Lab Record

Name: Alok Kumar Rastogi

Course code: 19CS4PCDBM

USN: 1BM19CS192

Course Name: DBMS Lab

Lab Program: 1: - INSURANCE DATABASE

```
create
database
insurance;
```

```
create table person (
driver_id varchar(10),
 name varchar(20) ,
 address varchar(100),
 primary key(driver_id)
);
create table car
(
reg_num varchar(10),
model varchar(10),
year int,
primary key(reg_num)
);

create table accident
(
report_num int,
accident_date date,
location varchar(20),
primary key(report_num)
);

create table owns
(
driver_id varchar(10),
reg_num varchar(10),
```

```
primary key(driver_id,reg_num),
foreign key(driver_id) references person(driver_id),
foreign key(reg_num) references car(reg_num)
);

create table participated
(
driver_id varchar(10),
reg_num varchar(10),
report_num int ,
damage_amount int,
primary key(driver_id,reg_num,report_num),
foreign key(driver_id) references person(driver_id),
foreign key(reg_num) references car(reg_num),
foreign key(report_num) references accident(report_num)
);
select *from car;

use insurance;
insert into person values('A01', 'Richard', 'Srinivas Nagar');
insert into person values('A02', 'Pradeep', 'Rajaji Nagar');
insert into person values('A03', 'Smith', 'Ashok Nagar');
insert into person values('A04', 'Viru', 'N. R Colony');
insert into person values('A05', 'John', 'HanumanthNagar');
select * from person;

insert into car values('KA052255', 'Indica', '1990');
insert into car values('KA052251', 'Lacer', '1957');
insert into car values('KA052252', 'Tyota', '1998');
insert into car values('KA052253', 'Honda', '2008');
insert into car values('KA052254', 'Audi', '2005');
select * from car;

insert into accident values('11', '2002-03-01', 'Basvangudi Road');
insert into accident values('12', '2008-04-05', 'KANAKPURA Road');
insert into accident values('13', '2000-09-10', 'Ring Road');
insert into accident values('14', '2004-05-12', 'Mysore Road');
insert into accident values('15', '2003-07-28', 'Mysore Road');
select * from accident;

insert into owns values('A01', 'KA052255');
insert into owns values('A02', 'KA052251');
insert into  owns values('A03', 'KA052252');
```

```
insert into  owns values('A04', 'KA052253');
insert into  owns values('A05', 'KA052254');
select * from owns;

INSERT INTO PARTICIPATED VALUES('A01', 'KA052255', 11, 10000);
INSERT INTO PARTICIPATED VALUES('A02', 'KA052251', 12, 50000);
INSERT INTO PARTICIPATED VALUES('A03', 'KA052252', 13, 25000);
INSERT INTO PARTICIPATED VALUES('A04', 'KA052253', 14, 3000);
INSERT INTO PARTICIPATED VALUES('A05', 'KA052254', 15, 5000);
select * from participated;

UPDATE PARTICIPATED SET DAMAGE_AMOUNT = 25000 WHERE REPORT_NUM = 12;
select *from participated;

INSERT INTO ACCIDENT VALUES (16, '2008-02-21', 'Bulltemple Road');
select * from accident;



SELECT COUNT(DISTINCT DRIVER_ID) FROM ACCIDENT, PARTICIPATED
WHERE ACCIDENT.REPORT_NUM = PARTICIPATED.REPORT_NUM
AND ACCIDENT_DATE LIKE '2008%';

SELECT COUNT(REPORT_NUM) FROM CAR, PARTICIPATED
WHERE CAR.REG_NUM = PARTICIPATED.REG_NUM
AND MODEL = "AUDI";
```

Tables And Outpus:

1.

```
use insurance;
insert into person values('A01', 'Richard', 'Srinivas Nagar');
insert into person values('A02', 'Pradeep', 'Rajaji Nagar');
insert into person values('A03', 'Smith', 'Ashok Nagar');
insert into person values('A04', 'Viru', 'N.R Colony');
insert into person values('A05', 'John', 'HanumanthNagar');
```

esult Grid | Filter Rows: | Edit: | Export/Import:

name	driver_id	address
A05	John	HanumanthNagar
A02	Pradeep	Rajaji Nagar
A01	Richard	Srinivas Nagar
A03	Smith	Ashok Nagar
A04	Viru	N.R Colony
NULL	NULL	NULL

```
52 •   insert into car values('KA0522558', 'Indica', '1990');
53 •   insert into car values('KA052251', 'Lacer', '1957');
54 •   insert into car values('KA052252', 'Tyota', '1998');
55 •   insert into car values('KA052253', 'Honda', '2008');
56 •   insert into car values('KA052254', 'Audi', '2005');
57 •   select * from car;
58
59
```

Result Grid | Filter Rows: | Edit: | Export/Import:

reg_num	model	year
KA052250	Indica	1990
KA052251	Lacer	1957
KA052252	Tyota	1998
KA052253	Honda	2008
KA052254	Audi	2005
KA0522558	Indica	1990
NULL	NULL	NULL

```
59 •   insert into  accident values('11', '2002-03-01', 'Basvangudi Road');
60 •   insert into  accident values('12', '200-04-05', 'KANAKPURA Road');
61 •   insert into  accident values('13', '2000-09-10', 'Ring Road');
62 •   insert into  accident values('14', '2004-05-12', 'Mysore Road');
63 •   insert into  accident values('15', '2003-07-28', 'Mysore Road');
64 •   select * from accident;
65
66
67
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Conter

report_num	accident_date	location
11	2002-03-01	Basvangudi Road
12	0200-04-05	KANAKPURA Road
13	2000-09-10	Ring Road
14	2004-05-12	Mysore Road
15	2003-07-28	Mysore Road
NULL	NULL	NULL

```
68 •    insert into  owns values('A01', 'KA052255');
69 •    insert into  owns values('A02', 'KA052251');
70 •    insert into  owns values('A03', 'KA052252');
71 •    insert into  owns values('A04', 'KA052253');
72 •    insert into  owns values('A05', 'KA052254');
73 •    select * from owns;
```

Result Grid | Filter Rows: | Edit: | Export/Im

driver_id	reg_num
A02	KA052251
A03	KA052252
A04	KA052253
A05	KA052254
A01	KA052255
NULL	NULL

Result Grid | Filter Rows: | Edit:

driver_id	reg_num	report_num	damage_amount
A01	KA052255	11	10000
A02	KA052251	12	50000
A03	KA052252	13	25000
A04	KA052253	14	3000
A05	KA052254	15	5000
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Imp

driver_id	reg_num	report_num	damage_amount
A01	KA052255	11	10000
A02	KA052251	12	25000
A03	KA052252	13	25000
A04	KA052253	14	3000
A05	KA052254	15	5000
NULL	NULL	NULL	NULL

participated 8 ×

Result Grid | Filter Rows: | Edit:

report_num	accident_date	location
11	2002-03-01	Basvangudi Road
12	0200-04-05	KANAKPURA Road
13	2000-09-10	Ring Road
14	2004-05-12	Mysore Road
15	2003-07-28	Mysore Road
16	2008-02-21	Bulltemple Road
NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

COUNT(DISTINCT DRIVER_ID)
1

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

COUNT(REPORT_NUM)
1

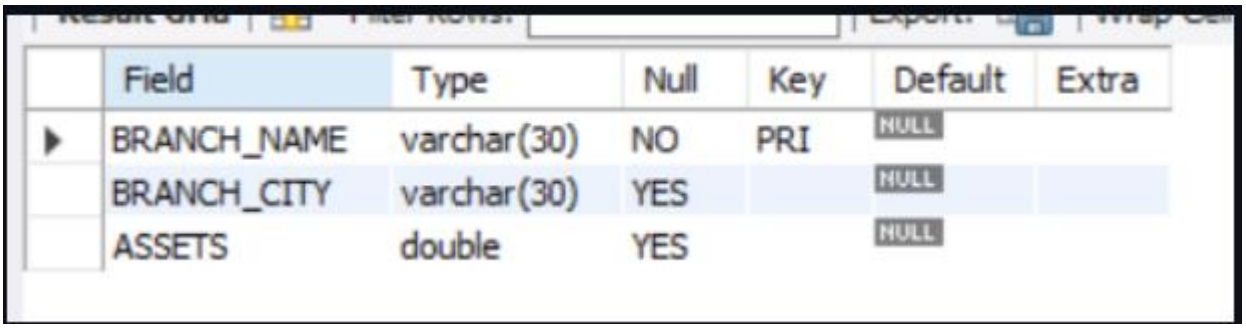
Lab Program 2:- Banking Enterprise Database

```
create
database
banking;
```

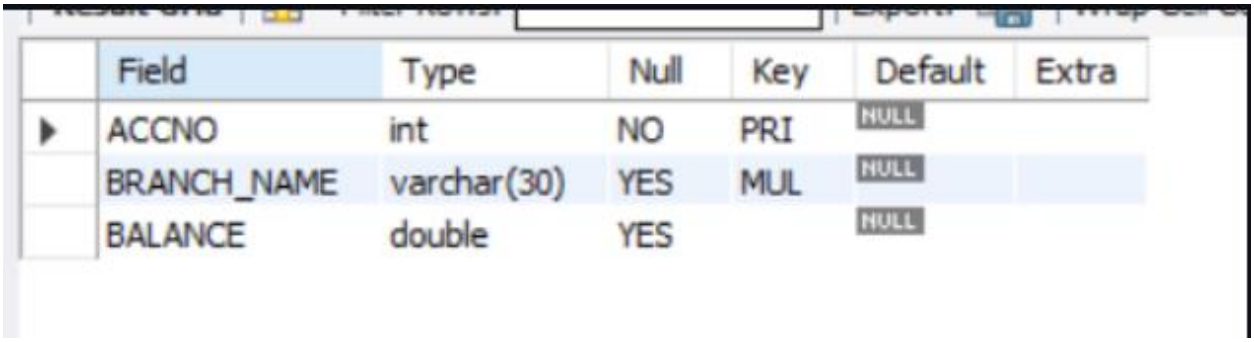
```
CREATE TABLE BRANCH (BRANCH_NAME VARCHAR(30), BRANCH_CITY VARCHAR(30), ASSETS REAL,
PRIMARY KEY (BRANCH_NAME));
CREATE TABLE BANK_ACCOUNT (ACCNO INT, BRANCH_NAME VARCHAR(30), BALANCE REAL, PRIMARY KEY
(ACCNO), FOREIGN KEY (BRANCH_NAME) REFERENCES BRANCH(BRANCH_NAME));
CREATE TABLE BANK_CUSTOMER (CUSTOMER_NAME VARCHAR(30), CUSTOMER_STREET VARCHAR(30),
CUSTOMER_CITY VARCHAR(30), PRIMARY KEY(CUSTOMER_NAME));
CREATE TABLE DEPOSITER (CUSTOMER_NAME VARCHAR(30), ACCNO INT, PRIMARY KEY(CUSTOMER_NAME,
ACCNO), FOREIGN KEY (CUSTOMER_NAME) REFERENCES BANK_CUSTOMER(CUSTOMER_NAME), FOREIGN KEY
(ACCNO) REFERENCES BANK_ACCOUNT(ACCNO));

CREATE TABLE LOAN (LOAN_NUMBER INT, BRANCH_NAME VARCHAR(30), AMOUNT REAL, PRIMARY KEY
(LOAN_NUMBER), FOREIGN KEY (BRANCH_NAME) REFERENCES BRANCH(BRANCH_NAME));



INSERT INTO BRANCH VALUES ('SBI_CHAMRAJPET', 'BANGALORE', 50000);
INSERT INTO BRANCH VALUES ('SBI_RESIDENCYROAD', 'BANGALORE', 10000);
INSERT INTO BRANCH VALUES ('SBI_SHIVAJIROAD', 'BOMBAY', 20000);
INSERT INTO BRANCH VALUES ('SBI_PARLIAMENTROAD', 'DELHI', 10000);
INSERT INTO BRANCH VALUES ('SBI_JANTARMANTAR', 'DELHI', 20000);

INSERT INTO BANK_ACCOUNT VALUES ( 1,'SBI_CHAMRAJPET', 2000);
INSERT INTO BANK_ACCOUNT VALUES ( 2,'SBI_RESIDENCYROAD', 5000);
INSERT INTO BANK_ACCOUNT VALUES ( 3,'SBI_SHIVAJIROAD', 6000);
INSERT INTO BANK_ACCOUNT VALUES ( 4,'SBI_PARLIAMENTROAD', 9000);
INSERT INTO BANK_ACCOUNT VALUES ( 5,'SBI_JANTARMANTAR', 8000);
INSERT INTO BANK_ACCOUNT VALUES ( 6,'SBI_SHIVAJIROAD', 4000);
INSERT INTO BANK_ACCOUNT VALUES ( 8,'SBI_RESIDENCYROAD', 4000);
INSERT INTO BANK_ACCOUNT VALUES ( 9,'SBI_PARLIAMENTROAD', 3000);
INSERT INTO BANK_ACCOUNT VALUES ( 10,'SBI_RESIDENCYROAD', 5000);
INSERT INTO BANK_ACCOUNT VALUES ( 11,'SBI_JANTARMANTAR', 2000);

INSERT INTO BANK_CUSTOMER VALUES ('AVINASH', 'BULL_TEMPLE_ROAD', 'BANGALORE');
INSERT INTO BANK_CUSTOMER VALUES ('DINESH', 'BANNERGATTA_ROAD', 'BANGALORE');
```

```
INSERT INTO BANK_CUSTOMER VALUES ('MOHAN', 'NATIONALCOLLEGE_ROAD', 'BANGALORE');
INSERT INTO BANK_CUSTOMER VALUES ('NIKHIL', 'AKBAR_ROAD', 'DELHI');
INSERT INTO BANK_CUSTOMER VALUES ('RAVI', 'PRITHVIRAJ_ROAD', 'DELHI');

INSERT INTO DEPOSITER VALUES('AVINASH', 1);
INSERT INTO DEPOSITER VALUES('DINESH', 2);
INSERT INTO DEPOSITER VALUES('NIKHIL', 4);
INSERT INTO DEPOSITER VALUES('RAVI', 5);
INSERT INTO DEPOSITER VALUES('AVINASH', 8);
INSERT INTO DEPOSITER VALUES('NIKHIL', 9);
INSERT INTO DEPOSITER VALUES('DINESH', 10);
INSERT INTO DEPOSITER VALUES('NIKHIL', 11);

INSERT INTO LOAN VALUES (1, 'SBI_CHAMRAJPET', 1000);
INSERT INTO LOAN VALUES (2, 'SBI_RESIDENCYROAD', 2000);
INSERT INTO LOAN VALUES (3, 'SBI_SHIVAJIROAD', 3000);
INSERT INTO LOAN VALUES (4, 'SBI_PARLIAMENTROAD', 4000);
INSERT INTO LOAN VALUES (5, 'SBI_JANTARMANTAR', 5000);


SELECT CUSTOMER_NAME, COUNT(CUSTOMER_NAME)
FROM DEPOSITER D, BANK_ACCOUNT B
WHERE D.ACCNO = B.ACCNO
AND B.BRANCH_NAME = 'SBI_RESIDENCYROAD'
GROUP BY CUSTOMER_NAME
HAVING COUNT(CUSTOMER_NAME) >= 2;


SELECT CUSTOMER_NAME
FROM DEPOSITER D, BANK_ACCOUNT BA, BRANCH B
WHERE BRANCH_CITY = 'DELHI'
;


DELETE FROM BANK_ACCOUNT
WHERE BRANCH_NAME IN (
        SELECT BRANCH_NAME
    FROM BRANCH
    WHERE BRANCH_CITY = 'BOMBAY'
);
SELECT * FROM BANK_ACCOUNT;
```

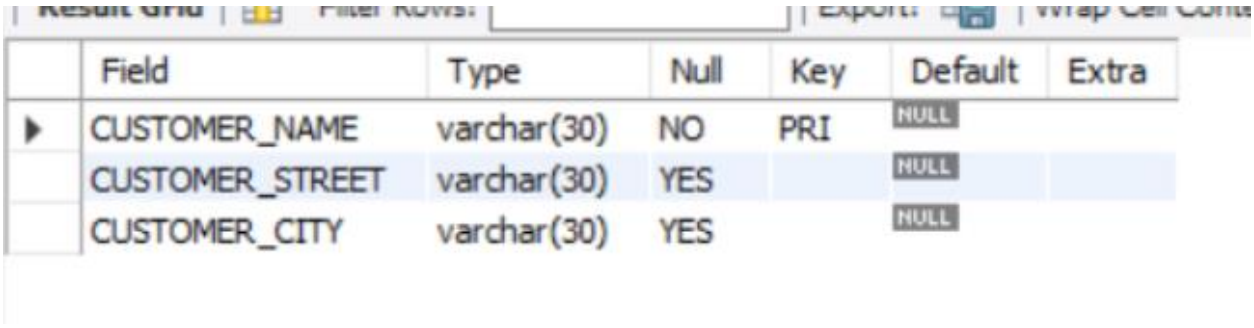
Outputs And Tables:-



Field	Type	Null	Key	Default	Extra
BRANCH_NAME	varchar(30)	NO	PRI	NULL	
BRANCH_CITY	varchar(30)	YES		NULL	
ASSETS	double	YES		NULL	



Field	Type	Null	Key	Default	Extra
ACCNO	int	NO	PRI	NULL	
BRANCH_NAME	varchar(30)	YES	MUL	NULL	
BALANCE	double	YES		NULL	



Field	Type	Null	Key	Default	Extra
CUSTOMER_NAME	varchar(30)	NO	PRI	NULL	
CUSTOMER_STREET	varchar(30)	YES		NULL	
CUSTOMER_CITY	varchar(30)	YES		NULL	

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
CUSTOMER_NAME	varchar(30)	NO	PRI	NULL	
ACCNO	int	NO	PRI	NULL	

Result Grid | Filter Rows: | Export: | Wrap Cell Co

Field	Type	Null	Key	Default	Extra
LOAN_NUMBER	int	NO	PRI	NULL	
BRANCH_NAME	varchar(30)	YES	MUL	NULL	
AMOUNT	double	YES		NULL	

```
65 •  SELECT * FROM BRANCH;
```

Result Grid | Filter Rows: | Edit:

BRANCH_NAME	BRANCH_CITY	ASSETS
SBI_CHAMRAJPET	BANGALORE	50000
SBI_JANTARMANTAR	DELHI	20000
SBI_PARLIAMENTROAD	DELHI	10000
SBI_RESIDENCYROAD	BANGALORE	10000
SBI_SHIVAJIROAD	BOMBAY	20000
NULL	NULL	NULL

```
65 •  SELECT * FROM BRANCH;
66 •  SELECT * FROM BANK_ACCOUNT;
```

Result Grid | Filter Rows:

ACCNO	BRANCH_NAME	BALANCE
1	SBI_CHAMRAJPET	2000
2	SBI_RESIDENCYROAD	5000
4	SBI_PARLIAMENTROAD	9000
5	SBI_JANTARMANTAR	8000
8	SBI_RESIDENCYROAD	4000
9	SBI_PARLIAMENTROAD	3000
10	SBI_RESIDENCYROAD	5000
11	SBI_JANTARMANTAR	2000
NULL	NULL	NULL

```
65 •  SELECT * FROM BRANCH;
66 •  SELECT * FROM BANK_ACCOUNT;
67 •  SELECT * FROM BANK_CUSTOMER;
```

Result Grid | Filter Rows: | Edit:

CUSTOMER_NAME	CUSTOMER_STREET	CUSTOMER_CITY
AVINASH	BULL_TEMPLE_ROAD	BANGALORE
DINESH	BANNERGATTA_ROAD	BANGALORE
MOHAN	NATIONALCOLLEGE_ROAD	BANGALORE
NIKHIL	AKBAR_ROAD	DELHI
RAVI	PRITHVIRAJ_ROAD	DELHI
NULL	NULL	NULL

```
65 •  SELECT * FROM BRANCH;
66 •  SELECT * FROM BANK_ACCOUNT;
67 •  SELECT * FROM BANK_CUSTOMER;
68 •  SELECT * FROM DEPOSITER;
```

esult Grid | Filter Rows:

CUSTOMER_NAME	ACCNO
AVINASH	1
DINESH	2
NIKHIL	4
RAVI	5
AVINASH	8
NIKHIL	9
DINESH	10
NIKHIL	11
NULL	NULL

```
69 •  SELECT * FROM LOAN;
```

Result Grid | Filter Rows:

CUSTOMER_NAME	ACCNO
AVINASH	1
DINESH	2
NIKHIL	4
RAVI	5
AVINASH	8
NIKHIL	9
DINESH	10
NIKHIL	11
NULL	NULL

```
45 •  SELECT CUSTOMER_NAME, COUNT(CUSTOMER_NAME)
46    FROM DEPOSITER D, BANK_ACCOUNT B
47    WHERE D.ACCNO = B.ACCNO
48    AND B.BRANCH_NAME = 'SBI_RESIDENCYROAD'
49    GROUP BY CUSTOMER_NAME
50    HAVING COUNT(CUSTOMER_NAME) >= 2;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

CUSTOMER_NAME	COUNT(CUSTOMER_NAME)
DINESH	2

```
57  DELETE FROM BANK_ACCOUNT
58  WHERE BRANCH_NAME IN (
59      SELECT BRANCH_NAME
60      FROM BRANCH
61      WHERE BRANCH_CITY = 'BOMBAY'
62  );
63  SELECT * FROM BANK_ACCOUNT;
```

Result Grid | Filter Rows:

ACCNO	BRANCH_NAME	BALANCE
1	SBI_CHAMRAJPET	2000
2	SBI_RESIDENCYROAD	5000
4	SBI_PARLIAMENTROAD	9000
5	SBI_JANTARMANTAR	8000
8	SBI_RESIDENCYROAD	4000
9	SBI_PARLIAMENTROAD	3000
10	SBI_RESIDENCYROAD	5000
11	SBI_JANTARMANTAR	2000
NULL	NULL	NULL

Lab Program 3:- Supplier Database

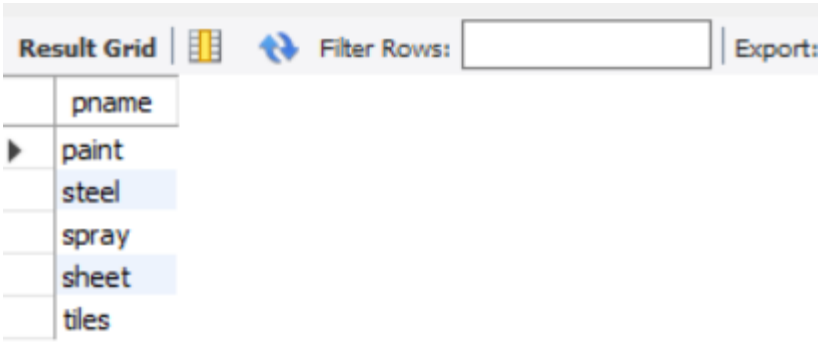
```
create
database
supplier;
```

```
use supplier;
create table suppliers(
       sid int primary key,
    sname varchar(30),
    address varchar(30)
);
create table parts(
       pid int primary key,
    pname varchar(30),
    color varchar(30)
);
create table catalog (
       sid int ,
    pid int ,
    cost real,
    constraint c_sid foreign key(sid) references suppliers(sid) ,
    constraint c_pid foreign key(pid) references parts(pid)
);
select * from suppliers;
select * from parts;
select * from catalog;

insert into suppliers values(1,'Acme Widget','kolkata') ;
insert into suppliers values(2,'Tata','bengaluru') ;
insert into suppliers values(3,'Reebok','delhi') ;
insert into suppliers values(4,'Nike','delhi') ;
insert into suppliers values(5,'Reliance','delhi') ;


insert into parts values(1,'paint','red') ;
insert into parts values(2,'steel','black') ;
insert into parts values(3,'spray','red') ;
insert into parts values(4,'sheet','green');
insert into parts values(5,'tiles','blue');
delete from parts where pid=5;
```

```
insert into catalog values(1,1,100);
insert into catalog values(1,2,200);
insert into catalog values(1,3,200);
insert into catalog values(1,4,100);
insert into catalog values(2,1,300);
insert into catalog values(2,2,100);
insert into catalog values(3,2,90);
insert into catalog values(3,3,110);
insert into catalog values(3,4,110);
insert into catalog values(4,1,100);
insert into catalog values(4,3,120);
insert into catalog values(4,4,130);

select * from suppliers;
select * from catalog;
select * from parts;



insert into parts values(5,'tiles','blue');
select p.pname from parts p where p.pid in (select pid from catalog c group by c.pid
having count(c.sid)>0);
insert into catalog values(1,5,140);
select p.pname from parts p where p.pid in (select pid from catalog c group by c.pid
having count(c.sid)>0);
delete from catalog where pid=5;
delete from parts where pid=5;
select * from catalog;
select * from parts;


select s.sname from suppliers s where s.sid in (select c.sid from catalog c group by c.sid
having count(distinct (c.pid))=(select count(p.pid) from parts p));



select s.sname from suppliers s where s.sid in (select ca.sid  from catalog ca,parts p
where ca.pid=p.pid and p.color='red' group by ca.sid having count(ca.pid)=(select count(*)
from parts p where p.color='red'));


select ca.pid from catalog ca  where ca.sid=(select s.sid from suppliers s where s.sname
='Acme Widget') having (select count(c.pid) from catalog c where c.pid=ca.pid)=1;
```

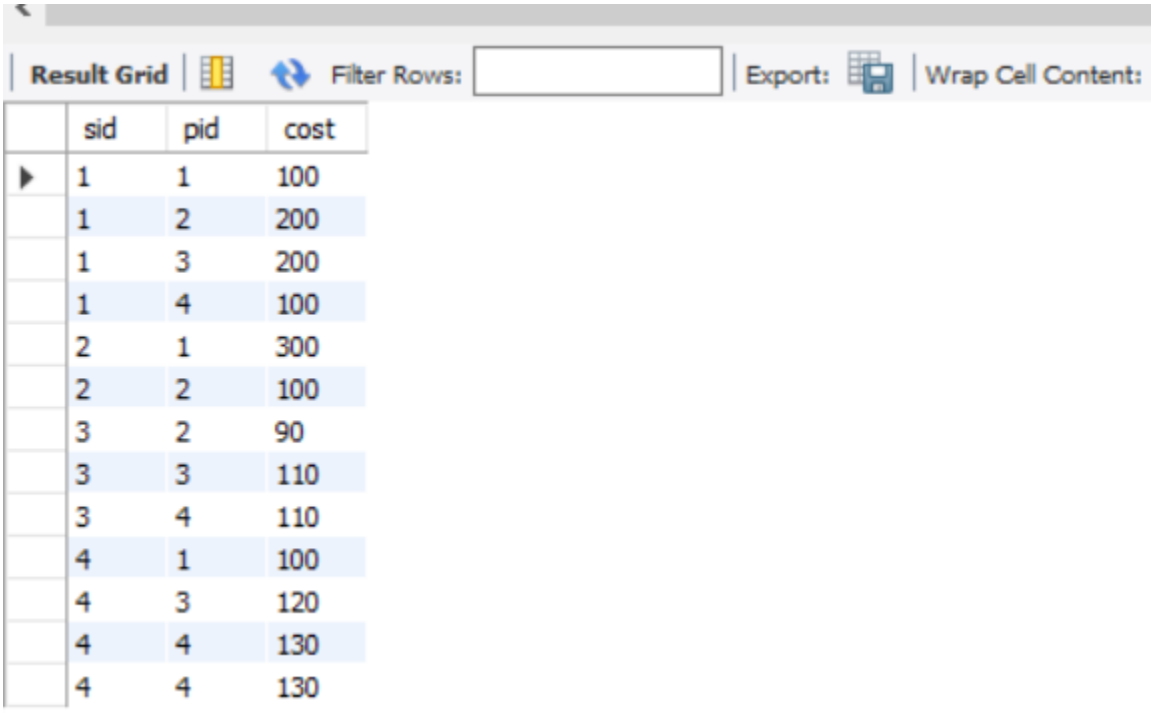
```
select distinct c.sid,c.pid from catalog c where c.cost > (select avg(ca.cost) from
catalog ca where ca.pid=c.pid);


select s.sname from suppliers s where s.sid in (select c.sid from catalog c where
c.cost=(select max(cost) from catalog ca where ca.pid=c.pid));


select s.sname from suppliers s where s.sid in(select c.sid from catalog c where c.sid not
in (select distinct(ca.sid) from catalog ca,parts p where ca.pid=p.pid and
p.color!='red'));
insert into catalog values(5,1,140);
select s.sname from suppliers s where s.sid in(select c.sid from catalog c where c.sid not
in (select distinct(ca.sid) from catalog ca,parts p where ca.pid=p.pid and
p.color!='red'));
delete from catalog where sid=5;
select * from catalog;
```

Outputs And Tables:



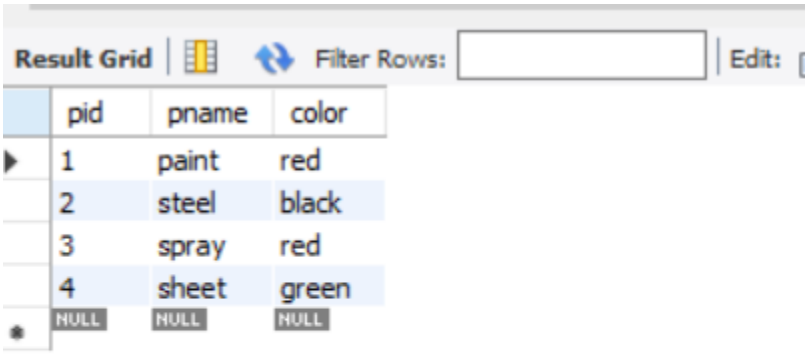
Result Grid | Filter Rows: | Export:

pname
paint
steel
spray
sheet
tiles

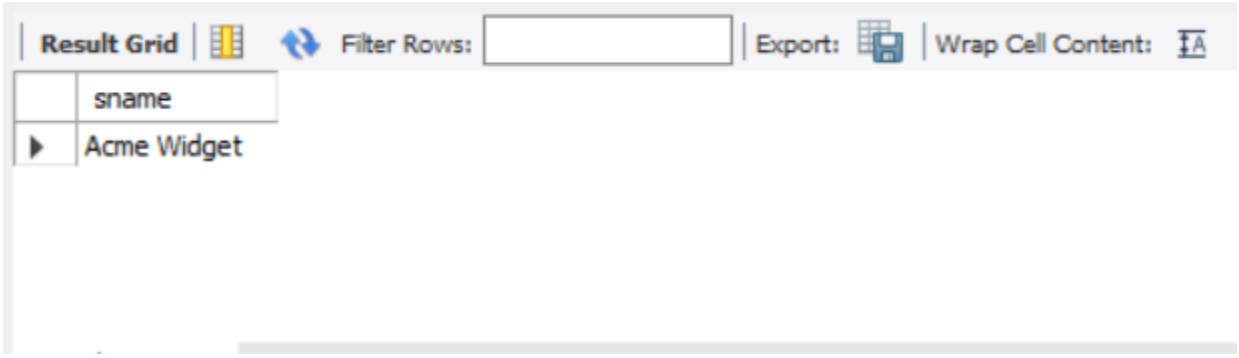


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

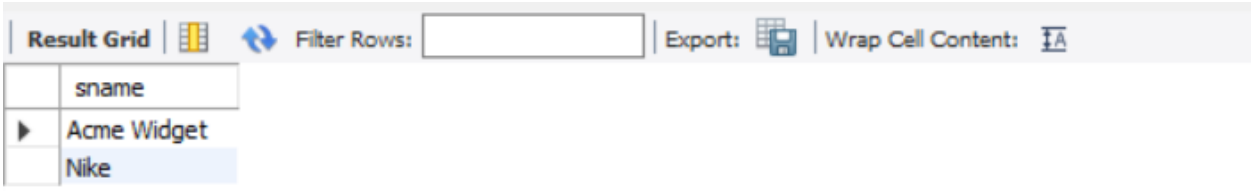
sid	pid	cost
1	1	100
1	2	200
1	3	200
1	4	100
2	1	300
2	2	100
3	2	90
3	3	110
3	4	110
4	1	100
4	3	120
4	4	130
4	4	130



pid	pname	color
1	paint	red
2	steel	black
3	spray	red
4	sheet	green
NULL	NULL	NULL



sname
Acme Widget



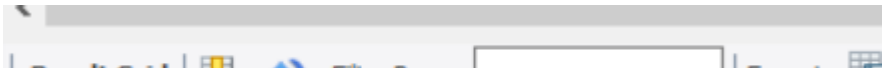
sname
Acme Widget
Nike

pid

sid	pid
1	2
1	3
2	1
4	4

4

sname
Acme Widget
Tata
Nike



sname

Result Grid | Filter Rows:

sname
Reliance

Result Grid | Filter Rows: | Export:

sid	pid	cost
1	1	100
1	2	200
1	3	200
1	4	100
2	1	300
2	2	100
3	2	90
3	3	110
3	4	110
4	1	100
4	3	120
4	4	130
4	4	130

Lab program 4: Student Faculty Database

```
CREATE DATABASE
student_faculty;
```

```
USE student_faculty;
CREATE TABLE student(
       snum INT,
       sname VARCHAR(10),
       major VARCHAR(2),
       lvl VARCHAR(2),
       age INT, primary key(snum));

CREATE TABLE faculty(
       fid INT,fname VARCHAR(20),
       deptid INT,
    PRIMARY KEY(fid));

CREATE TABLE class(
       cname VARCHAR(20),
       metts_at TIMESTAMP,
       room VARCHAR(10),
    fid INT,
       PRIMARY KEY(cname),
       FOREIGN KEY(fid) REFERENCES faculty(fid));

CREATE TABLE enrolled(
       snum INT,
       cname VARCHAR(20),
       PRIMARY KEY(snum,cname),
       FOREIGN KEY(snum) REFERENCES student(snum),
       FOREIGN KEY(cname) REFERENCES class(cname));
    use student_faculty;
    show tables;

INSERT INTO STUDENT VALUES(1, 'jhon', 'CS', 'Sr', 19);
INSERT INTO STUDENT VALUES(2, 'Smith', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(3 , 'Jacob', 'CV', 'Sr', 20);
INSERT INTO STUDENT VALUES(4, 'Tom ', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(5, 'Rahul', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(6, 'Rita', 'CS', 'Sr', 21);
```

```
INSERT INTO FACULTY VALUES(11, 'Harish', 1000);
INSERT INTO FACULTY VALUES(12, 'MV', 1000);
INSERT INTO FACULTY VALUES(13 , 'Mira', 1001);
INSERT INTO FACULTY VALUES(14, 'Shiva', 1002);
INSERT INTO FACULTY VALUES(15, 'Nupur', 1000);

insert into class values('class1', '12/11/15 10:15:16', 'R1', 14);
insert into class values('class10', '12/11/15 10:15:16', 'R128', 14);
insert into class values('class2', '12/11/15 10:15:20', 'R2', 12);
insert into class values('class3', '12/11/15 10:15:25', 'R3', 11);
insert into class values('class4', '12/11/15 20:15:20', 'R4', 14);
insert into class values('class5', '12/11/15 20:15:20', 'R3', 15);
insert into class values('class6', '12/11/15 13:20:20', 'R2', 14);
insert into class values('class7', '12/11/15 10:10:10', 'R3', 14);

insert into enrolled values(1, 'class1');
insert into enrolled values(2, 'class1');
insert into enrolled values(3, 'class3');
insert into enrolled values(4, 'class3');
insert into enrolled values(5, 'class4');
insert into enrolled values(1, 'class5');
insert into enrolled values(2, 'class5');
insert into enrolled values(3, 'class5');
insert into enrolled values(4, 'class5');
insert into enrolled values(5, 'class5');
select * from student;
select * from faculty;
select * from class;
select * from enrolled;

-- Query 1
SELECT DISTINCT S.Sname
FROM Student S, Class C, Enrolled E, Faculty F
WHERE S.snum = E.snum AND E.cname = C.cname AND C.fid = F.fid AND
F.fname = 'Harish' AND S.lvl = 'Jr';

-- Query 2
SELECT DISTINCT cname
       FROM class
       WHERE room='R128'
       OR
       cname IN (SELECT e.cname FROM enrolled e GROUP BY e.cname HAVING
COUNT(*)>=5);
```

```
-- Query 3
SELECT DISTINCT S.sname
FROM Student S
WHERE S.snum IN (SELECT E1.snum
                     FROM Enrolled E1, Enrolled E2, Class C1, Class C2
                     WHERE E1.snum = E2.snum AND E1.cname <> E2.cname
                     AND E1.cname = C1.cname
                     AND E2.cname = C2.cname AND C1.metts_at = C2.metts_at);


-- Query 4
SELECT f.fname,f.fid
                     FROM faculty f
              WHERE f.fid in ( SELECT fid FROM class
                     GROUP BY fid HAVING COUNT(*)=(SELECT COUNT(DISTINCT room)
FROM class) );

-- Query 5
SELECT DISTINCT F.fname
FROM Faculty F
WHERE 5 > (SELECT COUNT(E.snum)
FROM Class C, Enrolled E
WHERE C.cname = E.cname
AND C.fid = F.fid);

-- Query 6
SELECT DISTINCT S.sname
FROM Student S
WHERE S.snum NOT IN (SELECT E.snum
FROM Enrolled E );

-- Query 7
SELECT S.age, S.lvl
FROM STUDENT S
GROUP BY S.age, S.lvl
HAVING S.lvl IN(SELECT S1.lvl
       FROM STUDENT S1
       WHERE S1.age=S.age
       GROUP BY S1.age, S1.lvl
```

```
                        HAVING COUNT(*)  >= ALL (SELECT COUNT(*)
                                FROM STUDENT S2
                                WHERE S1.age=S2.age
                                GROUP BY S2.lvl, S2.age))
                ORDER BY S.age;
```

Output and Tables :

Tables_in_student_faculty
class
enrolled
faculty
student

snum	sname	major	lvl	age
1	jhon	CS	Sr	19
2	Smith	CS	Jr	20
3	Jacob	CV	Sr	20
4	Tom	CS	Jr	20
5	Rahul	CS	Jr	20
6	Rita	CS	Sr	21
NULL	NULL	NULL	NULL	NULL

fid	fname	deptid
11	Harish	1000
12	MV	1000
13	Mira	1001
14	Shiva	1002
15	Nupur	1000
NULL	NULL	NULL

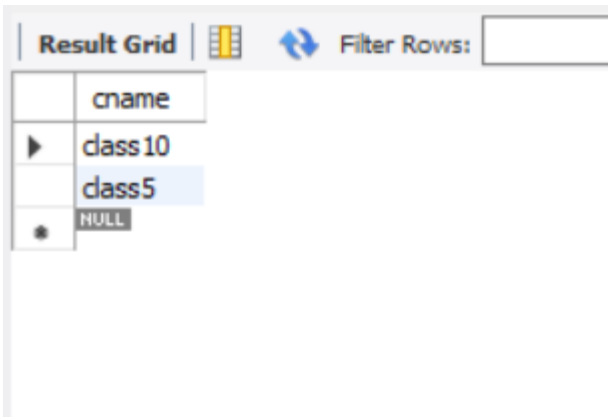
Result Grid | Filter Rows: | Edit:

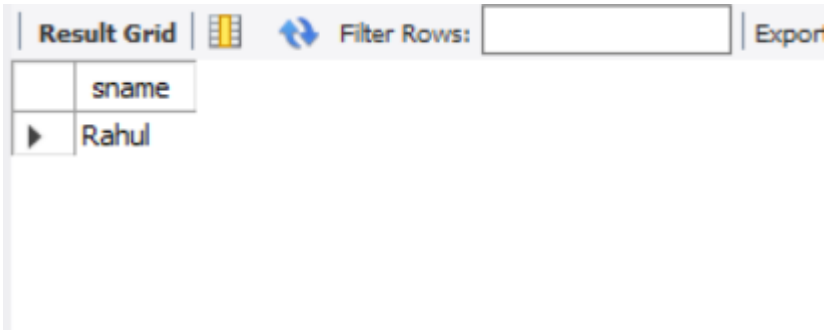
cname	metts_at	room	fid
class1	2012-11-15 10:15:16	R1	14
class10	2012-11-15 10:15:16	R128	14
class2	2012-11-15 10:15:20	R2	12
class3	2012-11-15 10:15:25	R3	11
class4	2012-11-15 20:15:20	R4	14
class5	2012-11-15 20:15:20	R3	15
class6	2012-11-15 13:20:20	R2	14
class7	2012-11-15 10:10:10	R3	14
NULL	NULL	NULL	NULL

Result Grid | Filter Rows:

snum	cname
1	class1
2	class1
3	class3
4	class3
5	class4
1	class5
2	class5
3	class5
4	class5
5	class5
NULL	NULL


Result Grid
Filter Rows:
Export:
Wrap Cell Content:
Sname
Tom


Result Grid
Filter Rows:
cname
class 10
class5
NULL


Result Grid
Filter Rows:
Expor
sname
Rahul

Result Grid | Filter Rows:

fname	fid
Shiva	14
NULL	NULL

Result Grid | Filter Rows:

fname
Harish
MV
Mira
Shiva

Result Grid | Filter Rows:

sname
Rita

Result Grid | Filter Rows:

age	lvl
19	Sr
20	Jr
21	Sr

Lab Program 5:- Airline Flight Database

```
create
database
flightdb;
use flightdb;

create table flights(
        flno int,
    fromplace varchar(15),
    toplace varchar(15),
    distance int,
    departs datetime,
    arrives datetime,
    price int,
    primary key (flno)
);
desc flights;
create table aircraft(
        aid int,
    aname varchar(15),
    cruisingrange int,
    primary key (aid)
);
desc aircraft;
create table employees (
        eid int,
    ename varchar(15),
    salary int,
    primary key (eid)
);
desc employees;
create table certified (
        eid int,
    aid int,
    foreign key (eid) references employees(eid),
    foreign key (aid) references aircraft(aid)
);
desc certified;
insert into flights values(101, 'Bangalore', 'Delhi', 2500, '2005-05-13 07:15:31', '2005-
05-13 18:15:31', 5000);
```

```
insert into flights values(102, 'Bangalore', 'Lucknow', 3000, '2013-05-05 07:15:31',
'2013-05-05 11:15:31', 6000);
insert into flights values(103, 'Lucknow', 'Delhi', 500, '2013-05-05 12:15:31', '2013-05-
05 17:15:31', 3000);
insert into flights values(107, 'Bangalore', 'Frankfurt', 8000, '2013-05-05 07:15:31',
'2013-05-05 22:15:31', 60000);
insert into flights values(104, 'Bangalore', 'Frankfurt', 8500, '2013-05-05 07:15:31',
'2013-05-05 23:15:31', 75000);
insert into flights values(105, 'Kolkata', 'Delhi', 3400, '2013-05-05 07:15:31', '2013-05-
05 09:15:31', 7000);
insert into flights values(106, 'Bangalore', 'Kolkata', 1000, '2013-05-05 01:15:30',
'2013-05-05 09:20:30', 10000);
insert into flights values(108, 'Lucknow', 'Kolkata', 1000, '2013-05-05 11:30:30', '2013-
05-05 15:20:30', 10000);

commit;

select * from flights;

insert into aircraft values(101, '747', 3000);
insert into aircraft values(102, 'Boeing', 900);
insert into aircraft values(103, '647', 800);
insert into aircraft values(104, 'Dreamliner', 10000);
insert into aircraft values(105, 'Boeing', 3500);
insert into aircraft values(106, '707', 1500);
insert into aircraft values(107, 'Dream', 120000);
insert into aircraft values(108, '707', 760);
insert into aircraft values(109, '747', 1000);
commit;

select * from aircraft;

insert into employees values(701, 'A', 50000);
insert into employees values(702, 'B', 100000);
insert into employees values(703, 'C', 150000);
insert into employees values(704, 'D', 90000);
insert into employees values(705, 'E', 40000);
insert into employees values(706, 'F', 60000);
insert into employees values(707, 'G', 90000);
commit;

select * from employees;
```

```
insert into certified values(701, 101);
insert into certified values(701, 102);
insert into certified values(701, 106);
insert into certified values(701, 105);

insert into certified values(702, 104);
insert into certified values(703, 104);
insert into certified values(704, 104);

insert into certified values(702, 107);
insert into certified values(703, 107);
insert into certified values(704, 107);

insert into certified values(702, 101);
insert into certified values(702, 108);
insert into certified values(701, 109);
commit;
select * from certified;

-- Query 1
select distinct a.aname from aircraft a where a.aid in (
        select c.aid from certified c, employees e where
    c.eid = e.eid and not exists(
                select * from employees e1 where e1.eid=e.eid and e1.salary<80000

    )
);

-- Query 2
select max(a.cruisingrange), c.eid from certified c, aircraft a where c.aid = a.aid group
by c.eid having count(c.eid)>3;

-- Query 3
select ename from employees where salary <(
select min(price) from flights where fromplace='Bangalore' and toplace='Frankfurt');



-- Query 4

select avg(e.salary), c.aid from certified c, employees e where c.aid in(
```

```
select aid from aircraft where cruisingrange>1000) and e.eid = c.eid group by c.aid;



-- Query 5
select ename from employees where eid in(
select eid from certified where aid in(
select aid from aircraft where aname = 'Boeing'));

-- Query 6

select aname from aircraft where cruisingrange > any (select distance from flights where
fromplace='Bangalore' and toplace='Delhi');



-- Query 7
SELECT F.flno, F.departs
FROM flights F
WHERE F.flno IN ( ( SELECT F0.flno
 FROM flights F0
 WHERE F0.fromplace = 'Bangalore' AND F0.toplace = 'Kolkata'
 AND extract(hour from F0.arrives) < 18 )
 UNION
( SELECT F0.flno
 FROM flights F0, flights F1
 WHERE F0.fromplace = 'Bangalore' AND F0.toplace <> 'Kolkata'
 AND F0.toplace = F1.fromplace AND F1.toplace = 'Kolkata'
 AND F1.departs > F0.arrives
 AND extract(hour from F1.arrives) < 18)
 UNION
( SELECT F0.flno
 FROM flights F0, flights F1, flights F2
 WHERE F0.fromplace = 'Bangalore'
 AND F0.toplace = F1.fromplace
 AND F1.toplace = F2.fromplace
 AND F2.toplace = 'Kolkata'
 AND F0.toplace <> 'Kolkata'
 AND F1.toplace <> 'Kolkata'
 AND F1.departs > F0.arrives
 AND F2.departs > F1.arrives
 AND extract(hour from F2.arrives) < 18));
```

Outputs and Tables :

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
flno	int	NO	PRI	NULL	
fromplace	varchar(15)	YES		NULL	
toplace	varchar(15)	YES		NULL	
distance	int	YES		NULL	
departs	datetime	YES		NULL	
arrives	datetime	YES		NULL	
price	int	YES		NULL	

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

flno	fromplace	toplace	distance	departs	arrives	price
101	Bangalore	Delhi	2500	2005-05-13 07:15:31	2005-05-13 18:15:31	5000
102	Bangalore	Lucknow	3000	2013-05-05 07:15:31	2013-05-05 11:15:31	6000
103	Lucknow	Delhi	500	2013-05-05 12:15:31	2013-05-05 17:15:31	3000
104	Bangalore	Frankfurt	8500	2013-05-05 07:15:31	2013-05-05 23:15:31	75000
105	Kolkata	Delhi	3400	2013-05-05 07:15:31	2013-05-05 09:15:31	7000
106	Bangalore	Kolkata	1000	2013-05-05 01:15:30	2013-05-05 09:20:30	10000
107	Bangalore	Frankfurt	8000	2013-05-05 07:15:31	2013-05-05 22:15:31	60000
108	Lucknow	Kolkata	1000	2013-05-05 11:30:30	2013-05-05 15:20:30	10000
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content

Field	Type	Null	Key	Default	Extra
aid	int	NO	PRI	NULL	
aname	varchar(15)	YES		NULL	
cruisingrange	int	YES		NULL	

Result Grid | Filter Rows: | Edit:

aid	aname	cruisingrange
101	747	3000
102	Boeing	900
103	647	800
104	Dreamliner	10000
105	Boeing	3500
106	707	1500
107	Dream	120000
108	707	760
109	747	1000
NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
eid	int	NO	PRI	NULL	
ename	varchar(15)	YES		NULL	
salary	int	YES		NULL	

Result Grid | Filter Rows: | Edit:

eid	ename	salary
701	A	50000
702	B	100000
703	C	150000
704	D	90000
705	E	40000
706	F	60000
707	G	90000
NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

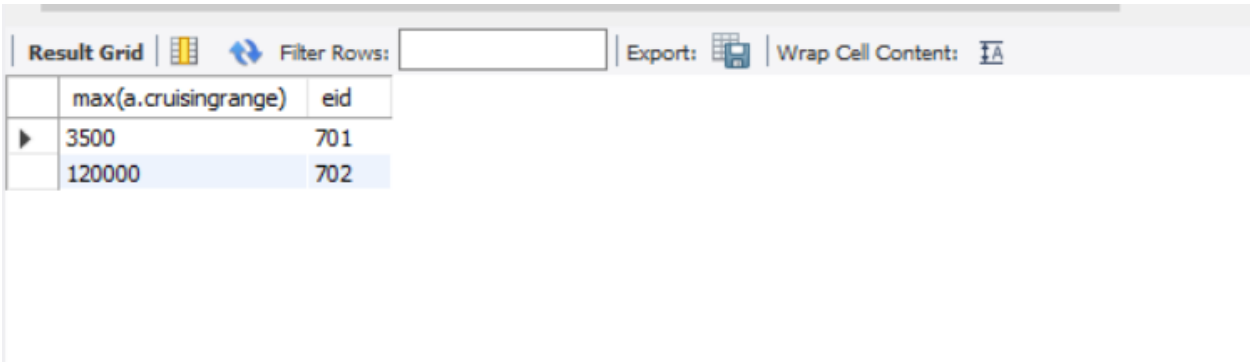
	Field	Type	Null	Key	Default	Extra
▶	eid	int	YES	MUL	NULL	
	aid	int	YES	MUL	NULL	

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

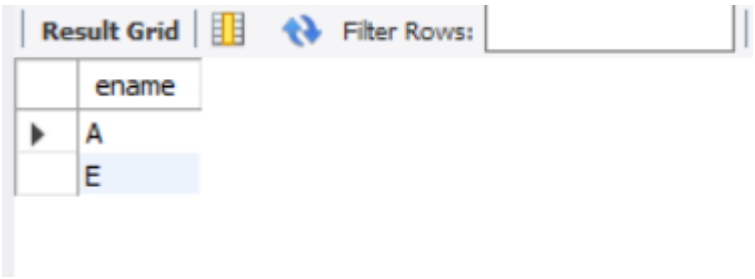
	eid	aid
▶	701	101
	701	102
	701	106
	701	105
	702	104
	703	104
	704	104
	702	107
	703	107
	704	107
	702	101
	702	108
	701	109

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

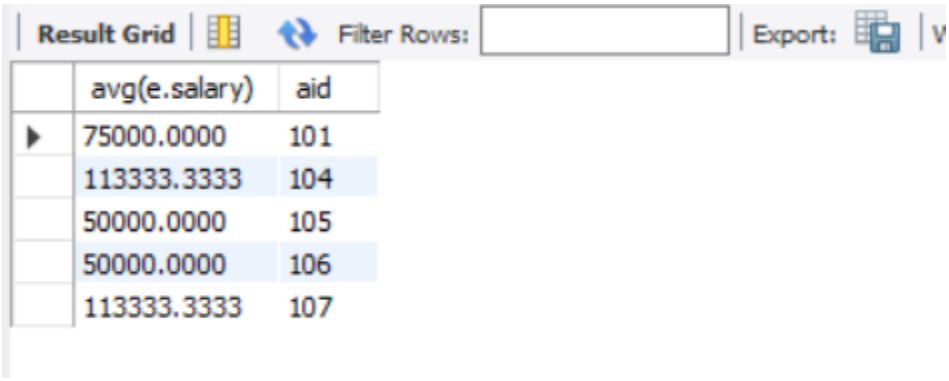
	aname
▶	747
	Dreamliner
	Dream
	707


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

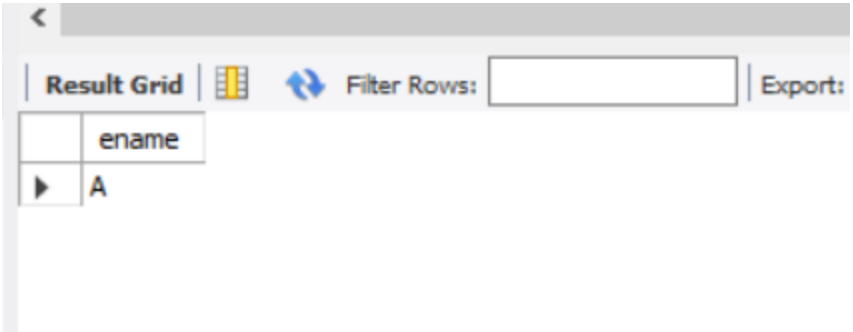
	max(a.cruisingrange)	eid
▶	3500	701
	120000	702


Result Grid | Filter Rows:

	ename
▶	A
	E


Result Grid | Filter Rows: | Export:

	avg(e.salary)	aid
▶	75000.0000	101
	113333.3333	104
	50000.0000	105
	50000.0000	106
	113333.3333	107


Result Grid | Filter Rows: | Export:

	ename
▶	A

Result Grid | Filter Rows: | Export:

aname
747
Dreamliner
Boeing
Dream

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

flno	departs
102	2013-05-05 07:15:31
106	2013-05-05 01:15:30

Lab program 6: ORDER PROCESSING DATABASE

```
create database
order_processsing;
use order_processsing;
create table salesman (salesman_id int, name varchar
(20), city varchar (20), commission varchar (20), primary
key (salesman_id));
create table customer (customer_id int, cust_name varchar
(20), city varchar (20), grade int, salesman_id int,
primary key (customer_id), foreign key(salesman_id)
references salesman(salesman_id)
on delete set null);
create table orders (ord_no int, purchase_amt real,
ord_date date, customer_id int, salesman_id int, primary
key (ord_no), foreign key (customer_id) references
customer(customer_id)
 on delete cascade, foreign key(salesman_id) references
salesman(salesman_id) on delete cascade);

insert into salesman values (1000, 'john','bangalore','25
%');
insert into salesman values (2000, 'ravi','bangalore','20
%');
insert into salesman values (3000, 'kumar','mysore','15
%');
insert into salesman values (4000, 'smith','delhi','30
%');
insert into salesman values (5000,
'harsha','hydrabad','15 %');

insert into customer values (10, 'preethi','bangalore',
100, 1000);
insert into customer values (11, 'vivek','mangalore',
300, 1000);
insert into customer values (12, 'bhaskar','chennai',
400, 2000);
insert into customer values (13, 'chethan','bangalore',
200, 2000);
insert into customer values (14, 'mamatha','bangalore',
400, 3000);
```

```
insert into orders values (50, 5000, '04-05-17', 10,
1000);
insert into orders values (51, 450, '20-01-17', 10,
2000);
insert into orders values (52, 1000, '24-02-17', 13,
2000);
insert into orders values (53, 3500, '13-04-17', 14,
3000);
insert into orders values (54, 550, '09-03-17', 12,
2000);

select * from salesman;
select * from customer;
select * from orders;
```

```
select grade, count(distinct customer_id) from customer
group by grade having grade > (select avg(grade) from
customer where city='bangalore');

select salesman_id, name from salesman a where 1 <
(select count(*) from customer where
salesman_id=a.salesman_id);

select salesman.salesman_id, name, cust_name, commission
from salesman, customer where salesman.city =
customer.city union
select salesman_id, name, 'no match', commission from
salesman where not city = any (select city from
customer);

create view salesman_view as select b.ord_date,
a.salesman_id, a.name from salesman a, orders b where
a.salesman_id = b.salesman_id and b.purchase_amt=(select
max(purchase_amt) from orders c where c.ord_date =
b.ord_date);
select * from salesman_view;

delete from salesman where salesman_id=1000;
select * from salesman;
```

Outputs: -

Result Grid | Filter Rows: | Edit: | Export/Impo

customer_id	cust_name	city	grade	salesman_id
10	preethi	bangalore	100	1000
11	vivek	mangalore	300	1000
12	bhaskar	chennai	400	2000
13	chethan	bangalore	200	2000
14	mamatha	bangalore	400	3000
NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Ce

ord_no	purchase_amt	ord_date	customer_id	salesman_id
50	5000	2004-05-17	10	1000
51	450	2020-01-17	10	2000
52	1000	2024-02-17	13	2000
53	3500	2013-04-17	14	3000
54	550	2009-03-17	12	2000
NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Expo

salesman_id	name	city	commission
1000	john	bangalore	25 %
2000	ravi	bangalore	20 %
3000	kumar	mysore	15 %
4000	smith	delhi	30 %
5000	harsha	hydrabad	15 %
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

grade	count(distinct customer_id)
300	1
400	2

Result Grid | Filter Rows: | Edit: | Export/Import:

salesman_id	name
1000	john
2000	ravi
NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

salesman_id	name	cust_name	commission
1000	john	preethi	25 %
2000	ravi	preethi	20 %
1000	john	chethan	25 %
2000	ravi	chethan	20 %
1000	john	mamatha	25 %
2000	ravi	mamatha	20 %
3000	kumar	no match	15 %
4000	smith	no match	30 %
5000	harsha	no match	15 %

Result 6

Result Grid | Filter Rows: | Export: | Wrap Cell Co

ord_date	salesman_id	name
2004-05-17	1000	john
2020-01-17	2000	ravi
2024-02-17	2000	ravi
2013-04-17	3000	kumar
2009-03-17	2000	ravi

Result Grid | Filter Rows: | Edit: | Export/Import:

salesman_id	name	city	commission
2000	ravi	bangalore	20 %
3000	kumar	mysore	15 %
4000	smith	delhi	30 %
5000	harsha	hydrabad	15 %
NULL	NULL	NULL	NULL

Lab Program 7: - Book Dealer Database

```
create database book_dealer;
```

```
use book_dealer;
create table publisher (name varchar (20) primary key, phone long, address varchar (20));
create table book (book_id int primary key, title varchar (20), pub_year varchar (20), publisher_name varchar(20) , foreign key(publisher_name) references publisher(name) on delete cascade);
```

```
create table book_authors (author_name varchar (20), book_id int, foreign key(book_id) references book(book_id) on delete cascade, primary key (book_id, author_name));
create table library_branch (branch_id int primary key, branch_name varchar (50), address varchar (50));
create table book_copies (no_of_copies integer, book_id int, branch_id int, primary key (book_id, branch_id), foreign key(book_id) references book(book_id) on delete cascade, foreign key(branch_id) references library_branch(branch_id) on delete cascade);
create table card(card_no int primary key);
create table book_lending (date_out date, due_date date, book_id int, branch_id int, card_no int, primary key (book_id, branch_id, card_no), foreign key(book_id) references book(book_id) on delete cascade,
foreign key(branch_id) references library_branch(branch_id) on delete cascade, foreign key(card_no) references card(card_no) on delete cascade );

insert into publisher values ('mcgraw-hill', 9989076587, 'bangalore');
insert into publisher values ('pearson', 9889076565, 'newdelhi');
insert into publisher values ('random house', 7455679345, 'hydrabad');
insert into publisher values ('hachette livre', 8970862340, 'chenai');
insert into publisher values ('grupo planeta', 7756120238, 'bangalore');


insert into book values (1,'dbms','jan-2017', 'mcgraw-hill');
insert into book values (2,'adbms','jun-2016', 'mcgraw-hill');
insert into book values (3,'cn','sep-2016', 'pearson');
insert into book values (4,'cg','sep-2015', 'grupo planeta');
insert into book values (5,'os','may-2016', 'pearson');

insert into book_authors values ('navathe', 1);
insert into book_authors values ('navathe', 2);
insert into book_authors values ('tanenbaum', 3);
insert into book_authors values ('edward angel', 4);
insert into book_authors values ('galvin', 5);
```

```
insert into library_branch values (10,'rr nagar','bangalore');
insert into library_branch values (11,'rnsit','bangalore');
insert into library_branch values (12,'rajaji nagar',
'bangalore');
insert into library_branch values (13,'nitte','mangalore');
insert into library_branch values (14,'manipal','udupi');

insert into book_copies values (10, 1, 10);
insert into book_copies values (5, 1, 11);
insert into book_copies values (2, 2, 12);
insert into book_copies values (5, 2, 13);
insert into book_copies values (7, 3, 14);
insert into book_copies values (1, 5, 10);
insert into book_copies values (3, 4, 11);

insert into card values (100);
insert into card values (101);
insert into card values (102);
insert into card values (103);
insert into card values (104);

insert into book_lending values ('17-01-01','17-06-01', 1, 10,
101);
insert into book_lending values ('17-01-11','17-03-11', 3, 14,
101);
insert into book_lending values ('17-02-21','17-04-21', 2, 13,
101);
insert into book_lending values ('17-03-15','17-07-15', 4, 11,
101);
insert into book_lending values ('17-04-12','17-05-12', 1, 11,
104);

select * from publisher;
select * from book;
select * from book_authors;
select * from library_branch;
select * from book_copies;
select * from card;
select * from book_lending;

select card_no from book_lending where date_out between '17-01-
01' and '17-07-01' group by card_no having count(*) > 3;
```

```
delete from book where book_id=3;
select * from book;

create view view_publication as select pub_year from book;
select * from view_publication;

create view view_books as select b.book_id, b.title,
c.no_of_copies from book b, book_copies c, library_branch l
where b.book_id=c.book_id and c.branch_id=l.branch_id;
select * from view_books;
```

Outputs: -

Result Grid | Filter Rows: | Export:

card_no
101

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

author_name	book_id
navathe	1
navathe	2
tanenbaum	3
edward angel	4
galvin	5
NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

no_of_copies	book_id	branch_id
10	1	10
5	1	11
2	2	12
5	2	13
7	3	14
3	4	11
1	5	10
NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

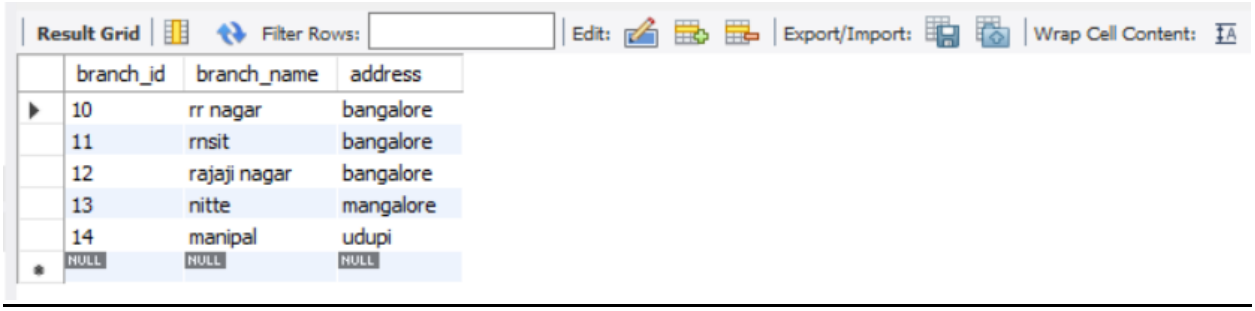
date_out	due_date	book_id	branch_id	card_no
2017-01-01	2017-06-01	1	10	101
2017-04-12	2017-05-12	1	11	104
2017-02-21	2017-04-21	2	13	101
2017-01-11	2017-03-11	3	14	101
2017-03-15	2017-07-15	4	11	101
NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	pub_year	publisher_name
1	dbms	jan-2017	mcgraw-hill
2	adbms	jun-2016	mcgraw-hill
3	cn	sep-2016	pearson
4	cg	sep-2015	grupo planeta
5	os	may-2016	pearson
NULL	NULL	NULL	NULL

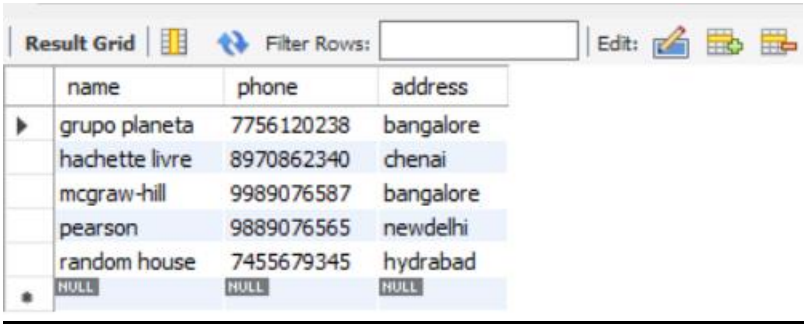
Result Grid | Filter Rows: | Edit:

card_no
100
101
102
103
104
NULL



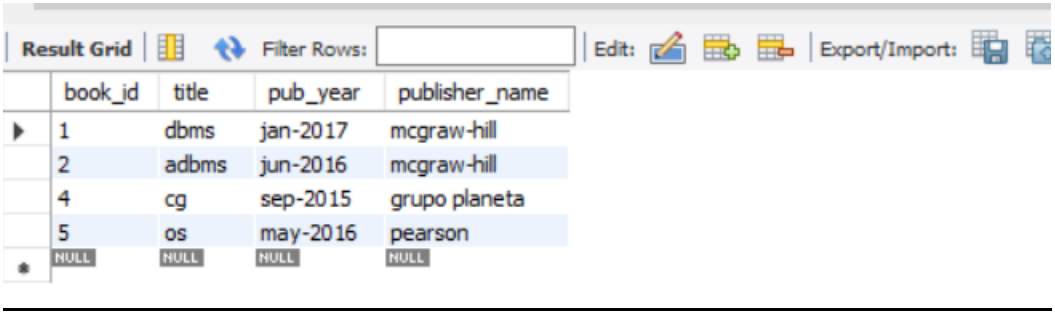
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

branch_id	branch_name	address
10	rr nagar	bangalore
11	rnsit	bangalore
12	rajaji nagar	bangalore
13	nitte	mangalore
14	manipal	udupi
NULL	NULL	NULL



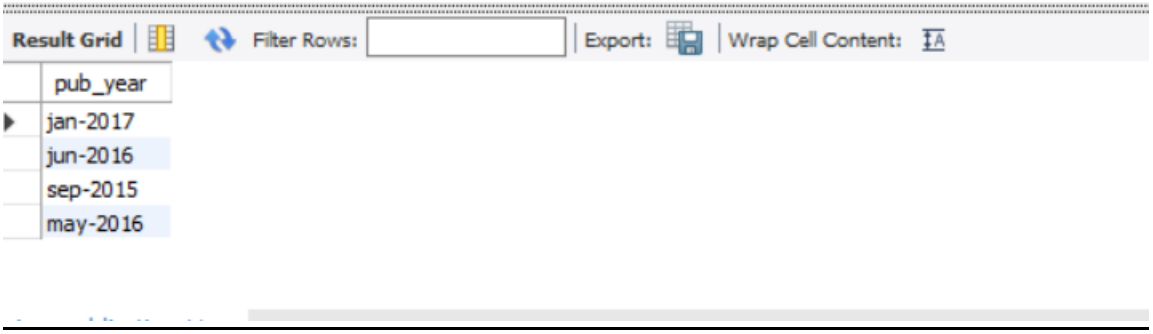
Result Grid | Filter Rows: | Edit:

name	phone	address
grupo planeta	7756120238	bangalore
hachette livre	8970862340	chenai
mcgraw-hill	9989076587	bangalore
pearson	9889076565	newdelhi
random house	7455679345	hydrabad
NULL	NULL	NULL



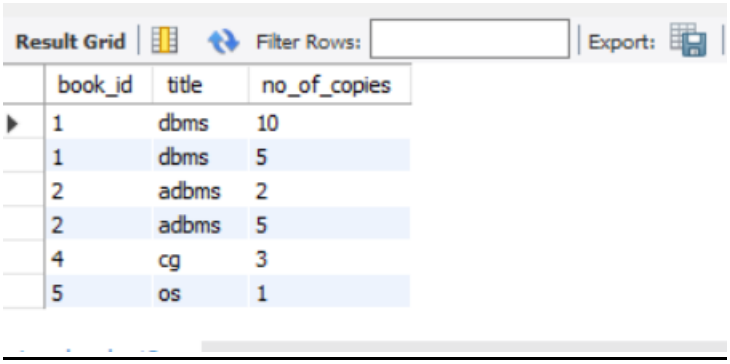
Result Grid | Filter Rows: | Edit: | Export/Import:

book_id	title	pub_year	publisher_name
1	dbms	jan-2017	mcgraw-hill
2	adbms	jun-2016	mcgraw-hill
4	cg	sep-2015	grupo planeta
5	os	may-2016	pearson
NULL	NULL	NULL	NULL



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

pub_year
jan-2017
jun-2016
sep-2015
may-2016



Result Grid | Filter Rows: | Export:

book_id	title	no_of_copies
1	dbms	10
1	dbms	5
2	adbms	2
2	adbms	5
4	cg	3
5	os	1

Lab Program8 : -Student Enrollment Database

```
create database
Student_Enrollment;
```

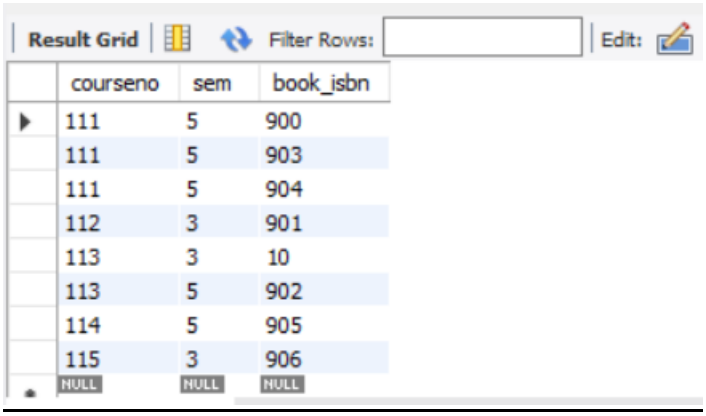
```
use Student_Enrollment;
create table student(regno varchar(30) primary key,
name varchar(30), major varchar(30), bdate date);

create table course(courseno int primary key, cname
varchar(30), dept varchar(30));
create table enroll(regno varchar(30), courseno int,
sem int, marks int,primary key(regno,courseno),
foreign key(regno) references student(regno),
foreign key(courseno) references course(courseno));

create table text(book_isbn int,book_title
varchar(20),publisher varchar(20),author
varchar(20),primary key (book_isbn));
create table book_adoption(courseno int,sem
int,book_isbn int,primary key
(courseno,book_isbn),foreign key (courseno)
references course (courseno),foreign key (book_isbn)
references text(book_isbn));

insert into student values ('1pe11cs001', 'a', 'jr'
, '19930912'),
('1pe11cs002','b','sr','19930924'),
('1pe11cs003','c','sr','19931127'),
('1pe11cs004','d','sr','19930413'),
('1pe11cs005','e','jr','19940824');

insert into course values (111,'os','cse'),
(112,'ec','ece'),
(113,'ss','ise'),
(114,'dbms','cse'),
(115,'signals','ece');

insert into text values (10,'database
systems','pearson','schield'),
(900,'operating sys','pearson','leland'),
```

```
(901,'circuits','hall india','bob'),
(902,'system software','peterson','jacob'),
(903,'scheduling','pearson','patil'),
(904,'database systems','pearson','jacob'),
(905,'database manager','pearson','bob'),
(906,'signals','hall india','sumit');

insert into enroll values ('1pe11cs001',115,3,100),
```

```
('1pe11cs002',114,5,100),
('1pe11cs003',113,5,100),
('1pe11cs004',111,5,100),
('1pe11cs005',112,3,100);

insert into book_adoption values (111,5,900),
(111,5,903),
(111,5,904),
(112,3,901),
(113,3,10),
(114,5,905),
(113,5,902),
(115,3,906);

select * from student;
select * from course;
select * from text;
select * from enroll;
select * from book_adoption;

select c.courseno,t.book_isbn,t.book_title from
course c, book_adoption ba, text t where
c.courseno=ba.courseno and ba.book_isbn=t.book_isbn
and c.dept='cse' and
2 < (select count(book_isbn)from book_adoption b
where c.courseno = b.courseno) order by
t.book_title;

select distinct c.dept from course c where c.dept in
( select c.dept from course c,book_adoption b,text t
where c.courseno=b.courseno and
t.book_isbn=b.book_isbn and t.publisher='pearson')
```

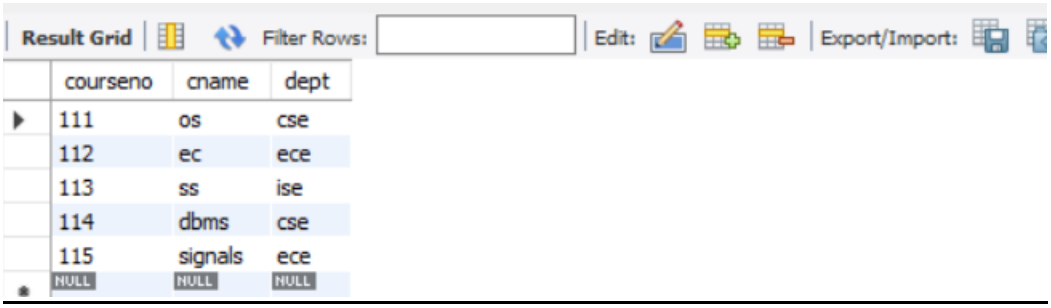
```
and c.dept not in (select c.dept from course
c,book_adoption b,text t where c.courseno=b.courseno
and t.book_isbn=b.book_isbn and t.publisher !=
'pearson');
```

Outputs: -



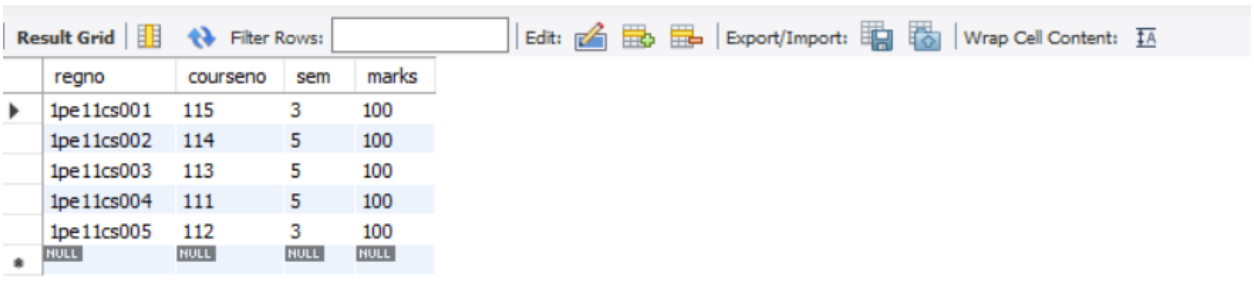
Result Grid | Filter Rows: | Edit:

courseno	sem	book_isbn
111	5	900
111	5	903
111	5	904
112	3	901
113	3	10
113	5	902
114	5	905
115	3	906
NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import:

courseno	cname	dept
111	os	cse
112	ec	ece
113	ss	ise
114	dbms	cse
115	signals	ece
NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

regno	courseno	sem	marks
1pe11cs001	115	3	100
1pe11cs002	114	5	100
1pe11cs003	113	5	100
1pe11cs004	111	5	100
1pe11cs005	112	3	100
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

regno	name	major	bdate
1pe11cs001	a	jr	1993-09-12
1pe11cs002	b	sr	1993-09-24
1pe11cs003	c	sr	1993-11-27
1pe11cs004	d	sr	1993-04-13
1pe11cs005	e	jr	1994-08-24
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_isbn	book_title	publisher	author
10	database systems	pearson	schield
900	operating sys	pearson	leland
901	circuits	hall india	bob
902	system software	peterson	jacob
903	scheduling	pearson	patil
904	database systems	pearson	jacob
905	database manager	pearson	bob
906	signals	hall india	sumit
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

courseno	book_isbn	book_title
111	904	database systems
111	900	operating sys
111	903	scheduling

Result Grid | Filter Rows: | Export: | Wrap Cell Con

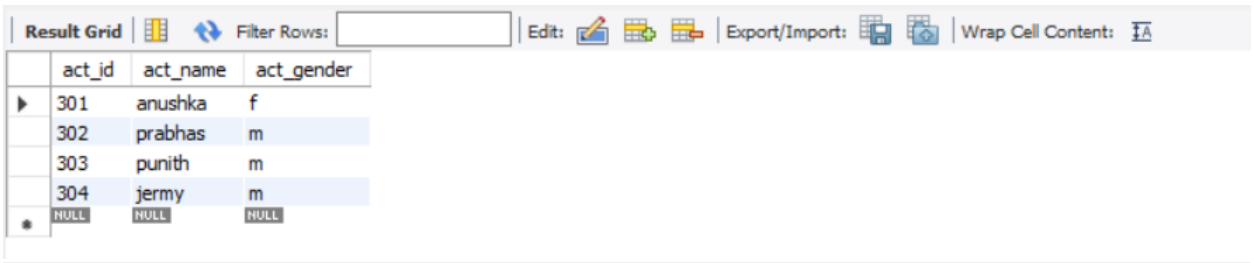
dept
cse

Lab Program 9: -Movie Database

```
create database movies;
use movies;
create table actor ( act_id int, act_name varchar (20), act_gender char(1), primary key (act_id));
create table director ( dir_id int, dir_name varchar (20), dir_phone long, primary key (dir_id));
create table movies ( mov_id int, mov_title varchar (25), mov_year int, mov_lang varchar (12), dir_id int, primary key (mov_id), foreign key (dir_id) references director (dir_id));
create table movie_cast ( act_id int, mov_id int, role varchar (10), primary key (act_id, mov_id), foreign key (act_id) references actor (act_id), foreign key (mov_id) references movies (mov_id));
create table rating ( mov_id int, rev_stars varchar (25), primary key (mov_id), foreign key (mov_id) references movies (mov_id));

insert into actor values (301,'anushka','f');
insert into actor values (302,'prabhas','m');
insert into actor values (303,'punith','m');
insert into actor values (304,'jermy','m');

insert into director values (60,'rajamouli', 8751611001);
insert into director values (61,'hitchcock', 7766138911);
insert into director values (62,'faran', 9986776531);
insert into director values (63,'steven spielberg', 8989776530);

insert into movies values (1001,'bahubali-2', 2017, 'telagu', 60);
insert into movies values (1002,'bahubali-1', 2015, 'telagu', 60);
insert into movies values (1003,'akash', 2008, 'kannada', 61);
```

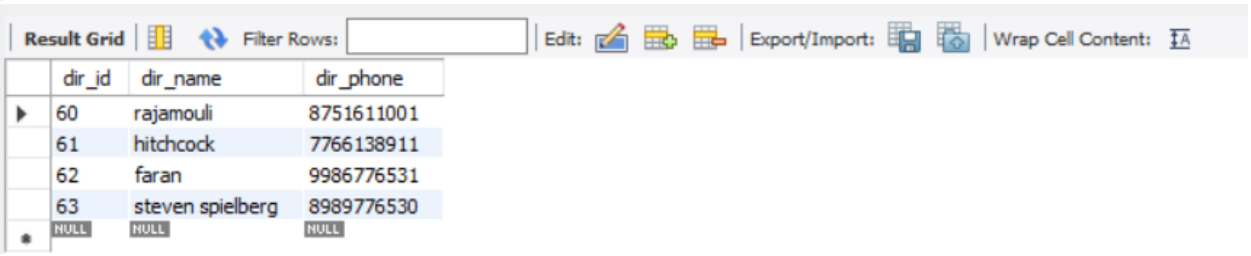
```
insert into movies values (1004,'war horse', 2011, 'english',
63);

insert into movie_cast values (301, 1002, 'heroine');
insert into movie_cast values (301, 1001, 'heroine');
insert into movie_cast values (303, 1003, 'hero');
insert into movie_cast values (303, 1002, 'guest');
insert into movie_cast values (304, 1004, 'hero');

insert into rating values (1001, 4);
insert into rating values (1002, 2);
insert into rating values (1003, 5);
insert into rating values (1004, 4);

select * from actor;
select * from director;
select * from movies;
select * from movie_cast;
select * from rating;

select mov_title from movies where dir_id in (select dir_id from
director where dir_name = 'hitchcock');

select mov_title from movies m, movie_cast mv where
m.mov_id=mv.mov_id and act_id in (select act_id from movie_cast
group by act_id having count(act_id) > 1) group by mov_title
having count(*)>1;

select a.act_name, c.mov_title, c.mov_year from actor a,
movie_cast b, movies c where a.act_id=b.act_id and
b.mov_id=c.mov_id and c.mov_year not between 2000 and 2015;

select mov_title, max(rev_stars) from movies inner join rating
using(mov_id) group by mov_title having max(rev_stars) > 0 order
by mov_title;

update rating set rev_stars=5 where mov_id in (select mov_id
from movies where dir_id in (select dir_id from director where
dir_name = 'steven spielberg'));
select * from rating;
```

Output: -



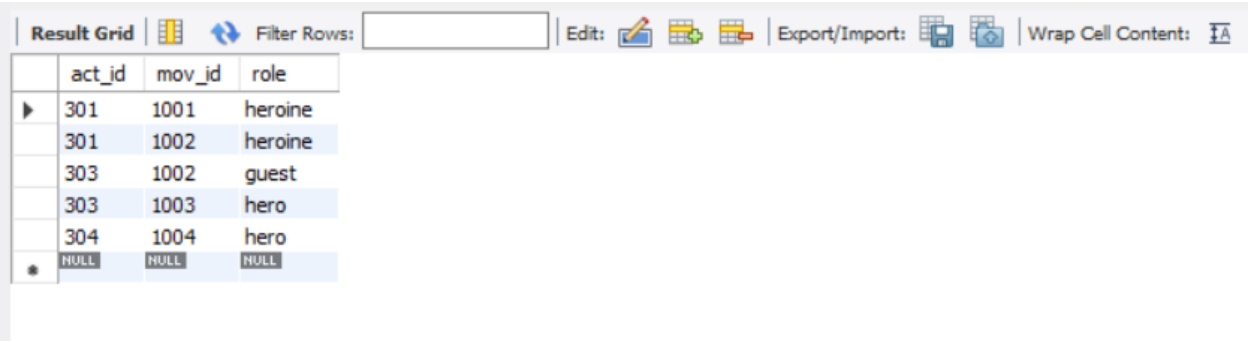
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

act_id	act_name	act_gender
301	anushka	f
302	prabhas	m
303	punith	m
304	jermy	m
NULL	NULL	NULL



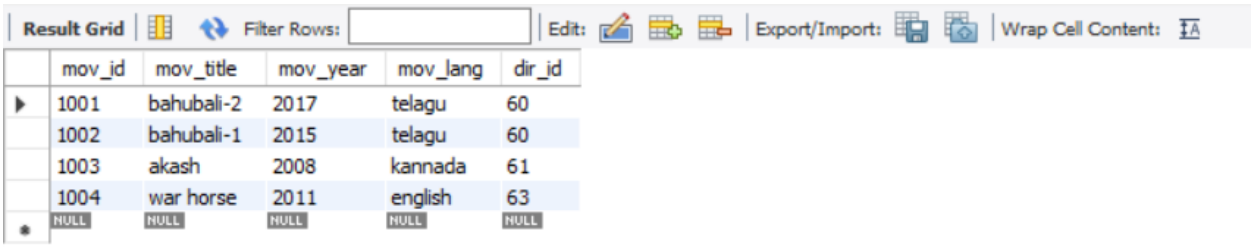
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

dir_id	dir_name	dir_phone
60	rajamouli	8751611001
61	hitchcock	7766138911
62	faran	9986776531
63	steven spielberg	8989776530
NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

act_id	mov_id	role
301	1001	heroine
301	1002	heroine
303	1002	guest
303	1003	hero
304	1004	hero
NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

mov_id	mov_title	mov_year	mov_lang	dir_id
1001	bahubali-2	2017	telagu	60
1002	bahubali-1	2015	telagu	60
1003	akash	2008	kannada	61
1004	war horse	2011	english	63
NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

mov_id	rev_stars
1001	4
1002	2
1003	5
1004	4
NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

mov_title
akash

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

mov_title
bahubali-1

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

act_name	mov_title	mov_year
anushka	bahubali-2	2017

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

mov_title	max(rev_stars)
akash	5
bahubali-1	2
bahubali-2	4
war horse	4

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

mov_id	rev_stars
1001	4
1002	2
1003	5
1004	5
NULL	NULL

Lab Program 10: - College Database

```
create database college;
use college;

CREATE TABLE STUDENT (
USN VARCHAR (10) PRIMARY KEY,
SNAME VARCHAR (25),
ADDRESS VARCHAR (25),
PHONE INT (10),
GENDER CHAR (1));

CREATE TABLE SEMSEC (
SSID VARCHAR (5) PRIMARY KEY,
SEM INT (2),
SEC CHAR (1));

CREATE TABLE CLASS (
USN VARCHAR (10),
SSID VARCHAR (5),
PRIMARY KEY (USN, SSID),
FOREIGN KEY (USN) REFERENCES STUDENT (USN),
FOREIGN KEY (SSID) REFERENCES SEMSEC (SSID));

CREATE TABLE SUBJECT (
SUBCODE VARCHAR (8),
TITLE VARCHAR (20),
SEM INT(2),
CREDITS INT (2),
PRIMARY KEY (SUBCODE));

CREATE TABLE IAMARKS (
USN VARCHAR (10),
SUBCODE VARCHAR (8),
SSID VARCHAR (5),
TEST1 INT(2),
TEST2 INT(2),
TEST3 INT(2),
FINALIA INT(2),
```

```
PRIMARY KEY (USN, SUBCODE, SSID),
FOREIGN KEY (USN) REFERENCES STUDENT (USN),
FOREIGN KEY (SUBCODE) REFERENCES SUBJECT (SUBCODE),
FOREIGN KEY (SSID) REFERENCES SEMSEC (SSID));

desc student;
desc semsec;
desc class;
desc subject;
desc iamarks;

INSERT INTO STUDENT VALUES ('1RN13CS020','AKSHAY','BELAGAVI',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN13CS062','SANDHYA','BENGALURU',
1232654578,'F');
INSERT INTO STUDENT VALUES ('1RN13CS091','TEESHA','BENGALURU',
1232654578,'F');
INSERT INTO STUDENT VALUES ('1RN13CS066','SUPRIYA','MANGALURU',
1232654578,'F');
INSERT INTO STUDENT VALUES ('1RN14CS010','ABHAY','BENGALURU',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN14CS032','BHASKAR','BENGALURU',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN14CS025','ASMI','BENGALURU',
1232654578,'F');
INSERT INTO STUDENT VALUES ('1RN15CS011','AJAY','TUMKUR',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN15CS029','CHITRA','DAVANGERE',
1232654578,'F');
INSERT INTO STUDENT VALUES ('1RN15CS045','JEEVA','BELLARY',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN15CS091','SANTOSH','MANGALURU',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN16CS045','ISMAIL','KALBURGI',
1232654578,'M');
INSERT INTO STUDENT VALUES ('1RN16CS088','SAMEERA','SHIMOGA',
1232654578,'F');
INSERT INTO STUDENT VALUES
('1RN16CS122','VINAYAKA','CHIKAMAGALUR', 1232654578,'M');
select * from student;

INSERT INTO SEMSEC VALUES ('CSE8A', 8,'A');
```

```
INSERT INTO SEMSEC VALUES ('CSE8B', 8,'B');
INSERT INTO SEMSEC VALUES ('CSE8C', 8,'C');
INSERT INTO SEMSEC VALUES ('CSE7A', 7,'A');
INSERT INTO SEMSEC VALUES ('CSE7B', 7,'B');
INSERT INTO SEMSEC VALUES ('CSE7C', 7,'C');
INSERT INTO SEMSEC VALUES ('CSE6A', 6,'A');
INSERT INTO SEMSEC VALUES ('CSE6B', 6,'B');
INSERT INTO SEMSEC VALUES ('CSE6C', 6,'C');
INSERT INTO SEMSEC VALUES ('CSE5A', 5,'A');
INSERT INTO SEMSEC VALUES ('CSE5B', 5,'B');
INSERT INTO SEMSEC VALUES ('CSE5C', 5,'C');
INSERT INTO SEMSEC VALUES ('CSE4A', 4,'A');
INSERT INTO SEMSEC VALUES ('CSE4B', 4,'B');
INSERT INTO SEMSEC VALUES ('CSE4C', 4,'C');
INSERT INTO SEMSEC VALUES ('CSE3A', 3,'A');
INSERT INTO SEMSEC VALUES ('CSE3B', 3,'B');
INSERT INTO SEMSEC VALUES ('CSE3C', 3,'C');
INSERT INTO SEMSEC VALUES ('CSE2A', 2,'A');
INSERT INTO SEMSEC VALUES ('CSE2B', 2,'B');
INSERT INTO SEMSEC VALUES ('CSE2C', 2,'C');
INSERT INTO SEMSEC VALUES ('CSE1A', 1,'A');
INSERT INTO SEMSEC VALUES ('CSE1B', 1,'B');
INSERT INTO SEMSEC VALUES ('CSE1C', 1,'C');
select * from semsec;

INSERT INTO CLASS VALUES ('1RN13CS020','CSE8A');
INSERT INTO CLASS VALUES ('1RN13CS062','CSE8A');
INSERT INTO CLASS VALUES ('1RN13CS066','CSE8B');
INSERT INTO CLASS VALUES ('1RN13CS091','CSE8C');
INSERT INTO CLASS VALUES ('1RN14CS010','CSE7A');
INSERT INTO CLASS VALUES ('1RN14CS025','CSE7A');
INSERT INTO CLASS VALUES ('1RN14CS032','CSE7A');
INSERT INTO CLASS VALUES ('1RN15CS011','CSE4A');
INSERT INTO CLASS VALUES ('1RN15CS029','CSE4A');
select * from class;

INSERT INTO SUBJECT VALUES ('10CS81','ACA', 8, 4);
INSERT INTO SUBJECT VALUES ('10CS82','SSM', 8, 4);
INSERT INTO SUBJECT VALUES ('10CS83','NM', 8, 4);
INSERT INTO SUBJECT VALUES ('10CS84','CC', 8, 4);
INSERT INTO SUBJECT VALUES ('10CS85','PW', 8, 4);
INSERT INTO SUBJECT VALUES ('10CS71','OOAD', 7, 4);
```

```
INSERT INTO SUBJECT VALUES ('10CS72','ECS', 7, 4);
INSERT INTO SUBJECT VALUES ('10CS73','PTW', 7, 4);
INSERT INTO SUBJECT VALUES ('10CS74','DWDM', 7, 4);
select * from subject;

INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)
VALUES ('1RN13CS091','10CS81','CSE8C', 15, 16, 18);
INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)
VALUES ('1RN13CS091','10CS82','CSE8C', 12, 19, 14);
INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)
VALUES ('1RN13CS091','10CS83','CSE8C', 19, 15, 20);
INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)
VALUES ('1RN13CS091','10CS84','CSE8C', 20, 16, 19);
INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)
VALUES ('1RN13CS091','10CS85','CSE8C', 15, 15, 12);
select * from iamarks;

/*  List all the student details studying in fourth semester ‘A’
section */

SELECT S.*, SS.SEM, SS.SEC
FROM STUDENT S, SEMSEC SS, CLASS C
WHERE S.USN = C.USN AND SS.SSID = C.SSID AND SS.SEM = 4 AND
SS.SEC='A';

/*  Compute the total number of male and female students in each
semester and in each section */

SELECT SS.SEM, SS.SEC, S.GENDER, COUNT(S.GENDER) AS COUNT
FROM STUDENT S, SEMSEC SS, CLASS C
WHERE S.USN = C.USN AND
SS.SSID = C.SSID
GROUP BY SS.SEM, SS.SEC, S.GENDER
ORDER BY SEM;

/*  Create a view of Test1 marks of student USN ‘1BI15CS101’ in
all subjects */

CREATE VIEW STU_TEST1_MARKS_VIEW
AS
SELECT TEST1, SUBCODE
FROM IAMARKS
```

```
WHERE USN = '1RN13CS091';
select * from STU_TEST1_MARKS_VIEW;

/*  Categorize students based on the following criterion:
If FinalIA = 17 to 20 then CAT = ‘Outstanding’
If FinalIA = 12 to 16 then CAT = ‘Average’
If FinalIA< 12 then CAT = ‘Weak’
Give these details only for 8th semester A, B, and C section
students */

SELECT S.USN,S.SNAME,S.ADDRESS,S.PHONE,S.GENDER,
(CASE
WHEN IA.FINALIA BETWEEN 17 AND 20 THEN 'OUTSTANDING'
WHEN IA.FINALIA BETWEEN 12 AND 16 THEN 'AVERAGE'
ELSE 'WEAK'
END) AS CAT
FROM STUDENT S, SEMSEC SS, IAMARKS IA, SUBJECT SUB
WHERE S.USN = IA.USN AND
SS.SSID = IA.SSID AND
SUB.SUBCODE = IA.SUBCODE AND
SUB.SEM = 8;
commit
```

Outputs: -

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

USN	SSID
1RN15CS011	CSE4A
1RN15CS029	CSE4A
1RN14CS010	CSE7A
1RN14CS025	CSE7A
1RN14CS032	CSE7A
1RN13CS020	CSE8A
1RN13CS062	CSE8A
1RN13CS066	CSE8B
1RN13CS091	CSE8C
NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

USN	SUBCODE	SSID	TEST1	TEST2	TEST3	FINALIA
1RN13CS091	10CS81	CSE8C	15	16	18	NULL
1RN13CS091	10CS82	CSE8C	12	19	14	NULL
1RN13CS091	10CS83	CSE8C	19	15	20	NULL
1RN13CS091	10CS84	CSE8C	20	16	19	NULL
1RN13CS091	10CS85	CSE8C	15	15	12	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

SSID	SEM	SEC
CSE1A	1	A
CSE1B	1	B
CSE1C	1	C
CSE2A	2	A
CSE2B	2	B
CSE2C	2	C
CSE3A	3	A
CSE3B	3	B
CSE3C	3	C
CSE4A	4	A
CSE4B	4	B
CSE4C	4	C
CSE5A	5	A
CSE5B	5	B
CSE5C	5	C
CSE6A	6	A
CSE6B	6	B
CSE6C	6	C
CSE7A	7	A
CSE7B	7	B
CSE7C	7	C
CSE8A	8	A
CSE8B	8	B
CSE8C	8	C
NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

USN	SNAME	ADDRESS	PHONE	GENDER
1RN13CS020	AKSHAY	BELAGAVI	1232654578	M
1RN13CS062	SANDHYA	BENGALURU	1232654578	F
1RN13CS066	SUPRIYA	MANGALURU	1232654578	F
1RN13CS091	TEESHA	BENGALURU	1232654578	F
1RN14CS010	ABHAY	BENGALURU	1232654578	M
1RN14CS025	ASMI	BENGALURU	1232654578	F
1RN14CS032	BHASKAR	BENGALURU	1232654578	M
1RN15CS011	AJAY	TUMKUR	1232654578	M
1RN15CS029	CHITRA	DAVANGERE	1232654578	F
1RN15CS045	JEEVA	BELLARY	1232654578	M
1RN15CS091	SANTOSH	MANGALURU	1232654578	M
1RN16CS045	ISMAIL	KALBURGI	1232654578	M
1RN16CS088	SAMEERA	SHIMOGA	1232654578	F
1RN16CS122	VINAYAKA	CHIKAMAG...	1232654578	M
NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

SUBCODE	TITLE	SEM	CREDITS
10CS71	OOAD	7	4
10CS72	ECS	7	4
10CS73	PTW	7	4
10CS74	DWDM	7	4
10CS81	ACA	8	4
10CS82	SSM	8	4
10CS83	NM	8	4
10CS84	CC	8	4
10CS85	PW	8	4
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

USN	SNAME	ADDRESS	PHONE	GENDER	SEM	SEC
1RN15CS011	AJAY	TUMKUR	1232654578	M	4	A
1RN15CS029	CHITRA	DAVANGERE	1232654578	F	4	A

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SEM	SEC	GENDER	COUNT
4	A	F	1
4	A	M	1
7	A	F	1
7	A	M	2
8	A	F	1
8	A	M	1
8	B	F	1
8	C	F	1

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

TEST1	SUBCODE
15	10CS81
12	10CS82
19	10CS83
20	10CS84
15	10CS85

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

USN	SNAME	ADDRESS	PHONE	GENDER	CAT
1RN13CS091	TEESHA	BENGALURU	1232654578	F	WEAK
1RN13CS091	TEESHA	BENGALURU	1232654578	F	WEAK
1RN13CS091	TEESHA	BENGALURU	1232654578	F	WEAK
1RN13CS091	TEESHA	BENGALURU	1232654578	F	WEAK
1RN13CS091	TEESHA	BENGALURU	1232654578	F	WEAK

The End