of his selected is item are recorded by the inventory manager in the transaction table. The front end for this system is developed using python and backend using mysql database.

ER Diagram



Relational Schema





DDL statements - Building the database

```
CREATE TABLE brand
(
  bid INT NOT NULL,
  bname VARCHAR(20) NOT NULL,
  PRIMARY KEY (bid)
);

CREATE TABLE categories
(
  cid INT NOT NULL,
  category_name VARCHAR(20) NOT NULL,
  PRIMARY KEY (cid)
);

CREATE TABLE supplier
(
  sid INT NOT NULL,
  sname VARCHAR(20) NOT NULL,
  address VARCHAR(20) NOT NULL,
  mobno INT NOT NULL,
  PRIMARY KEY (sid)
);
```

```
CREATE TABLE customer
(
  mobno INT NOT NULL,
  cust_id INT NOT NULL,
  cname VARCHAR(20) NOT NULL,
  PRIMARY KEY (cust_id)
);

CREATE TABLE transaction
(
  tid INT NOT NULL,
  total_amt_paid INT NOT NULL,
  due INT NOT NULL,
  gst INT NOT NULL,
  discount INT NOT NULL,
  payment_method VARCHAR(20) NOT NULL,
  cust_id INT NOT NULL,
  PRIMARY KEY (tid),
  FOREIGN KEY (cust_id) REFERENCES customer(cust_id)
);

CREATE TABLE provides
(
  discount INT NOT NULL,
  bid INT NOT NULL,
  sid INT NOT NULL,
  PRIMARY KEY (bid, sid),
  FOREIGN KEY (bid) REFERENCES Brand(bid),
  FOREIGN KEY (sid) REFERENCES supplier(sid)
);

CREATE TABLE product
(
```

```
  pid INT NOT NULL,
  pname VARCHAR(20) NOT NULL,
  p_stock INT NOT NULL,
  price INT NOT NULL,
  bid INT NOT NULL,
  cid INT NOT NULL,
  sid INT NOT NULL,
  PRIMARY KEY (pid),
  FOREIGN KEY (bid) REFERENCES brand(bid),
  FOREIGN KEY (cid) REFERENCES categories(cid),
  FOREIGN KEY (sid) REFERENCES supplier(sid)
);

CREATE TABLE selects
(
  quantity INT NOT NULL,
  cust_id INT NOT NULL,
  pid INT NOT NULL,
  PRIMARY KEY (cust_id, pid),
  FOREIGN KEY (cust_id) REFERENCES customer(cust_id),
  FOREIGN KEY (pid) REFERENCES product(pid)
);
```

Populating the Database

Brand table:

```
MariaDB [ims]> insert into brand values(1,'Apple');
Query OK, 1 row affected (0.016 sec)

MariaDB [ims]> insert into brand values(2,'Samsung');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into brand values(3,'Sony');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into brand values(4,'LG');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into brand values(5,'One-Plus');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into brand values(6,'Panasonic');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> select * from brand;
+-----+-----------+
| bid | bname     |
+-----+-----------+
|   1 | Apple     |
|   2 | Samsung   |
|   3 | Sony      |
|   4 | LG        |
|   5 | One-Plus  |
|   6 | Panasonic |
+-----+-----------+
6 rows in set (0.000 sec)
```

Categories table:

```
MariaDB [ims]> insert into categories values(1,'mobiles');
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into categories values(2,'television');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into categories values(3,'speakers');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into categories values(4,'laptops');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into categories values(5,'spare_accessories');
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [ims]> select * from categories;
+-----+-------------------+
| cid | category_name     |
+-----+-------------------+
|   1 | mobiles           |
|   2 | television        |
|   3 | speakers          |
|   4 | laptops           |
|   5 | spare_accessories |
+-----+-------------------+
5 rows in set (0.000 sec)
```

Product table:

```
MariaDB [ims]> insert into product values(100,'iphone 10',5,60000,1,1,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(101,'iphone 11',5,75000,1,1,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(102,'iphone 12',5,80000,1,1,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(103,'iphone 13',5,90000,1,1,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(104,'iphone 14',5,110000,1,1,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(105,'galaxy m11',6,12000,2,1,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(106,'galaxy flip',4,120000,2,1,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(107,'bravia',4,150000,3,2,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(108,'smart tv',4,13000,3,2,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(109,'ultimate woofer',7,24000,4,3,7845);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(110,'ultimate sub woofer',7,2000,4,3,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(111,'slide a1',8,70000,5,4,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(112,'flux z3',8,124000,5,4,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(113,'m1 bionic',8,170000,1,4,7845);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into product values(114,'m5 spionic',8,200000,1,4,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(115,'qled',8,100000,2,2,7845);
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [ims]> insert into product values(116,'oled',8,180000,2,2,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(117,'play station 1',5,3000,3,5,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(118,'play station 3',5,10000,3,5,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(119,'web cam 1',5,1000,6,5,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(120,'keyboard x',5,2000,6,5,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(121,'power bank',5,1500,6,5,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(122,'65 inch curved tv',5,150000,4,2,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(123,'75 inch curved tv',5,250000,4,2,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(124,'55 inch curved tv',5,50000,4,2,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(125,'10R',5,60000,5,1,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(126,'11R',5,66000,5,1,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(127,'nord',5,21000,5,1,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(128,'smart watch f1 pro',3,51000,2,5,8147);
Query OK, 1 row affected (0.002 sec)
```

```
MariaDB [ims]> insert into product values(129,'smart watch s5 ultra',3,75000,2,5,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(130,'smart watch ez',3,7000,2,5,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into product values(131,'apple tv',3,70000,1,2,9863);
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into product values(132,'apple tv uhd',4,89000,1,2,9863);
Query OK, 1 row affected (0.003 sec)

MariaDB [ims]> insert into product values(133,'iwatch series 7',4,90000,1,5,9863);
Query OK, 1 row affected (0.003 sec)

MariaDB [ims]> insert into product values(134,'iwatch series 8',4,100000,1,5,9863);
Query OK, 1 row affected (0.003 sec)

MariaDB [ims]> insert into product values(135,'play station 4',2,40000,3,5,9863);
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into product values(136,'play station 5',1,50000,3,5,9863);
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into product values(137,'headphones 1sl',1,5000,3,5,9863);
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into product values(138,'bluetooth soundbar',1,55000,3,3,9863);
Query OK, 1 row affected (0.003 sec)

MariaDB [ims]> insert into product values(139,'home theatre speakers',1,200000,3,3,9863);
Query OK, 1 row affected, 1 warning (0.003 sec)

MariaDB [ims]> delete from product where pid=139;
Query OK, 1 row affected (0.004 sec)

MariaDB [ims]> insert into product values(139,'hometheatrespeakers',1,200000,3,3,9863);
Query OK, 1 row affected (0.004 sec)
```

```
MariaDB [ims]> select * from product;
+-----+----------------------+---------+--------+-----+-----+------+
| pid | pname                | p_stock | price  | bid | cid | sid  |
+-----+----------------------+---------+--------+-----+-----+------+
| 100 | iphone 10            |       5 |  60000 |   1 |   1 | 4218 |
| 101 | iphone 11            |       5 |  75000 |   1 |   1 | 4218 |
| 102 | iphone 12            |       5 |  80000 |   1 |   1 | 4218 |
| 103 | iphone 13            |       5 |  90000 |   1 |   1 | 4218 |
| 104 | iphone 14            |       5 | 110000 |   1 |   1 | 4218 |
| 105 | galaxy m11           |       6 |  12000 |   2 |   1 | 4218 |
| 106 | galaxy flip          |       4 | 120000 |   2 |   1 | 4218 |
| 107 | bravia               |       4 | 150000 |   3 |   2 | 4218 |
| 108 | smart tv             |       4 |  13000 |   3 |   2 | 4218 |
| 109 | ultimate woofer      |       7 |  24000 |   4 |   3 | 7845 |
| 110 | ultimate sub woofer  |       7 |   2000 |   4 |   3 | 7845 |
| 111 | slide a1             |       8 |  70000 |   5 |   4 | 7845 |
| 112 | flux z3              |       8 | 124000 |   5 |   4 | 7845 |
| 113 | m1 bionic            |       8 | 170000 |   1 |   4 | 7845 |
| 114 | m5 spionic           |       8 | 200000 |   1 |   4 | 7845 |
| 115 | qled                 |       8 | 100000 |   2 |   2 | 7845 |
| 116 | oled                 |       8 | 180000 |   2 |   2 | 7845 |
| 117 | play station 1       |       5 |   3000 |   3 |   5 | 7845 |
| 118 | play station 3       |       5 |  10000 |   3 |   5 | 7845 |
| 119 | web cam 1            |       5 |   1000 |   6 |   5 | 7845 |
| 120 | keyboard x           |       5 |   2000 |   6 |   5 | 7845 |
| 121 | power bank           |       5 |   1500 |   6 |   5 | 7845 |
| 122 | 65 inch curved tv    |       5 | 150000 |   4 |   2 | 8147 |
| 123 | 75 inch curved tv    |       5 | 250000 |   4 |   2 | 8147 |
| 124 | 55 inch curved tv    |       5 |  50000 |   4 |   2 | 8147 |
| 125 | 10R                  |       5 |  60000 |   5 |   1 | 8147 |
| 126 | 11R                  |       5 |  66000 |   5 |   1 | 8147 |
| 127 | nord                 |       5 |  21000 |   5 |   1 | 8147 |
| 128 | smart watch f1 pro   |       3 |  51000 |   2 |   5 | 8147 |
| 129 | smart watch s5 ultra |       3 |  75000 |   2 |   5 | 8147 |
| 130 | smart watch ez       |       3 |   7000 |   2 |   5 | 8147 |
| 131 | apple tv             |       3 |  70000 |   1 |   2 | 9863 |
| 132 | apple tv uhd         |       4 |  89000 |   1 |   2 | 9863 |
| 133 | iwatch series 7      |       4 |  90000 |   1 |   5 | 9863 |
| 134 | iwatch series 8      |       4 | 100000 |   1 |   5 | 9863 |
| 135 | play station 4       |       2 |  40000 |   3 |   5 | 9863 |
| 136 | play station 5       |       1 |  50000 |   3 |   5 | 9863 |
| 137 | headphones 1sl       |       1 |   5000 |   3 |   5 | 9863 |
| 138 | bluetooth soundbar   |       1 |  55000 |   3 |   3 | 9863 |
| 139 | hometheatrespeakers  |       1 | 200000 |   3 |   3 | 9863 |
+-----+----------------------+---------+--------+-----+-----+------+
40 rows in set (0.002 sec)
```

Provides table:

```
MariaDB [ims]> insert into provides values(5,1,4218);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into provides values(5,6,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(3.5,3,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(15,2,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(7,2,4218);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(10,4,7845);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(6,4,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(8,5,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(20,6,8147);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(2,4,9863);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(4,2,9863);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(10,3,9863);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into provides values(3,1,7845);
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [ims]> select * from provides;
+----------+-----+------+
| discount | bid | sid  |
+----------+-----+------+
|        5 |   1 | 4218 |
|        3 |   1 | 7845 |
|        7 |   2 | 4218 |
|       15 |   2 | 7845 |
|        4 |   2 | 9863 |
|        4 |   3 | 4218 |
|       10 |   3 | 9863 |
|       10 |   4 | 7845 |
|        6 |   4 | 8147 |
|        2 |   4 | 9863 |
|        8 |   5 | 8147 |
|        5 |   6 | 4218 |
|       20 |   6 | 8147 |
+----------+-----+------+
13 rows in set (0.000 sec)
```

Supplier table:

```
MariaDB [ims]> insert into supplier values(4218,'jacob','indiranagar',885241862);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into supplier values(7845,'suraj','malleshwaram',978451485);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into supplier values(9863,'kumar','vijaynagar',685788874);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into supplier values(8147,'simha','sadashivnagar',777581236);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> select * from supplier;
+------+-------+---------------+-----------+
| sid  | sname | address       | mobno     |
+------+-------+---------------+-----------+
| 4218 | jacob | indiranagar   | 885241862 |
| 7845 | suraj | malleshwaram  | 978451485 |
| 8147 | simha | sadashivnagar | 777581236 |
| 9863 | kumar | vijaynagar    | 685788874 |
+------+-------+---------------+-----------+
4 rows in set (0.000 sec)
```

Customer table:

```
MariaDB [ims]> insert into customer values(987542548,500,'ronaldo');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(887552548,501,'benzema');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(785412684,502,'maradona');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(785875963,503,'nadal');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(658775963,504,'williams');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(954455963,505,'tendulkar');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(987563458,506,'akthar');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(857463458,507,'tyson');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into customer values(45785458,508,'woods');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(578245458,509,'ozil');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into customer values(558999857,510,'jordan');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(896499857,511,'brady');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(891247457,512,'virat');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(474587457,513,'neeraj');
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into customer values(998574457,514,'pawan');
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into customer values(774857777,515,'jayasimha');
Query OK, 1 row affected (0.002 sec)
```

```
MariaDB [ims]> select * from customer;
+-----------+---------+-----------+
| mobno     | cust_id | cname     |
+-----------+---------+-----------+
| 987542548 |     500 | ronaldo   |
| 887552548 |     501 | benzema   |
| 785412684 |     502 | maradona  |
| 785875963 |     503 | nadal     |
| 658775963 |     504 | williams  |
| 954455963 |     505 | tendulkar |
| 987563458 |     506 | akthar    |
| 857463458 |     507 | tyson     |
| 457854581 |     508 | woods     |
| 578245458 |     509 | ozil      |
| 558999857 |     510 | jordan    |
| 896499857 |     511 | brady     |
| 891247457 |     512 | virat     |
| 474587457 |     513 | neeraj    |
| 998574457 |     514 | pawan     |
| 774857777 |     515 | jayasimha |
+-----------+---------+-----------+
16 rows in set (0.000 sec)
```

Selects table:

```
MariaDB [ims]> insert into selects values (5,500,100);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (4,500,101);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into selects values (4,501,114);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (2,502,135);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into selects values (6,503,131);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (7,503,100);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into selects values (1,504,105);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into selects values (10,505,121);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into selects values (8,506,124);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (2,506,117);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (3,507,130);
Query OK, 1 row affected (0.002 sec)
```

```
MariaDB [ims]> insert into selects values (13,507,134);
Query OK, 1 row affected (0.007 sec)

MariaDB [ims]> insert into selects values (12,508,129);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (50,509,110);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (20,509,103);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (10,510,118);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (15,510,139);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into selects values (1,515,119);
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [ims]> select * from selects;
+----------+---------+-----+
| quantity | cust_id | pid |
+----------+---------+-----+
|        5 |     500 | 100 |
|        4 |     500 | 101 |
|        4 |     501 | 114 |
|        2 |     502 | 135 |
|        7 |     503 | 100 |
|        6 |     503 | 131 |
|        1 |     504 | 105 |
|       10 |     505 | 121 |
|        2 |     506 | 117 |
|        8 |     506 | 124 |
|        3 |     507 | 130 |
|       13 |     507 | 134 |
|       12 |     508 | 129 |
|       20 |     509 | 103 |
|       50 |     509 | 110 |
|       10 |     510 | 118 |
|       15 |     510 | 139 |
|        1 |     515 | 119 |
+----------+---------+-----+
18 rows in set (0.000 sec)
```

Transaction table:

```
MariaDB [ims]> insert into transaction values(1000,250000,50000,2,5,'credit card',500);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1001,300000,0,2,5,'debit card',500);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1002,700000,100000,2,3,'upi',501);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1003,40000,40000,2,10,'upi',502);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1004,400000,20000,2,5,'cash',503);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1005,300000,120000,2,0,'credit card',503);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1006,12000,0,2,7,'debit card',504);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1007,15000,0,2,0,'cash',505);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1008,3000,3000,2,0,'cash',506);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1009,300000,20000,2,6,'debit card',506);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1010,21000,0,2,0,'debit card',507);
Query OK, 1 row affected (0.002 sec)
```

```
MariaDB [ims]> insert into transaction values(1011,1000000,300000,2,0,'upi',507);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1012,600000,300000,2,0,'credit card',508);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1013,1000000,800000,2,5,'debit card',509);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1014,100000,0,2,10,'cash',509);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1015,85000,15000,2,0,'cash',510);
Query OK, 1 row affected (0.001 sec)

MariaDB [ims]> insert into transaction values(1016,2500000,500000,2,10,'debit card',510);
Query OK, 1 row affected (0.002 sec)

MariaDB [ims]> insert into transaction values(1017,1000,0,2,0,'upi',515);
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [ims]> select * from transaction;
+------+----------------+--------+------------+-----------------+----------------+---------+
| tid  | total_amt_paid | due    | gstpercent | discountpercent | payment_method | cust_id |
+------+----------------+--------+------------+-----------------+----------------+---------+
| 1000 |         250000 |  50000 |          2 |               5 | credit card    |     500 |
| 1001 |         300000 |      0 |          2 |               5 | debit card     |     500 |
| 1002 |         700000 | 100000 |          2 |               3 | upi            |     501 |
| 1003 |          40000 |  40000 |          2 |              10 | upi            |     502 |
| 1004 |         400000 |  20000 |          2 |               5 | cash           |     503 |
| 1005 |         300000 | 120000 |          2 |               0 | credit card    |     503 |
| 1006 |          12000 |      0 |          2 |               7 | debit card     |     504 |
| 1007 |          15000 |      0 |          2 |               0 | cash           |     505 |
| 1008 |           3000 |   3000 |          2 |               0 | cash           |     506 |
| 1009 |         300000 |  20000 |          2 |               6 | debit card     |     506 |
| 1010 |          21000 |      0 |          2 |               0 | debit card     |     507 |
| 1011 |        1000000 | 300000 |          2 |               0 | upi            |     507 |
| 1012 |         600000 | 300000 |          2 |               0 | credit card    |     508 |
| 1013 |        1000000 | 800000 |          2 |               5 | debit card     |     509 |
| 1014 |         100000 |      0 |          2 |              10 | cash           |     509 |
| 1015 |          85000 |  15000 |          2 |               0 | cash           |     510 |
| 1016 |        2500000 | 500000 |          2 |              10 | debit card     |     510 |
| 1017 |           1000 |      0 |          2 |               0 | upi            |     515 |
+------+----------------+--------+------------+-----------------+----------------+---------+
18 rows in set (0.000 sec)
```

Tools Used

Tool used for front end – Python (Tkinter)

Tool used for back end – mysql database

Queries

Join queries (at least 6)

Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results.

Include 2 regular join, 2 co-related and 2 nested queries

Regular joins:

1) Display the product name, brand id, category id and supplier id of all the products that have been bought by the customers.

Query: select pname,bid,cid,sid from product join selects on product.pid = selects.pid;

```
MariaDB [ims]> select pname,bid,cid,sid from product join selects on product.pid = selects.pid;
+----------------------+-----+-----+------+
| pname                | bid | cid | sid  |
+----------------------+-----+-----+------+
| iphone 10            |   1 |   1 | 4218 |
| iphone 10            |   1 |   1 | 4218 |
| iphone 11            |   1 |   1 | 4218 |
| iphone 13            |   1 |   1 | 4218 |
| galaxy m11           |   2 |   1 | 4218 |
| ultimate sub woofer  |   4 |   3 | 7845 |
| m5 spionic           |   1 |   4 | 7845 |
| play station 1       |   3 |   5 | 7845 |
| play station 3       |   3 |   5 | 7845 |
| web cam 1            |   6 |   5 | 7845 |
| power bank           |   6 |   5 | 7845 |
| 55 inch curved tv    |   4 |   2 | 8147 |
| smart watch s5 ultra |   2 |   5 | 8147 |
| smart watch ez       |   2 |   5 | 8147 |
| apple tv             |   1 |   2 | 9863 |
| iwatch series 8      |   1 |   5 | 9863 |
| play station 4       |   3 |   5 | 9863 |
| hometheatrespeakers  |   3 |   3 | 9863 |
+----------------------+-----+-----+------+
18 rows in set (0.001 sec)
```

2) select the transaction id and customer name of all the customers who have completed their payment using ‘debit card’

Query: select tid,cname from transaction join customer on transaction.cust_id = customer_cust_id where

Payment_method = ‘debit card’;

```
MariaDB [ims]> select tid,cname from transaction join customer on transaction.cust_id = customer.cust_id where payment_method = 'debit card';
+------+----------+
| tid  | cname    |
+------+----------+
| 1001 | ronaldo  |
| 1006 | williams |
| 1009 | akthar   |
| 1010 | tyson    |
| 1013 | ozil     |
| 1016 | jordan   |
+------+----------+
6 rows in set (0.000 sec)
```

Co - related Queries:

1) Display the product details of the all the products bought by the customers and whose price is greater than 50000.

Query: select * from product p where EXISTS (select pid from selects s where p.pid = s.pid AND price > 50000);

```
MariaDB [ims]> select * from product p where EXISTS (select pid from selects s where p.pid = s.pid AND price>50000);
+-----+----------------------+---------+--------+-----+-----+------+
| pid | pname                | p_stock | price  | bid | cid | sid  |
+-----+----------------------+---------+--------+-----+-----+------+
| 100 | iphone 10            |       5 |  60000 |   1 |   1 | 4218 |
| 101 | iphone 11            |       5 |  75000 |   1 |   1 | 4218 |
| 103 | iphone 13            |       5 |  90000 |   1 |   1 | 4218 |
| 114 | m5 spionic           |       8 | 200000 |   1 |   4 | 7845 |
| 129 | smart watch s5 ultra |       3 |  75000 |   2 |   5 | 8147 |
| 131 | apple tv             |       3 |  70000 |   1 |   2 | 9863 |
| 134 | iwatch series 8      |       4 | 100000 |   1 |   5 | 9863 |
| 139 | hometheatrespeakers  |       1 | 200000 |   3 |   3 | 9863 |
+-----+----------------------+---------+--------+-----+-----+------+
8 rows in set (0.002 sec)
```

2) select the names of all the brands which provide discount to the products sold by the supplier with supplier id 4218.

Query: select bname from brand b where EXISTS (select bid from provides p where b.bid = p.bid AND p.sid = 4218);

```
MariaDB [ims]> select bname from brand b where EXISTS(select bid from provides p where b.bid = p.bid AND p.sid = 4218);
+-----------+
| bname     |
+-----------+
| Apple     |
| Samsung   |
| Sony      |
| Panasonic |
+-----------+
4 rows in set (0.000 sec)
```

Nested Queries:

1) select the name and price of all the products which have been bought by the customers with customer id 500 or 507.

Query: select pname.price from products where pid in (select pid from selects where cust_id = 500 or cust_id = 507);

```
MariaDB [ims]> select pname,price from product where pid in (select pid from selects where cust_id = 500 or cust_id = 507);
+-----------------+--------+
| pname           | price  |
+-----------------+--------+
| iphone 10       |  60000 |
| iphone 11       |  75000 |
| smart watch ez  |   7000 |
| iwatch series 8 | 100000 |
+-----------------+--------+
4 rows in set (0.002 sec)
```

2) Display all the transaction details of the customer by name ‘jordan’.

Query: select * from transaction where cust_id in (select cust_id from customer where cname = ‘jordan’).

```
MariaDB [ims]> select * from transaction where cust_id in (select cust_id from customer where cname = 'jordan');
+------+----------------+--------+------------+----------------+----------------+---------+
| tid  | total_amt_paid | due    | gstpercent | discountpercent | payment_method | cust_id |
+------+----------------+--------+------------+----------------+----------------+---------+
| 1015 |          85000 |  15000 |          2 |               0 | cash           |     510 |
| 1016 |        2500000 | 500000 |          2 |              10 | debit card     |     510 |
+------+----------------+--------+------------+----------------+----------------+---------+
2 rows in set (0.001 sec)
```

Aggregate Functions (at least 2)

Showcase at least 2 Aggregate function queries. Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results

1) Total revenue from the sale of the brand with brand id = 1 i.e apple products.

Query: select sum(total_amt_paid) from transaction join selects on transaction.cust_id = selects.cust_id join product on selects.pid = product.pid where bid = 1;

```
MariaDB [ims]> select sum(total_amt_paid) from transaction join selects on transaction.cust_id = selects.cust_id join product on selects.pid = product.pid where bid = 1;
+--------------------+
| sum(total_amt_paid) |
+--------------------+
|            5321000 |
+--------------------+
1 row in set (0.001 sec)
```

2) Display the count of the total number of transactions the inventory management system has recorded.

Query: select count(tid) from transaction;

```
MariaDB [ims]> select count(tid) from transaction;
+------------+
| count(tid) |
+------------+
|         18 |
+------------+
1 row in set (0.000 sec)
```

Set Operations (at least 2)

Showcase at least 2 Set Operations queries. Write the query in English Language, Show the equivalent SQL statement and also screenshot of the query and the results.

1) Display the customer id's of all the customers who have made transactions and order them by their respective customer id's

Query: select cust_id from transaction UNION select cust_id from customer ORDER BY cust_id;

```
MariaDB [ims]> select cust_id from transaction UNION select cust_id from customer ORDER BY cust_id;
+---------+
| cust_id |
+---------+
|     500 |
|     501 |
|     502 |
|     503 |
|     504 |
|     505 |
|     506 |
|     507 |
|     508 |
|     509 |
|     510 |
|     511 |
|     512 |
|     513 |
|     514 |
|     515 |
+---------+
16 rows in set (0.001 sec)
```

2) Display the supplier id of all the suppliers who supply the products and order them by their sid.

Query: select sid from supplier UNION select sid from product order by sid;

```
MariaDB [ims]> select sid from supplier UNION select sid from product order by sid;
+------+
| sid  |
+------+
| 4218 |
| 7845 |
| 8147 |
| 9863 |
+------+
4 rows in set (0.001 sec)
```

View (atleast 1)

Demonstrate creation and querying one view

View to display the product details, customer id, quantity of all products sold which have been supplied by the supplier with supplier id = 4218.

Creation: create view v as(select p.pid,p.pname,p.pstock,p.price,p.bid,p.cid,p.sid,cust_id,quantity from product p , selects s where p.pid = s.pid and p.sid = 4218);

Querying: select * from v;

```
MariaDB [ims]> create view v as(select p.pid,p.pname,p.p_stock,p.price,p.bid,p.cid,p.sid,s.cust_id,s.quantity from product p , selects s where p.pid=s.pid and p.sid = 4218);
Query OK, 0 rows affected (0.003 sec)

MariaDB [ims]> select * from v;
+-----+------------+--------+-------+-----+-----+------+---------+----------+
| pid | pname      | p_stock | price | bid | cid | sid  | cust_id | quantity |
+-----+------------+--------+-------+-----+-----+------+---------+----------+
| 100 | iphone 10  |       5 | 60000 |   1 |   1 | 4218 |     500 |        5 |
| 101 | iphone 11  |       5 | 75000 |   1 |   1 | 4218 |     500 |        4 |
| 100 | iphone 10  |       5 | 60000 |   1 |   1 | 4218 |     503 |        7 |
| 105 | galaxy m11 |       6 | 12000 |   2 |   1 | 4218 |     504 |        1 |
| 103 | iphone 13  |       5 | 90000 |   1 |   1 | 4218 |     509 |       20 |
+-----+------------+--------+-------+-----+-----+------+---------+----------+
5 rows in set (0.002 sec)
```

Triggers (Functions or Procedures)

PROCEDURE:

Create a Function or a Procedure. State the objective of the function / Procedure. Run and

display the results.

Procedure: This procedure takes one input parameter named var where the user provides the id of the respective brand (bid), and the procedure displays all the products and their details of that particular brand and also displays the products that have incurred sales in that particular brand.

Creation:

DELIMITER &&

CREATE PROCEDURE Sales_Details(IN var INT)

BEGIN

select * from product where bid = var;

select pname,price,cid,sid from product p join selects s on p.pid = s.pid where bid = var;

END &&

DELIMITER ;

Calling the stored Procedure: CALL Sales_Details(2);

```
MariaDB [ims]> CALL Sales_Details(2);
+-----+----------------------+---------+--------+-----+-----+------+
| pid | pname                | p_stock | price  | bid | cid | sid  |
+-----+----------------------+---------+--------+-----+-----+------+
| 105 | galaxy m11           |       6 |  12000 |   2 |   1 | 4218 |
| 106 | galaxy flip          |       4 | 120000 |   2 |   1 | 4218 |
| 115 | qled                 |       8 | 100000 |   2 |   2 | 7845 |
| 116 | oled                 |       8 | 180000 |   2 |   2 | 7845 |
| 128 | smart watch f1 pro   |       3 |  51000 |   2 |   5 | 8147 |
| 129 | smart watch s5 ultra |       3 |  75000 |   2 |   5 | 8147 |
| 130 | smart watch ez       |       3 |   7000 |   2 |   5 | 8147 |
+-----+----------------------+---------+--------+-----+-----+------+
7 rows in set (0.001 sec)

+----------------------+-------+-----+------+
| pname                | price | cid | sid  |
+----------------------+-------+-----+------+
| galaxy m11           | 12000 |   1 | 4218 |
| smart watch s5 ultra | 75000 |   5 | 8147 |
| smart watch ez       |  7000 |   5 | 8147 |
+----------------------+-------+-----+------+
3 rows in set (0.014 sec)

Query OK, 0 rows affected (0.019 sec)
```

TRIGGER:

Here a trigger has been created to check weather the user has entered a valid number for the product stock.

Creation:

DELIMITER $$

CREATE TRIGGER stock_check

BEFORE INSERT ON product FOR EACH ROW

BEGIN

DECLARE error_msg VARCHAR(255);

SET error_msg = ('enter valid product stock');

IF NEW.p_stock < 0

THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT = error_msg;

END IF;

END $$

DELIMITER;

```
MariaDB [ims]> DELIMITER $$
MariaDB [ims]> CREATE TRIGGER stock_check
    -> BEFORE INSERT ON product FOR EACH ROW
    -> BEGIN
    -> DECLARE error_msg VARCHAR(255);
    -> SET error_msg = ('enter valid product stock');
    -> IF NEW.p_stock < 0
    -> THEN SIGNAL SQLSTATE '45000'
    -> SET MESSAGE_TEXT = error_msg;
    -> END IF;
    -> END $$
Query OK, 0 rows affected (0.005 sec)

MariaDB [ims]> DELIMITER;
```

This trigger raises an error when invalid stock is entered i.e less than zero.

```
MariaDB [ims]> insert into product values(142,'sdf',-5,5000,1,2,4218);
ERROR 1644 (45000): enter valid product stock
MariaDB [ims]>
```

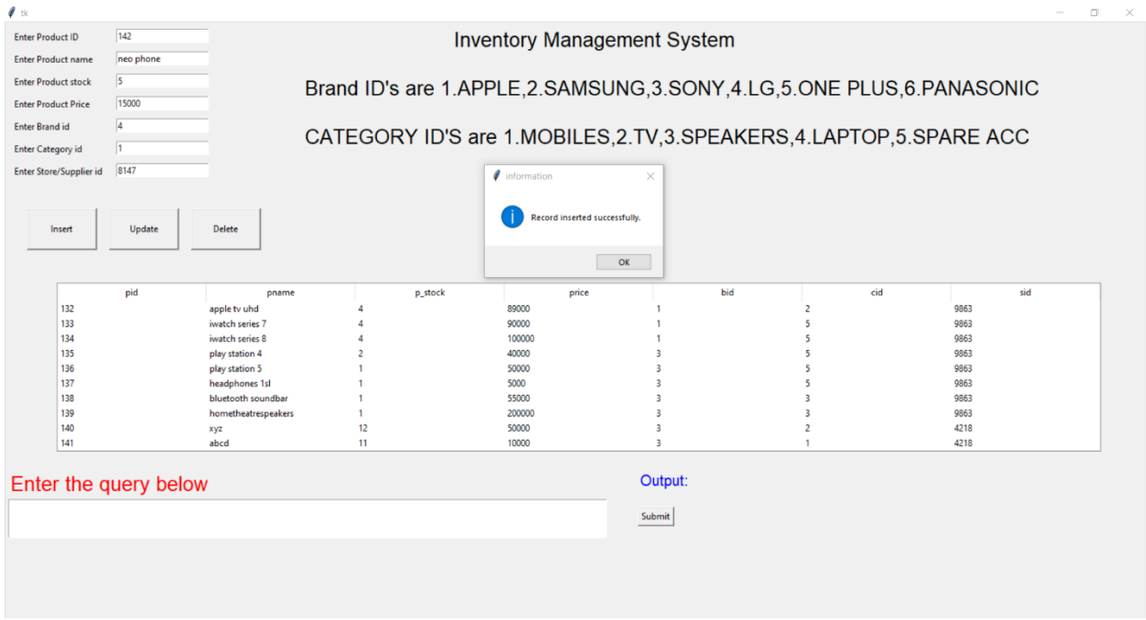
Developing a Frontend

The frontend should support

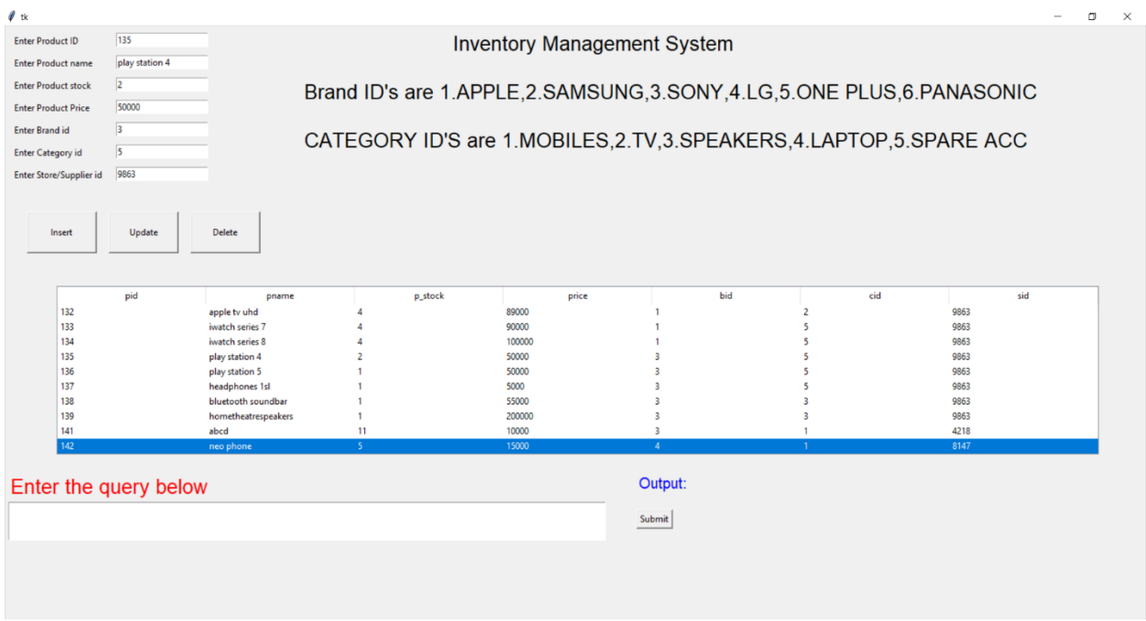
1. Addition, Modification and Deletion of records from any chosen table

2. There should be a window to accept and run any SQL statement and display the result

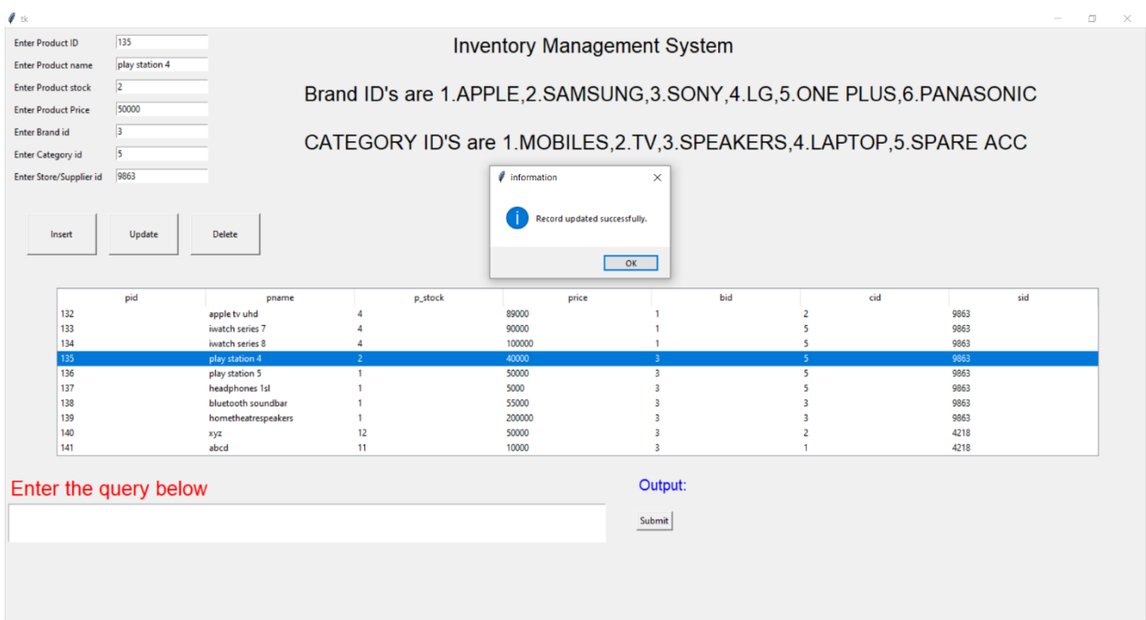
Addition:



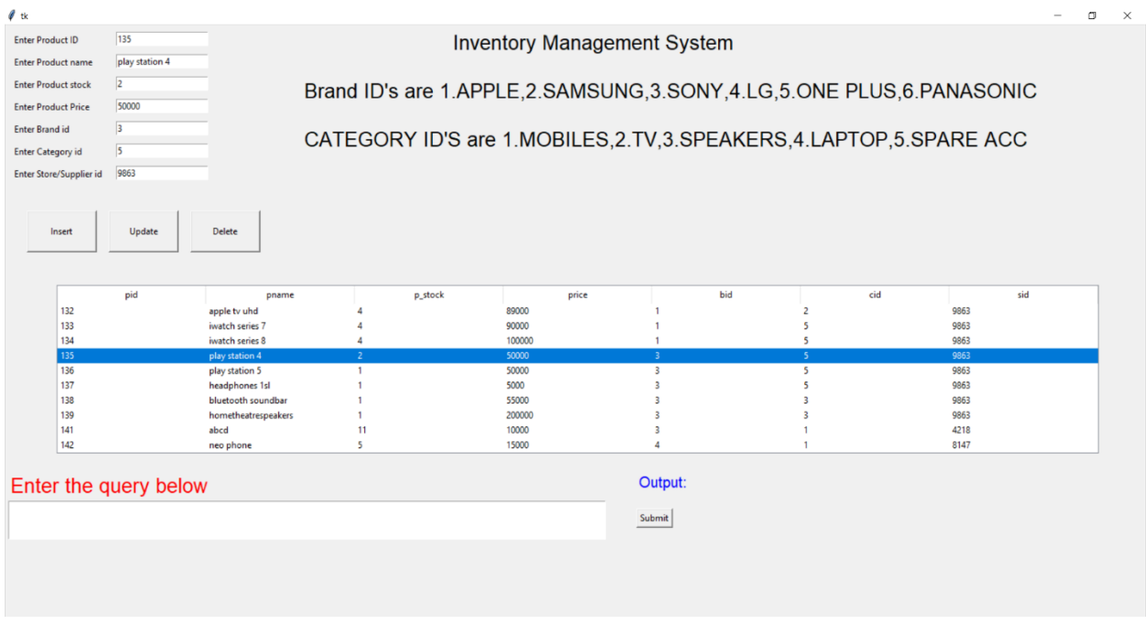
Record inserted successfully:



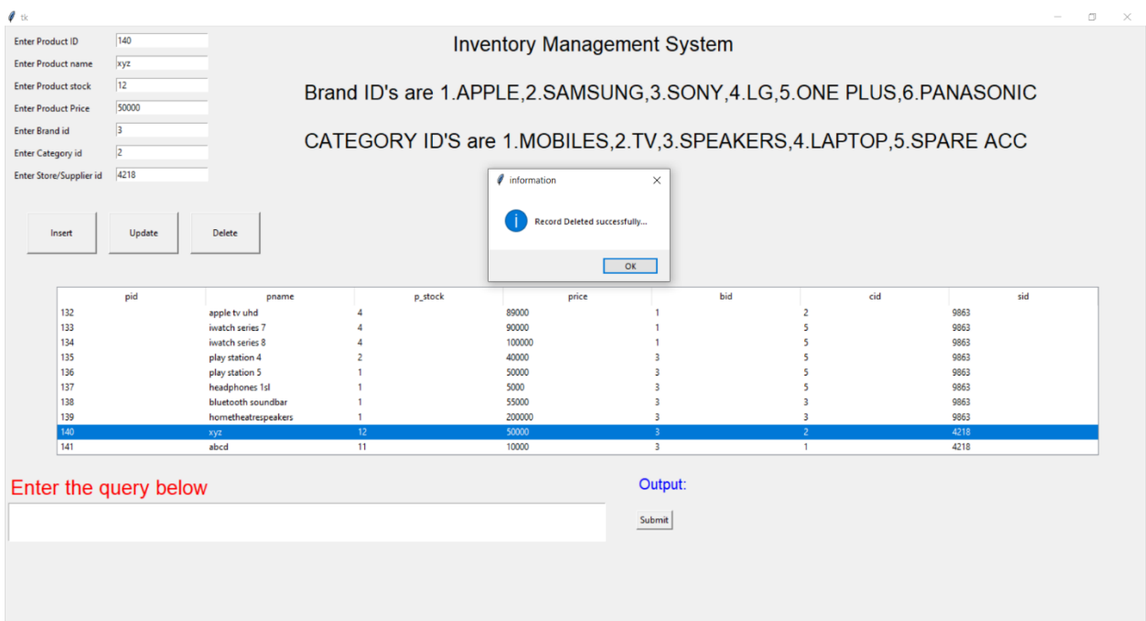
Modification:



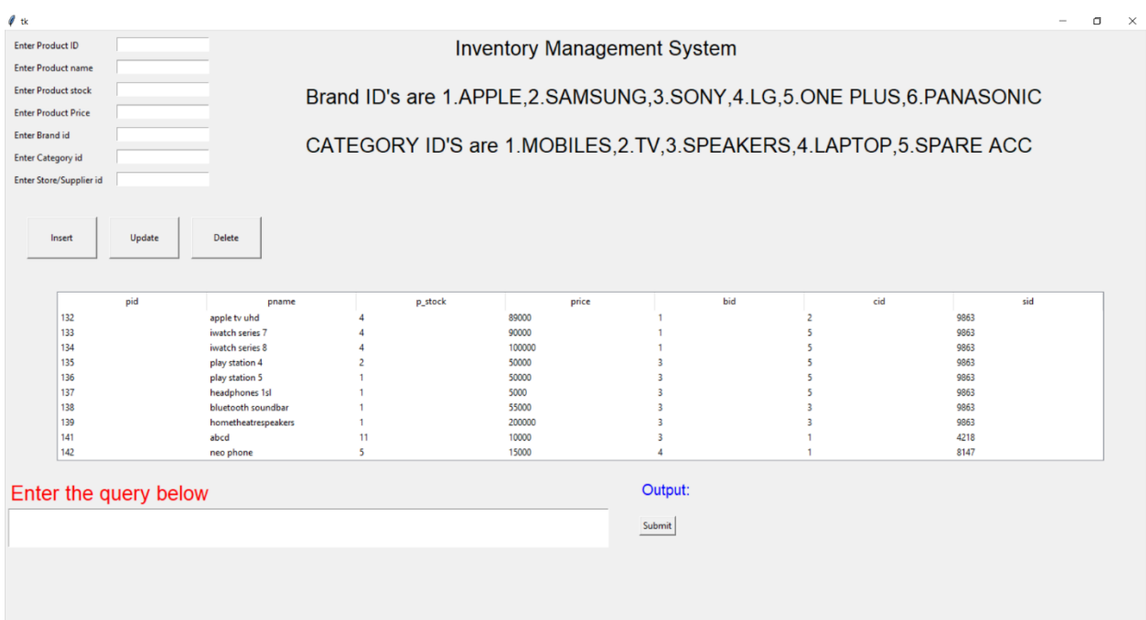
Record Updated successfully:



Deletion:



Record deleted successfully:



2) Window to accept and run any SQL statement and display the result.