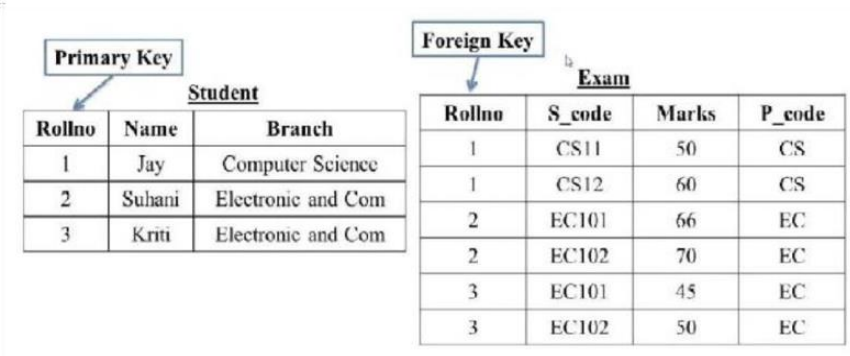
1. Create Table Name : Student and Exam



Student

Rollno	Name	Branch
1	Jay	Computer Science
2	Suhani	Electronic and Com
3	Kriti	Electronic and Com

Exam

Rollno	S_code	Marks	P_code
1	CS11	50	CS
1	CS12	60	CS
2	EC101	66	EC
2	EC102	70	EC
3	EC101	45	EC
3	EC102	50	EC

Ans :

```
CREATE TABLE student
(
   Rollno int not null PRIMARY KEY,
   Name varchar(25),
   Branch varchar(25)
);
INSERT INTO student VALUES(1,'Jay','Computer Science');
INSERT INTO student VALUES(2,'Suhani','Electronic and com');
INSERT INTO student VALUES(3,'Kriti','Electorinic and com');
```

	Rollno	Name	Branch
Edit Copy Delete	1	Jay	Computer Science
Edit Copy Delete	2	Suhani	Electronic and com
Edit Copy Delete	3	Kriti	Electorinic and com

```
CREATE TABLE exam
(
   Rollno int NOT null,
   S_code varchar(30),
   Marks int,
   P_code varchar(30),
   Rollno int,
   FOREIGN KEY(Rollno) REFERENCES student(Rollno),
);
INSERT INTO exam VALUES(1,'CS11',50,'CS');
INSERT INTO exam VALUES(1,'CS12',60,'CS');
INSERT INTO exam VALUES(2,'EC101',66,'EC');
INSERT INTO exam VALUES(2,'EC102',70,'EC');
INSERT INTO exam VALUES(3,'EC101',45,'EC');
INSERT INTO exam VALUES(3,'EC102',50,'EC');
```

Rollno	S_code	Marks	P_code
1	CS11	50	CS
1	CS12	60	CS
2	EC101	66	EC
2	EC102	70	EC
3	EC101	45	EC
3	EC102	50	EC

2. Create table given below: Employee and IncentiveTable

Employee_id	First_name	Last_name	Salary	Joining_date	Department
1	John	Abraham	1000000	01-JAN-13 12.00.00 AM	Banking
2	Michael	Clarke	800000	01-JAN-13 12.00.00 AM	Insurance
3	Roy	Thomas	700000	01-FEB-13 12.00.00 AM	Banking
4	Tom	Jose	600000	01-FEB-13 12.00.00 AM	Insurance
5	Jerry	Pinto	650000	01-FEB-13 12.00.00 AM	Insurance
6	Philip	Mathew	750000	01-JAN-13 12.00.00 AM	Services
7	TestName1	123	650000	01-JAN-13 12.00.00 AM	Services
8	TestName2	Lname%	600000	01-FEB-13 12.00.00 AM	Insurance

Name: Employee

Table Name:

Incentive

Employee_ref_id	Incentive_date	Incentive_amount
1	01-FEB-13	5000
2	01-FEB-13	3000
3	01-FEB-13	4000
1	01-JAN-13	4500
2	01-JAN-13	3500

Ans :

```
CREATE TABLE Employee
(
   Employee_id int NOT null PRIMARY KEY,
   First_name varchar(25),
   Last_name varchar(25),
   Salary int,
   Joining_date varchar(30),
   Department varchar(30)
);
INSERT INTO employee VALUES (1,'John','Abraham',1000000,'01-JAN-03 12.00.00
AM','Banking');
INSERT INTO employee VALUES (2,'Michael','Clarke',800000,'01-JAN-03 12.00.00
AM','Insurance');
INSERT INTO employee VALUES (3,'Roy','Thomas',700000,'01-FEB-03 12.00.00
AM','Banking');
INSERT INTO employee VALUES (4,'Tom','Jose',600000,'01-FEB-03 12.00.00 AM','Insurance');
INSERT INTO employee VALUES (5,'Jerry','Pinto',650000,'01-FEB-03 12.00.00
AM','Insurance');
INSERT INTO employee VALUES (6,'Philip','Mathew',750000,'01-JAN-03 12.00.00
AM','Services');
INSERT INTO employee VALUES (7,'Testname1','123',650000,'01-JAN-03 12.00.00
AM','Services');
INSERT INTO employee VALUES (8,'Testname2','Lname%',600000,'01-FEB-03 12.00.00
AM','Insurance');
```

	Employee_id	First_name	Last_name	Salary	Joining_date	Department
Edit Copy Delete	1	John	Abraham	1000000	01-JAN-03 12.00.00 AM	Banking
Edit Copy Delete	2	Michael	Clarke	800000	01-JAN-03 12.00.00 AM	Insurance
Edit Copy Delete	3	Roy	Thomas	700000	01-FEB-03 12.00.00 AM	Banking
Edit Copy Delete	4	Tom	Jose	600000	01-FEB-03 12.00.00 AM	Insurance
Edit Copy Delete	5	Jerry	Pinto	650000	01-FEB-03 12.00.00 AM	Insurance
Edit Copy Delete	6	Philip	Mathew	750000	01-JAN-03 12.00.00 AM	Services
Edit Copy Delete	7	Testname1	123	650000	01-JAN-03 12.00.00 AM	Services
Edit Copy Delete	8	Testname2	Lname%	600000	01-FEB-03 12.00.00 AM	Insurance

```
CREATE TABLE Incentive
(
   Employee_id int,
   Incentive_date varchar(30),
   Incentive_amount int,
   FOREIGN KEY(Employee_id) REFERENCES employee(Employee_id)
);
INSERT INTO incentive VALUES (1,'01-FEB-2013',5000);
INSERT INTO incentive VALUES (2,'01-FEB-2013',3000);
INSERT INTO incentive VALUES (3,'01-FEB-2013',4000);
INSERT INTO incentive VALUES (1,'01-JAN-2013',4500);
INSERT INTO incentive VALUES (2,'01-JAN-2013',3500);
```

Employee_id	Incentive_date	Incentive_amount
1	01-FEB-2013	5000
2	01-FEB-2013	3000
3	01-FEB-2013	4000
1	01-JAN-2013	4500
2	01-JAN-2013	3500

3. Get First_Name from employee table using Tom name “Employee Name”.

Ans :

```
SELECT First_name FROM employee WHERE First_name='Tom';
```

	First_name
Edit Copy Delete	Tom

4. Get FIRST_NAME, Joining Date, and Salary from employee table.

Ans :

```
SELECT First_name ,Joining_date ,Salary FROM employee;
```

	First_name	Joining_date	Salary
Edit Copy Delete	John	01-JAN-03 12.00.00 AM	1000000
Edit Copy Delete	Michael	01-JAN-03 12.00.00 AM	800000
Edit Copy Delete	Roy	01-FEB-03 12.00.00 AM	700000
Edit Copy Delete	Tom	01-FEB-03 12.00.00 AM	600000
Edit Copy Delete	Jerry	01-FEB-03 12.00.00 AM	650000
Edit Copy Delete	Philip	01-JAN-03 12.00.00 AM	750000
Edit Copy Delete	Testname1	01-JAN-03 12.00.00 AM	650000
Edit Copy Delete	Testname2	01-FEB-03 12.00.00 AM	600000

5. Get all employee details from the employee table order by First_Name Ascending and Salary descending?

Ans :

SELECT * FROM Employee ORDER BY First_name ,Salary DESC;

	Employee_id	First_name 1	Last_name	Salary 2	Joining_date	Department
Edit Copy Delete	5	Jerry	Pinto	650000	01-FEB-03 12.00.00 AM	Insurance
Edit Copy Delete	1	John	Abraham	1000000	01-JAN-03 12.00.00 AM	Banking
Edit Copy Delete	2	Michael	Clarke	800000	01-JAN-03 12.00.00 AM	Insurance
Edit Copy Delete	6	Philip	Mathew	750000	01-JAN-03 12.00.00 AM	Services
Edit Copy Delete	3	Roy	Thomas	700000	01-FEB-03 12.00.00 AM	Banking
Edit Copy Delete	7	Testname1	123	650000	01-JAN-03 12.00.00 AM	Services
Edit Copy Delete	8	Testname2	Lname%	600000	01-FEB-03 12.00.00 AM	Insurance
Edit Copy Delete	4	Tom	Jose	600000	01-FEB-03 12.00.00 AM	Insurance

6. Get employee details from employee table whose first name contains ‘J’.

Ans :

SELECT * FROM Employee WHERE First_name LIKE 'J%';

	Employee_id	First_name	Last_name	Salary	Joining_date	Department
Edit Copy Delete	1	John	Abraham	1000000	01-JAN-03 12.00.00 AM	Banking
Edit Copy Delete	5	Jerry	Pinto	650000	01-FEB-03 12.00.00 AM	Insurance

7. Get department wise maximum salary from employee table order by salaryascending?

Ans :

SELECT Department, MAX(Salary) FROM employee GROUP BY Department;

	Department	MAX(Salary)
Edit Copy Delete	Banking	1000000
Edit Copy Delete	Insurance	800000
Edit Copy Delete	Services	750000

9. Select first_name, incentive amount from employee and incentives table forthose employees who have incentives and incentive amount greater than 3000

Ans :

SELECT First_Name, Incentive_amount FROM employee

JOIN incentive ON employee.Employee_id=incentive.Employee_id WHERE incentive.Incentive_amount>3000;

First_Name	Incentive_amount
John	5000
Roy	4000
John	4500
Michael	3500

10. Create After Insert trigger on Employee table which insert records in viewtable.

Ans :

```
        CREATE TABLE viewtable
(
   Employee_id int,
   First_name varchar(20),
   Last_name varchar(20),
   Salary int,
   Joining_date varchar(20),
   Department varchar(20),
   Date_Time timestamp,
   Action_Performed text
);
DELIMITER $$

CREATE TRIGGER employee1 AFTER INSERT ON employee FOR EACH ROW
BEGIN
                INSERT INTO
viewtable(Employee_ID,First_name,Last_name,Salary,Joining_date,Department,
action_performed)

VALUES(new.Employee_ID,new.First_name,new.Last_name,new.Salary,new.Joining_date,
new.Department,'Record inserted');
END
INSERT INTO employee VALUES(9,'James','Bond',800000,'1-Jan-13 12.00.00 AM','Banking');
```

Employee_id	First_name	Last_name	Salary	Joining_date	Department	Date_Time	Action_Performed
9	James	Bond	800000	1-Jan-13 12.00.00 AM	Banking	2024-09-06 19:11:16	Record inserted

11. Create table given below: Salesperson and Customer

TABLE-1

TABLE NAME- SALSEPERSON

(PK)SNo	SNAME	CITY	COMM
1001	Peel	London	.12
1002	Serres	San Jose	.13
1004	Motika	London	.11
1007	Rafkin	Barcelona	.15
1003	Axelrod	New York	.1

TABLE-2

TABLE NAME- CUSTOMER

(PK)CNM.	CNAME	CITY	RATING	(FK)SNo
201	Hoffman	London	100	1001
202	Giovanne	Roe	200	1003
203	Liu	San Jose	300	1002
204	Grass	Barcelona	100	1002
206	Clemens	London	300	1007
207	Pereira	Roe	100	1004

Ans :

```
CREATE TABLE Salesperson
(
  PKSNo int PRIMARY KEY,
  SNAME varchar(20),
  CITY varchar(20),
  COMM varchar(20)
)
INSERT INTO salesperson VALUES(1001,'Peel','London','.12');
INSERT INTO salesperson VALUES(1002,'Serres','San Jose','.13');
INSERT INTO salesperson VALUES(1004,'Motika','London','.11');
INSERT INTO salesperson VALUES(1007,'Rafkin','Barcelona','.15');
INSERT INTO salesperson VALUES(1003,'Axelrod','New York','.1');
```

			PKSNo	SNAME	CITY	COMM
☐	Edit	Copy Delete	1001	Peel	London	.12
☐	Edit	Copy Delete	1002	Serres	San Jose	.13
☐	Edit	Copy Delete	1003	Axelrod	New York	.1
☐	Edit	Copy Delete	1004	Motika	London	.11
☐	Edit	Copy Delete	1007	Rafkin	Barcelona	.15

```
CREATE TABLE customer
(
```

```
    PKCNM int,
    CNAME varchar(20),
    CITY varchar(20),
    RATING int,
    PKSNo int,
    FOREIGN KEY(PKSNo) REFERENCES salesperson(PKSNo)
);
INSERT INTO customer VALUES(201,'Hoffman','London',100,1001);
INSERT INTO customer VALUES(202,'Giovanne','Roe',200,1003);
INSERT INTO customer VALUES(203,'Liu','San Jose',300,1002);
INSERT INTO customer VALUES(204,'Grass','Barcelona',100,1002);
INSERT INTO customer VALUES(206,'Clemens','London',300,1007);
INSERT INTO customer VALUES(207,'Pereira','Reo',100,1004);
```

PKCNM	CNAME	CITY	RATING	PKSNo
201	Hoffman	London	100	1001
202	Giovanne	Roe	200	1003
203	Liu	San Jose	300	1002
204	Grass	Barcelona	100	1002
206	Clemens	London	300	1007
207	Pereira	Reo	100	1004

13. All orders for more than $1000.

    Ans :

```
SELECT * FROM customer WHERE amount>1000;
```

14. Names and cities of all salespeople in London with commission above 0.12

    Ans :

```
SELECT SNAME, CITY FROM salesperson WHERE CITY='London' AND COMM>0.12;
```

SNAME	CITY

15. All salespeople either in Barcelona or in London

    Ans :

```
SELECT * FROM salesperson WHERE CITY='Barcelona' OR CITY='London';
```

	PKSNo	SNAME	CITY	COMM
Edit Copy Delete	1001	Peel	London	.12
Edit Copy Delete	1004	Motika	London	.11
Edit Copy Delete	1007	Rafkin	Barcelona	.15

16. All salespeople with commission between 0.10 and 0.12. (Boundary valuesshould be excluded).

Ans :

SELECT * FROM salesperson WHERE COMM BETWEEN 0.10 AND 0.12;

	PKSNo	SNAME	CITY	COMM
Edit Copy Delete	1001	Peel	London	.12
Edit Copy Delete	1003	Axelrod	New York	.1
Edit Copy Delete	1004	Motika	London	.11

17. All customers excluding those with rating <= 100 unless they are located in Rome.

Ans :

SELECT * FROM customer WHERE rating<=100 AND CITY='Reo';

PKCNM	CNAME	CITY	RATING	PKSNo
207	Pereira	Reo	100	1004

18. Write a SQL statement that displays all the information about all salespeople.

Ans :

SELECT * FROM salesperson;

	PKSNo	SNAME	CITY	COMM
Edit Copy Delete	1001	Peel	London	.12
Edit Copy Delete	1002	Serres	San Jose	.13
Edit Copy Delete	1003	Axelrod	New York	.1
Edit Copy Delete	1004	Motika	London	.11
Edit Copy Delete	1007	Rafkin	Barcelona	.15

19. From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord_no, ord_date, purch_amt.

```
salesman_id |    name     |   city    | commission
------------------+----------------+-------------+----------------
5001 | James Hoog | New York |        0.15
5002 | Nail Knite | Paris    |        0.13
5005 | Pit Alex   | London   |        0.11
5006 | Mc Lyon    | Paris    |        0.14
5007 | Paul Adam  | Rome     |        0.13
5003 | Lauson Hen | San Jose |        0.12
```

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70007	948.5	2012-09-10	3005	5002
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001

Ans :

```
CREATE TABLE salesman
(
 salesman_id int PRIMARY KEY,
```

```
        name varchar(20),
        city varchar(20),
        commission varchar(20)
);
INSERT INTO salesman VALUES (5001,'James Hoog','New York','0.15');
INSERT INTO salesman VALUES (5002,'Nail Knite','Paris ','0.13');
INSERT INTO salesman VALUES (5005,'Pit Alex','London','0.11');
INSERT INTO salesman VALUES (5006,'Mc Lyon','Paris','0.14');
INSERT INTO salesman VALUES (5007,'Paul Adam','Rome','0.13');
INSERT INTO salesman VALUES (5003,'Lauson Hen','San Jose','0.12');
```

	salesman_id	name	city	commission
Edit Copy Delete	5001	James Hoog	New York	0.15
Edit Copy Delete	5002	Nail Knite	Paris	0.13
Edit Copy Delete	5003	Lauson Hen	San Jose	0.12
Edit Copy Delete	5005	Pit Alex	London	0.11
Edit Copy Delete	5006	Mc Lyon	Paris	0.14
Edit Copy Delete	5007	Paul Adam	Rome	0.13

```
CREATE TABLE orders
(
   ord_no int PRIMARY KEY,
        purch_amt varchar(20),
        ord_date date,
        customer_id int,
        salesman_id int,
        FOREIGN KEY(salesman_id) REFERENCES salesman(salesman_id)
);
INSERT INTO orders VALUES (70001,'150.5','2012-10-05','3005','5002');
INSERT INTO orders VALUES (70009,'270.65','2012-09-10','3001','5005');
INSERT INTO orders VALUES (70002,'65.26','2012-10-05','3002','5001');
INSERT INTO orders VALUES (70004,'110.5','2012-08-17','3009','5003');
INSERT INTO orders VALUES (70007,'948.5','2012-09-10','3005','5002');
INSERT INTO orders VALUES (70005,'2400.6','2012-07-27','3007','5001');
INSERT INTO orders VALUES (70008,'5760','2012-09-10','3002','5001');
INSERT INTO orders VALUES (70010,'1983.43','2012-10-10','3004','5006');
INSERT INTO orders VALUES (70003,'2480.4','2012-10-10','3009','5003');
INSERT INTO orders VALUES (70012,'250.45','2012-06-27','3008','5002');
INSERT INTO orders VALUES (70011,'75.29','2012-08-17','3003','5007');
INSERT INTO orders VALUES (70013,'3045.6','2012-04-25','3002','5001');
```

	ord_no	purch_amt	ord_date	customer_id	salesman_id
Edit Copy Delete	70001	150.5	2012-10-05	3005	5002
Edit Copy Delete	70002	65.26	2012-10-05	3002	5001
Edit Copy Delete	70003	2480.4	2012-10-10	3009	5003
Edit Copy Delete	70004	110.5	2012-08-17	3009	5003
Edit Copy Delete	70005	2400.6	2012-07-27	3007	5001
Edit Copy Delete	70007	948.5	2012-09-10	3005	5002
Edit Copy Delete	70008	5760	2012-09-10	3002	5001
Edit Copy Delete	70009	270.65	2012-09-10	3001	5005
Edit Copy Delete	70010	1983.43	2012-10-10	3004	5006
Edit Copy Delete	70011	75.29	2012-08-17	3003	5007
Edit Copy Delete	70012	250.45	2012-06-27	3008	5002
Edit Copy Delete	70013	3045.6	2012-04-25	3002	5001

SELECT ord_no, ord_date, purch_amt FROM orders WHERE salesman_id=5001;

	ord_no	ord_date	purch_amt
Edit Copy Delete	70002	2012-10-05	65.26
Edit Copy Delete	70005	2012-07-27	2400.6
Edit Copy Delete	70008	2012-09-10	5760
Edit Copy Delete	70013	2012-04-25	3045.6

20. From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro_id, pro_name, pro_price, and pro_com.

Sample table: item_mast

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

Ans :

```
CREATE TABLE item_mast
(
  PRO_ID int,
  PRO_NAME varchar(20),
  PRO_PRICE varchar(20),
  PRO_COM int
);
INSERT INTO item_mast VALUES (101,'Mother Board','3200.00',15);
INSERT INTO item_mast VALUES (102,'Key Board','450.00',16);
INSERT INTO item_mast VALUES (103,'ZIP drive','250.00',14);
INSERT INTO item_mast VALUES (104,'Speaker','550.00',16);
INSERT INTO item_mast VALUES (105,'Monitor','5000.00',11);
INSERT INTO item_mast VALUES (106,'DVD drive','900.00',12);
INSERT INTO item_mast VALUES (107,'CD drive','800.00',12);
INSERT INTO item_mast VALUES (108,'Printer','2600.00',13);
INSERT INTO item_mast VALUES (109,'Refill cartridge','350.00',13);
INSERT INTO item_mast VALUES (110,'Mouse','250.00',12);
```

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

SELECT pro_id, pro_name, pro_price, pro_com FROM item_mast WHERE pro_price BETWEEN 200 AND 600;

pro_id	pro_name	pro_price	pro_com
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

21. From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg.

Sample table: item_mast

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

Ans :

```
SELECT AVG(PRO_PRICE) AS avg FROM item_mast WHERE PRO_COM=16;
```

avg
500

22. From the following table, write a SQL query to display the pro_name as 'Item Name' and pro_priceas 'Price in Rs.'

Sample table: item_mast

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

Ans :

SELECT PRO_NAME AS 'Item Name', PRO_PRICE AS'Price in Rs.' FROM item_mast;

Item Name	Price in Rs.
Mother Board	3200.00
Key Board	450.00
ZIP drive	250.00
Speaker	550.00
Monitor	5000.00
DVD drive	900.00
CD drive	800.00
Printer	2600.00
Refill cartridge	350.00
Mouse	250.00

23. From the following table, write a SQL query to find the items whose prices are higher than or equal to $250. Order the result by product price in descending, then product name in ascending. Return pro_name and pro_price.

Sample table: item_mast

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

Ans :

SELECT PRO_NAME,PRO_PRICE FROM item_mast WHERE PRO_PRICE>=250 ORDER BY PRO_NAME ASC,PRO_PRICE DESC;

PRO_NAME ▲ 1	PRO_PRICE ▼ 2
CD drive	800.00
DVD drive	900.00
Key Board	450.00
Monitor	5000.00
Mother Board	3200.00
Mouse	250.00
Printer	2600.00
Refill cartridge	350.00
Speaker	550.00
ZIP drive	250.00

24. From the following table, write a SQL query to calculate average price of the items for each company. Return average price and company code.

Sample table: item_mast

PRO_ID	PRO_NAME	PRO_PRICE	PRO_COM
101	Mother Board	3200.00	15
102	Key Board	450.00	16
103	ZIP drive	250.00	14
104	Speaker	550.00	16
105	Monitor	5000.00	11
106	DVD drive	900.00	12
107	CD drive	800.00	12
108	Printer	2600.00	13
109	Refill cartridge	350.00	13
110	Mouse	250.00	12

Ans :

```
SELECT AVG(PRO_PRICE)AS average_price,PRO_COM AS companycode FROM item_mast GROUP BY PRO_COM;
```

average_price	companycode
5000	11
650	12
1475	13
250	14
3200	15
500	16