DBMS LAB REPORT 2

Mallika Prasad

1BM19CS081

Lab 6

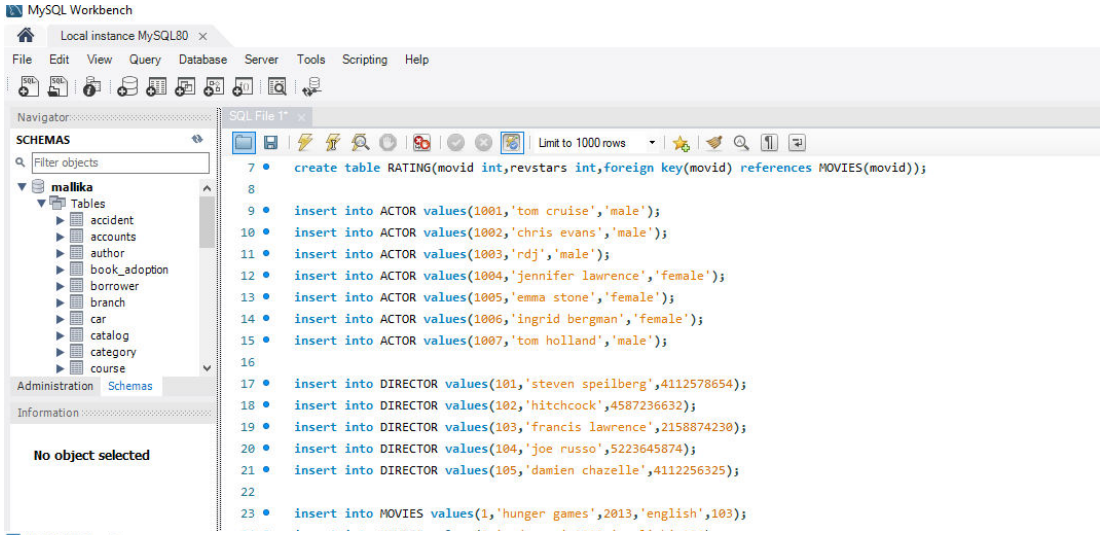
MOVIE Database

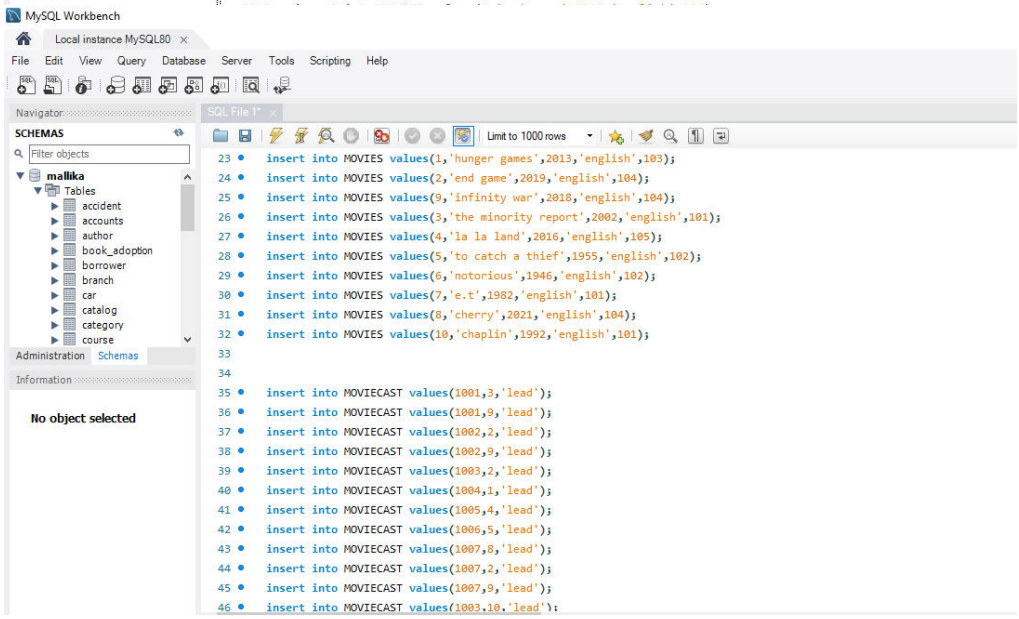
```
use mallika;

create table ACTOR(actid int, actname varchar(30), actgender varchar(30), primary key(actid));

create table DIRECTOR(dirid int,dirname varchar(30),dirphone int,primary key(dirid));

alter table DIRECTOR modify column dirphone varchar(30);

create table MOVIES(movid int,movtitle varchar(50),movyear int,movlang varchar(30),dirid int, primary key(movid),foreign key(dirid) references DIRECTOR(dirid));

create table MOVIECAST(actid int,movid int, role varchar(30), foreign key(actid) references ACTOR(actid),foreign key(movid) references MOVIES(movid));

create table RATING(movid int,revstars int,foreign key(movid) references MOVIES(movid));


insert into ACTOR values(1001,'tom cruise','male');

insert into ACTOR values(1002,'chris evans','male');

insert into ACTOR values(1003,'rdj','male');

insert into ACTOR values(1004,'jennifer lawrence','female');

insert into ACTOR values(1005,'emma stone','female');

insert into ACTOR values(1006,'ingrid bergman','female');

insert into ACTOR values(1007,'tom holland','male');
```

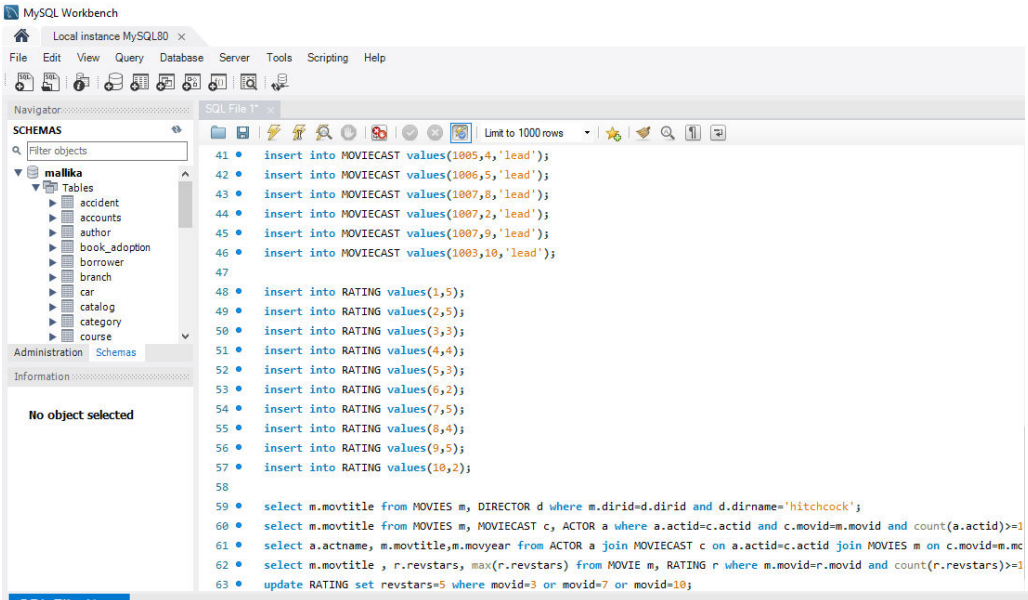
```
insert into DIRECTOR values(101,'steven speilberg',4112578654);
insert into DIRECTOR values(102,'hitchcock',4587236632);
insert into DIRECTOR values(103,'francis lawrence',2158874230);
insert into DIRECTOR values(104,'joe russo',5223645874);
insert into DIRECTOR values(105,'damien chazelle',4112256325);

insert into MOVIES values(1,'hunger games',2013,'english',103);
insert into MOVIES values(2,'end game',2019,'english',104);
insert into MOVIES values(9,'infinity war',2018,'english',104);
insert into MOVIES values(3,'the minority report',2002,'english',101);
insert into MOVIES values(4,'la la land',2016,'english',105);
insert into MOVIES values(5,'to catch a thief',1955,'english',102);
insert into MOVIES values(6,'notorious',1946,'english',102);
insert into MOVIES values(7,'e.t',1982,'english',101);
insert into MOVIES values(8,'cherry',2021,'english',104);
insert into MOVIES values(10,'chaplin',1992,'english',101);

insert into MOVIECAST values(1001,3,'lead');
insert into MOVIECAST values(1001,9,'lead');
insert into MOVIECAST values(1002,2,'lead');
insert into MOVIECAST values(1002,9,'lead');
insert into MOVIECAST values(1003,2,'lead');
insert into MOVIECAST values(1004,1,'lead');
insert into MOVIECAST values(1005,4,'lead');
```

```
insert into MOVIECAST values(1006,5,'lead');

insert into MOVIECAST values(1007,8,'lead');

insert into MOVIECAST values(1007,2,'lead');

insert into MOVIECAST values(1007,9,'lead');

insert into MOVIECAST values(1003,10,'lead');


insert into RATING values(1,5);

insert into RATING values(2,5);

insert into RATING values(3,3);

insert into RATING values(4,4);

insert into RATING values(5,3);

insert into RATING values(6,2);

insert into RATING values(7,5);

insert into RATING values(8,4);

insert into RATING values(9,5);

insert into RATING values(10,2);


select m.movtitle from MOVIES m, DIRECTOR d where m.dirid=d.dirid and d.dirname='hitchcock';

select m.movtitle from MOVIES m, MOVIECAST c where m.movid=c.movid and actid in (select actid from
MOVIECAST group by actid having count(actid)>1) group by movtitle having count(movtitle)>1;

select a.actname, m.movtitle,m.movyear from ACTOR a join MOVIECAST c on a.actid=c.actid join
MOVIES m on c.movid=m.movid where m.movyear and m.movyear not between 2000 and 2015;

select m.movtitle , r.revstars, max(r.revstars) from MOVIES m, RATING r where m.movid=r.movid and
count(r.revstars)>=1 order by m.movtitle;

select movtitle, max(revstars) from MOVIES inner join RATING using (movid) group by movtitle having
max(revstars)>0 order by movtitle;

update RATING set revstars=5 where movid=3 or movid=7 or movid=10;
```

select * from RATING;







SQL File 1*

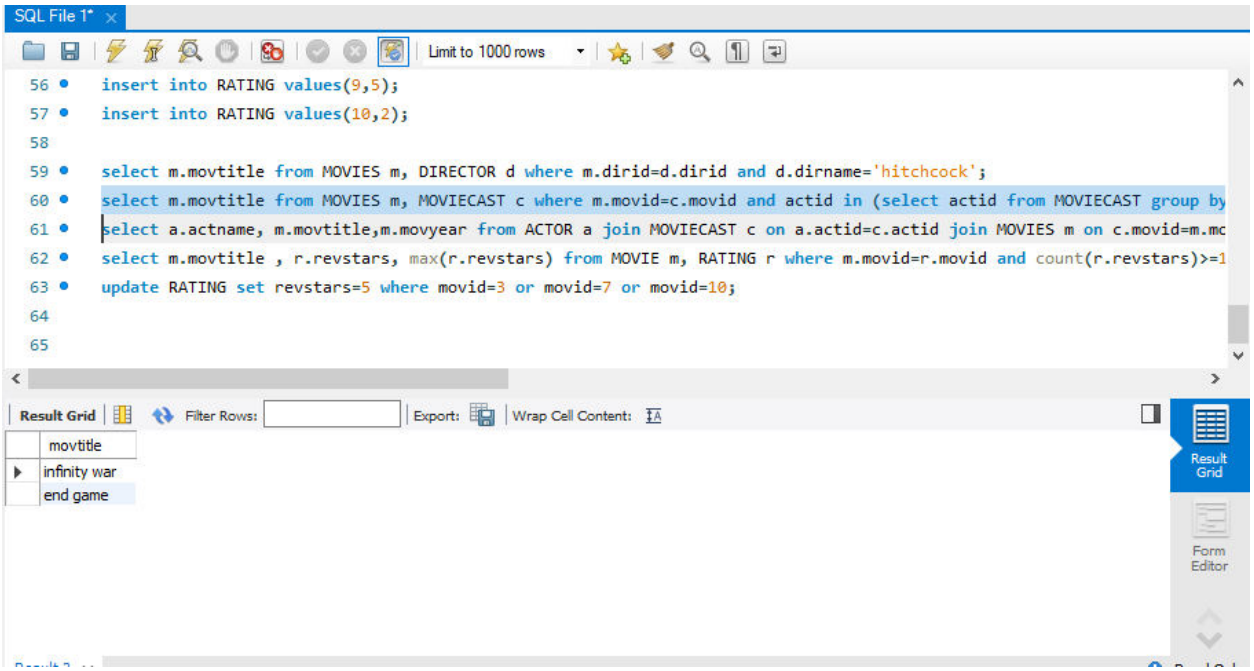
Limit to 1000 rows

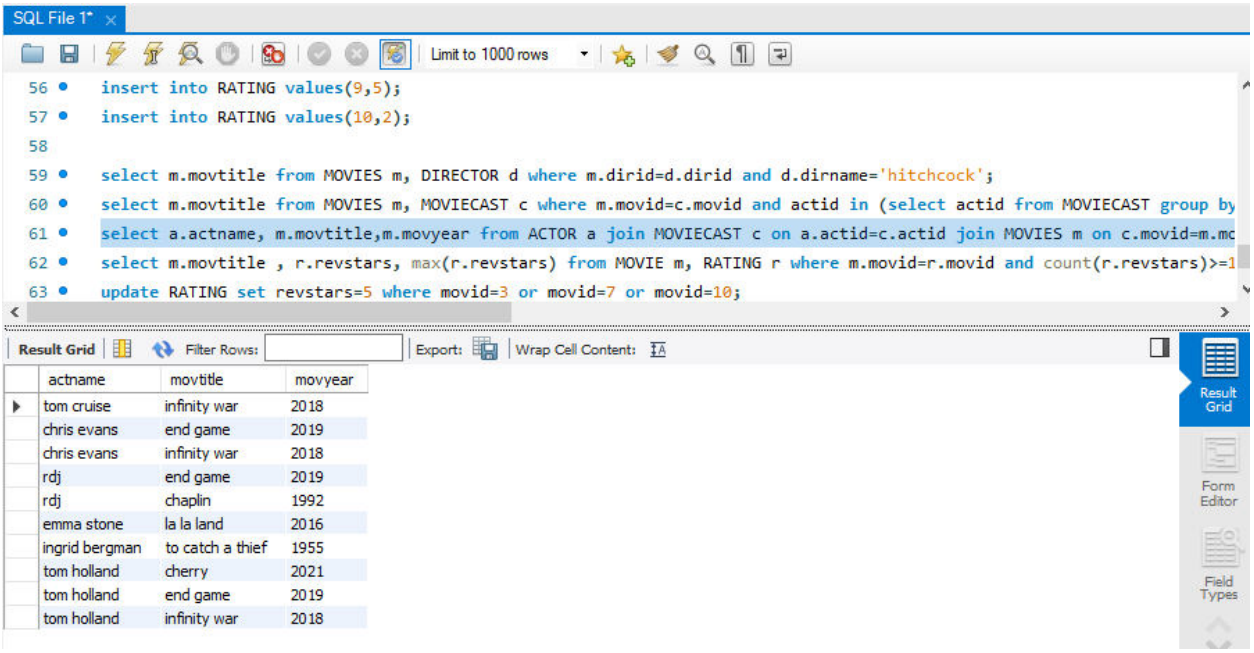
```
48 •  insert into RATING values(1,5);
49 •  insert into RATING values(2,5);
50 •  insert into RATING values(3,3);
51 •  insert into RATING values(4,4);
52 •  insert into RATING values(5,3);
53 •  insert into RATING values(6,2);
54 •  insert into RATING values(7,5);
55 •  insert into RATING values(8,4);
56 •  insert into RATING values(9,5);
57 •  insert into RATING values(10,2);
58
59 •  select m.movtitle from MOVIES m, DIRECTOR d where m.dirid=d.dirid and d.dirname='hitchcock';
```

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

movtitle
to catch a thief
notorious

Result Grid
Form Editor
Field Types


SQL File 1*
Limit to 1000 rows
56 insert into RATING values(9,5);
57 insert into RATING values(10,2);
58
59 select m.movtitle from MOVIES m, DIRECTOR d where m.dirid=d.dirid and d.dirname='hitchcock';
60 select m.movtitle from MOVIES m, MOVIECAST c where m.movid=c.movid and actid in (select actid from MOVIECAST group by
61 select a.actname, m.movtitle,m.movyear from ACTOR a join MOVIECAST c on a.actid=c.actid join MOVIES m on c.movid=m.mc
62 select m.movtitle , r.revstars, max(r.revstars) from MOVIE m, RATING r where m.movid=r.movid and count(r.revstars)>=1
63 update RATING set revstars=5 where movid=3 or movid=7 or movid=10;
64
65
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
movtitle
infinity war
end game
Result Grid
Form Editor


SQL File 1*
Limit to 1000 rows
56 insert into RATING values(9,5);
57 insert into RATING values(10,2);
58
59 select m.movtitle from MOVIES m, DIRECTOR d where m.dirid=d.dirid and d.dirname='hitchcock';
60 select m.movtitle from MOVIES m, MOVIECAST c where m.movid=c.movid and actid in (select actid from MOVIECAST group by
61 select a.actname, m.movtitle,m.movyear from ACTOR a join MOVIECAST c on a.actid=c.actid join MOVIES m on c.movid=m.mc
62 select m.movtitle , r.revstars, max(r.revstars) from MOVIE m, RATING r where m.movid=r.movid and count(r.revstars)>=1
63 update RATING set revstars=5 where movid=3 or movid=7 or movid=10;
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
actname movtitle movyear
tom cruise infinity war 2018
chris evans end game 2019
chris evans infinity war 2018
rdj end game 2019
rdj chaplin 1992
emma stone la la land 2016
ingrid bergman to catch a thief 1955
tom holland cherry 2021
tom holland end game 2019
tom holland infinity war 2018
Result Grid
Form Editor
Field Types

SQL File 1*

```
54  insert into RATING values(7,5);
55  insert into RATING values(8,4);
56  insert into RATING values(9,5);
57  insert into RATING values(10,2);
58
59  select m.movtitle from MOVIES m, DIRECTOR d where m.dirid=d.dirid and d.dirname='hitchcock';
60  select m.movtitle from MOVIES m, MOVIECAST c where m.movid=c.movid and actid in (select actid from MOVIECAST group by
61  select a.actname, m.movtitle,m.movyear from ACTOR a join MOVIECAST c on a.actid=c.actid join MOVIES m on c.movid=m.mc
62  select m.movtitle , r.revstars, max(r.revstars) from MOVIES m, RATING r where m.movid=r.movid and count(r.revstars)>=
63  select movtitle, max(revstars) from MOVIES inner join RATING using (movid) group by movtitle having max(revstars)>0 c
64  update RATING set revstars=5 where movid=3 or movid=7 or movid=10;
```

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

movtitle	max(revstars)
chaplin	2
cherry	4
e.t	5
end game	5
hunger games	5
infinity war	5
la la land	4
notorious	2
the minority report	3
to catch a thief	3

Result Grid | Form Editor | Field Types

SQL File 1*

```
62  select m.movtitle , r.revstars, max(r.revstars) from MOVIES m, RATING r where m.movid=r.movid and count(r.revstars)>=
63  select movtitle, max(revstars) from MOVIES inner join RATING using (movid) group by movtitle having max(revstars)>0 c
64  update RATING set revstars=5 where movid=3 or movid=7 or movid=10;
65  select * from RATING;
66
```

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

movid	revstars
1	5
2	5
3	5
4	4
5	3
6	2
7	5
8	4
9	5
10	5

Result Grid | Form Editor | Field Types

RATING 6 | Read Only

LAB 7

AIRLINES Database

```
use mallika;

create table FLIGHTS(flno int,fromm varchar(30),too varchar(30),distance int,departs time,arrives time,
price int,primary key(flno));

create table AIRCRAFT(aid int,aname varchar(30),cruisingrange int, primary key(aid));

create table CERTIFIED(eid int,aid int,foreign key(eid) references EMPLOYEE(eid), foreign key(aid)
references AIRCRAFT(aid));

create table EMPLOYEE(eid int, ename varchar(30),salary int, primary key(eid));


INSERT INTO FLIGHTS (flno,fromm,too,distance,departs,arrives,price) VALUES

(1,'Bangalore','Chennai',360,'08:45','10:00',10000),

(2,'Bangalore','Delhi',1700,'12:15','15:00',37000),

(3,'Bangalore','Kolkata',1500,'15:15','05:25',30000),

(4,'Mumbai','Delhi',1200,'10:30','12:30',28000),

(5,'Bangalore','New york',14000,'05:45','02:30',90000),

(6,'Delhi','Chicago',12000,'10:00','05:45',95000),

(7,'Bangalore','Frankfurt',15000,'12:00','06:30',98000),

(8,'Madison','New york',1500,'10:15','14:25',30000);



INSERT INTO AIRCRAFT (aid,aname,cruisingrange) values

    (1,'Airbus 380',1000),

    (2,'Boeing 737',4000),

    (3,'Lockheed',5500),
```

```
    (4,'Airbus A220',9500),
    (5,'Boeing 747',800),
    (6,'Douglas DC3',900);


INSERT INTO EMPLOYEE (eid,ename,salary) VALUES
(1,'Zoya',95000),
(2,'Akshay',65000),
(3,'Niveditha',70000),
(4,'Safan',45000),
(5,'Peter',95000),
(6,'Nayan',100000),
(7,'Ajay',50000);


INSERT INTO CERTIFIED (eid,aid) VALUES
(1,1),
(1,3),
(1,4),
(5,4),
(5,3),
(1,2),
 (2,6),
(2,5),
(4,5),
```

(6,4),

(6,3),

(3,6),

(3,2);

SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eid AND NOT EXISTS (SELECT * FROM EMPLOYEE E1 WHERE E1.eid = E.eid AND E1.salary < 80000));

SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) > 3;

SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore' AND f.too='Frankfurt');

SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cruisingrange>1000 GROUP BY a.aid,a.aname;

SELECT distinct e.ename FROM EMPLOYEE e,AIRCRAFT a,CERTIFIED c WHERE e.eid=c.eid AND c.aid=a.aid AND a.aname like 'Boeing%';

SELECT a.aid FROM AIRCRAFT a WHERE a.cruisingrange> (SELECT MIN(f.distance) FROM FLIGHTS f WHERE f.fromm='Bangalore' AND f.too='Delhi');

SELECT F.departs FROM FLIGHTS F WHERE F.flno IN (SELECT F0.flno FROM FLIGHTS F0 WHERE F0.fromm = 'Madison' AND F0.too = 'New york' AND F0.arrives < '18:00');

SELECT E.ename, E.salary FROM EMPLOYEE E WHERE E.eid NOT IN (SELECT DISTINCT C.eid FROM CERTIFIED C) AND E.salary > (SELECT AVG (E1.salary) FROM EMPLOYEE E1 WHERE E1.eid IN (SELECT DISTINCT C1.eid FROM CERTIFIED C1));

airlinesdatabase

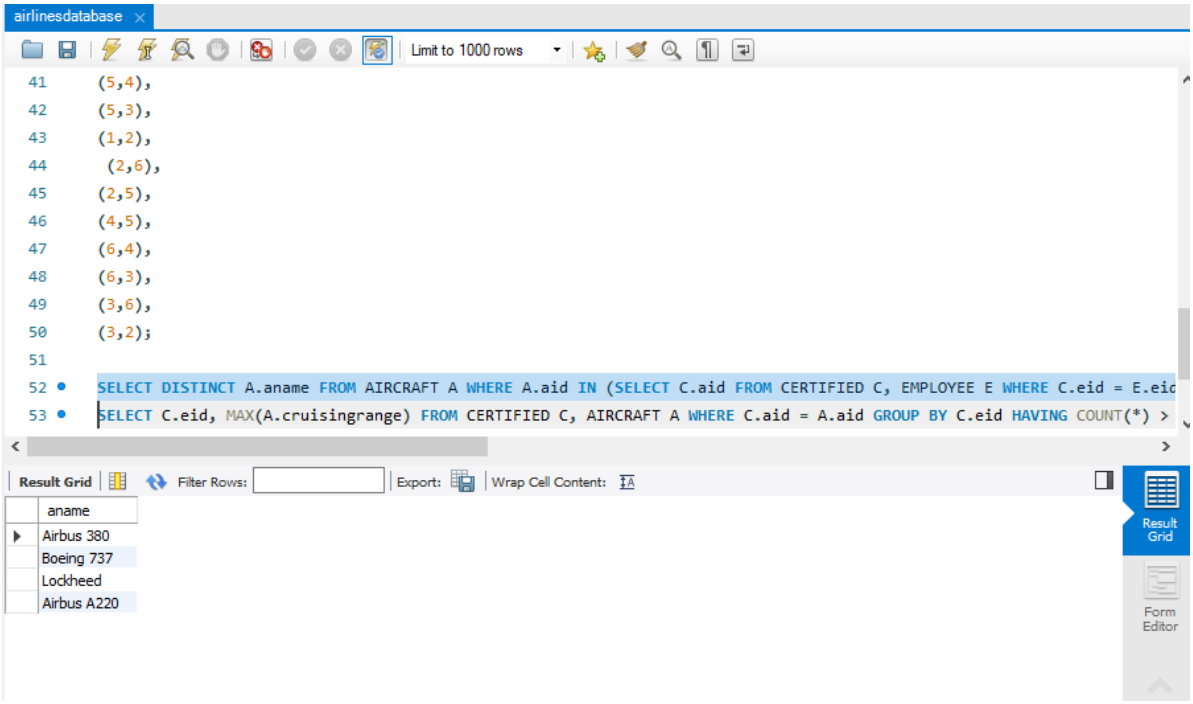
Limit to 1000 rows

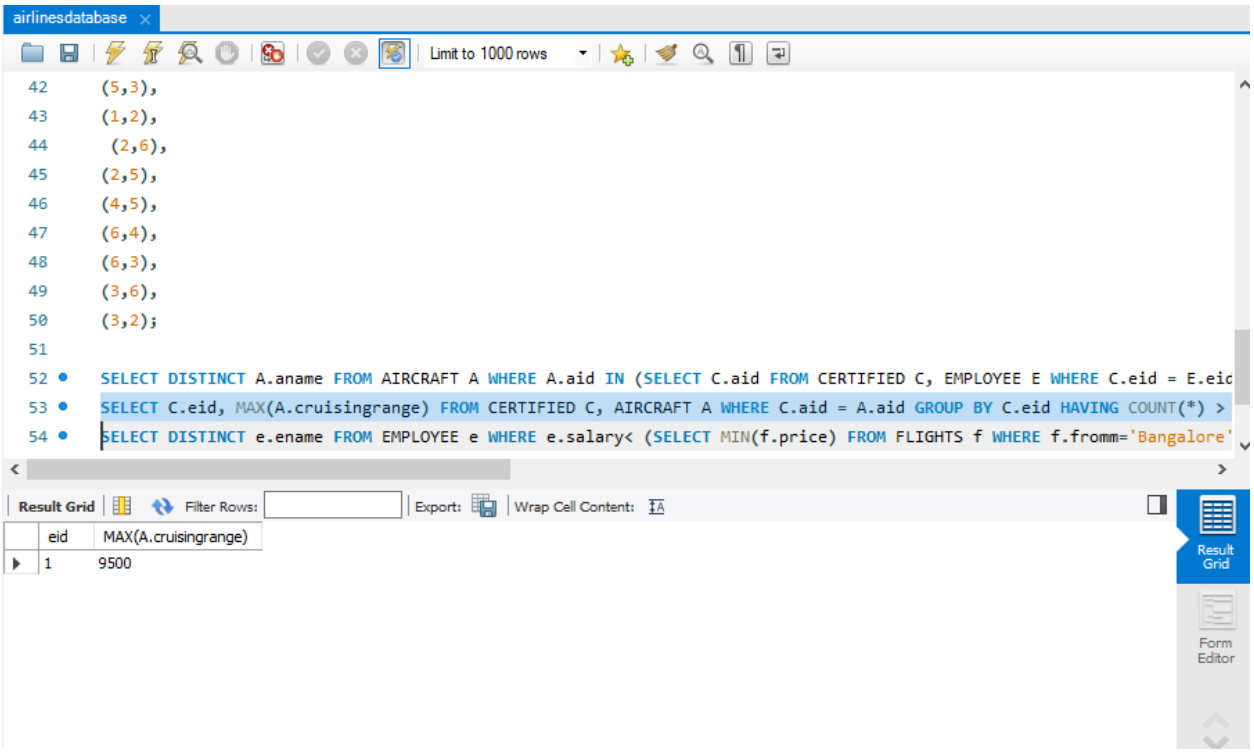
```
6
7   INSERT INTO FLIGHTS (flno,fromm,too,distance,departs,arrives,price) VALUES
8   (1,'Bangalore','Chennai',360,'08:45','10:00',10000),
9   (2,'Bangalore','Delhi',1700,'12:15','15:00',37000),
10  (3,'Bangalore','Kolkata',1500,'15:15','05:25',30000),
11  (4,'Mumbai','Delhi',1200,'10:30','12:30',28000),
12  (5,'Bangalore','New york',14000,'05:45','02:30',90000),
13  (6,'Delhi','Chicago',12000,'10:00','05:45',95000),
14  (7,'Bangalore','Frankfurt',15000,'12:00','06:30',98000),
15  (8,'Madison','New york',1500,'10:15','14:25',30000);
16
17
18  INSERT INTO AIRCRAFT (aid,aname,cruisingrange) values
19          (1,'Airbus 380',1000),
20          (2,'Boeing 737',4000),
21          (3,'Lockheed',5500),
22          (4,'Airbus A220',9500),
23          (5,'Boeing 747',800),
24          (6,'Douglas DC3',900);
25
26
27  INSERT INTO EMPLOYEE (eid,ename,salary) VALUES
28  (1,'Zoya',95000),
```

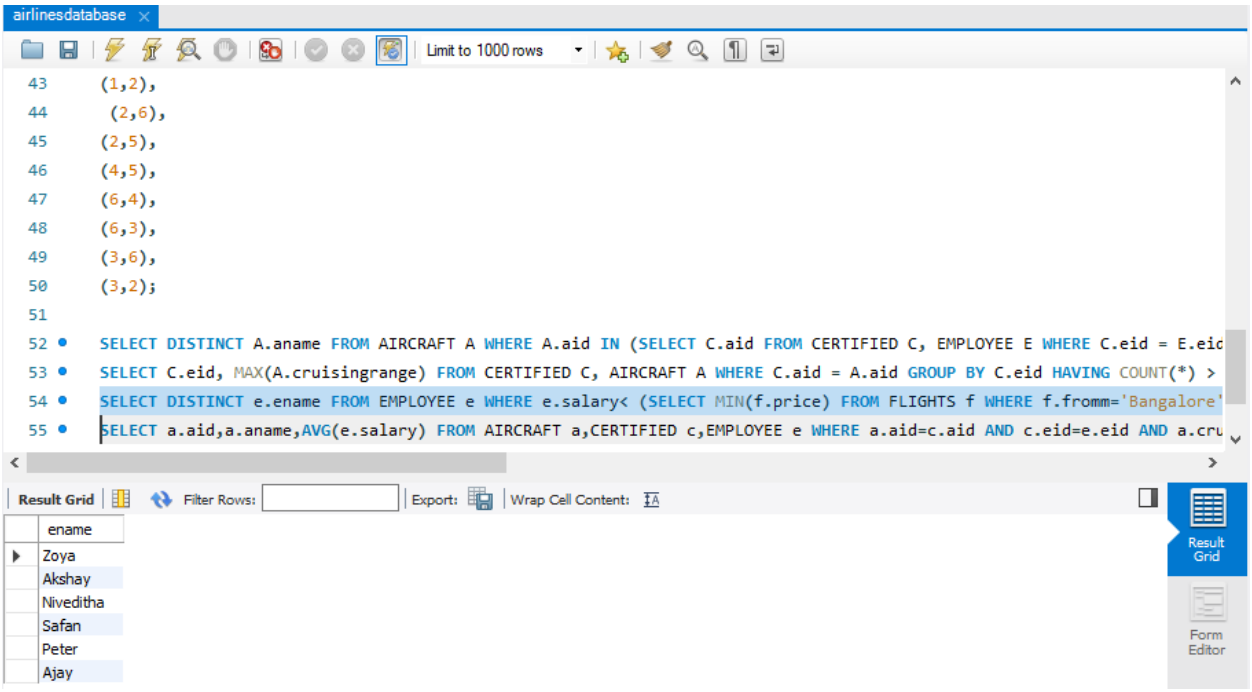
airlinesdatabase

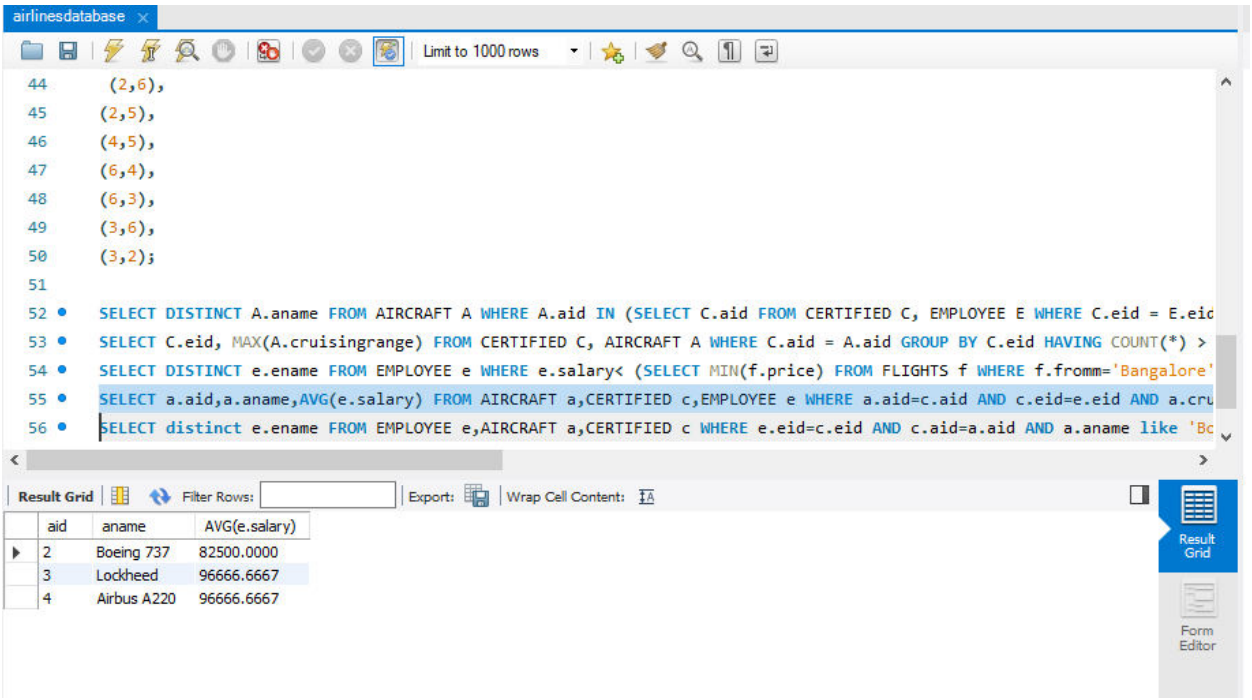
Limit to 1000 rows

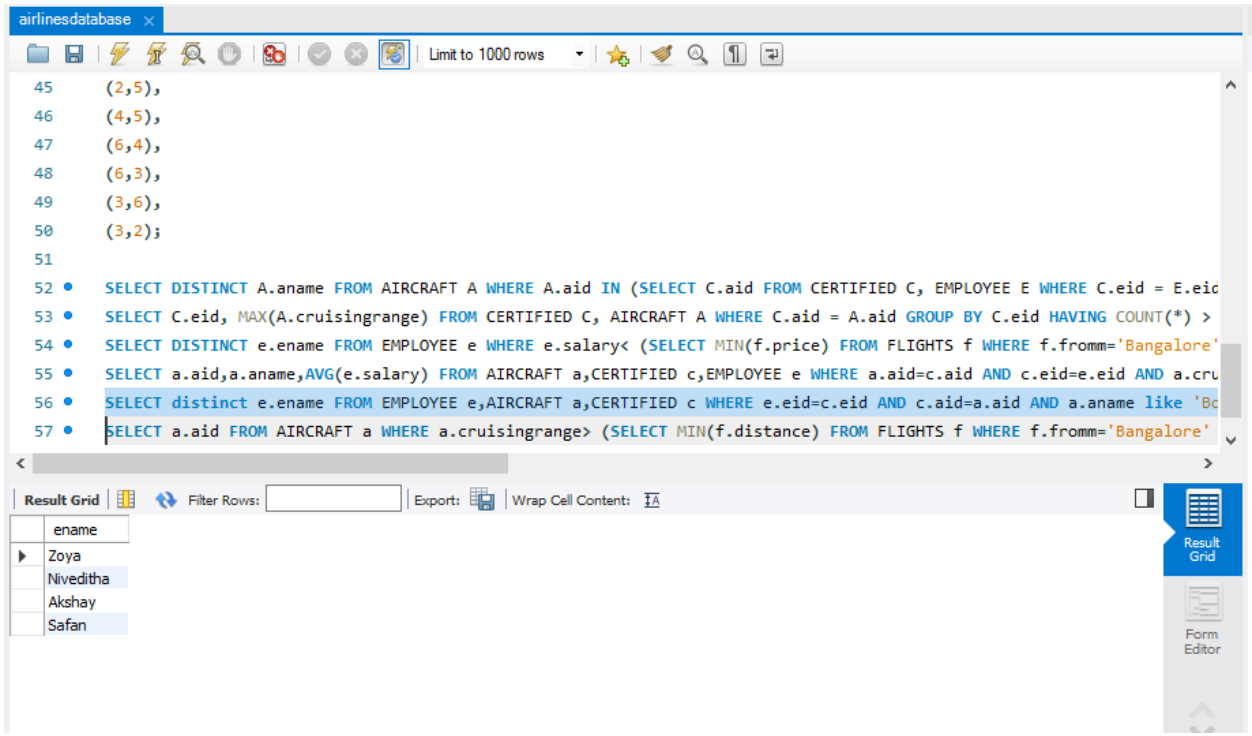
```
26
27  INSERT INTO EMPLOYEE (eid,ename,salary) VALUES
28  (1,'Zoya',95000),
29  (2,'Akshay',65000),
30  (3,'Niveditha',70000),
31  (4,'Safan',45000),
32  (5,'Peter',95000),
33  (6,'Nayan',100000),
34  (7,'Ajay',50000);
35
36
37  INSERT INTO CERTIFIED (eid,aid) VALUES
38  (1,1),
39  (1,3),
40  (1,4),
41  (5,4),
42  (5,3),
43  (1,2),
44   (2,6),
45  (2,5),
46  (4,5),
47  (6,4),
48  (6,3),
49  (3,6),
```

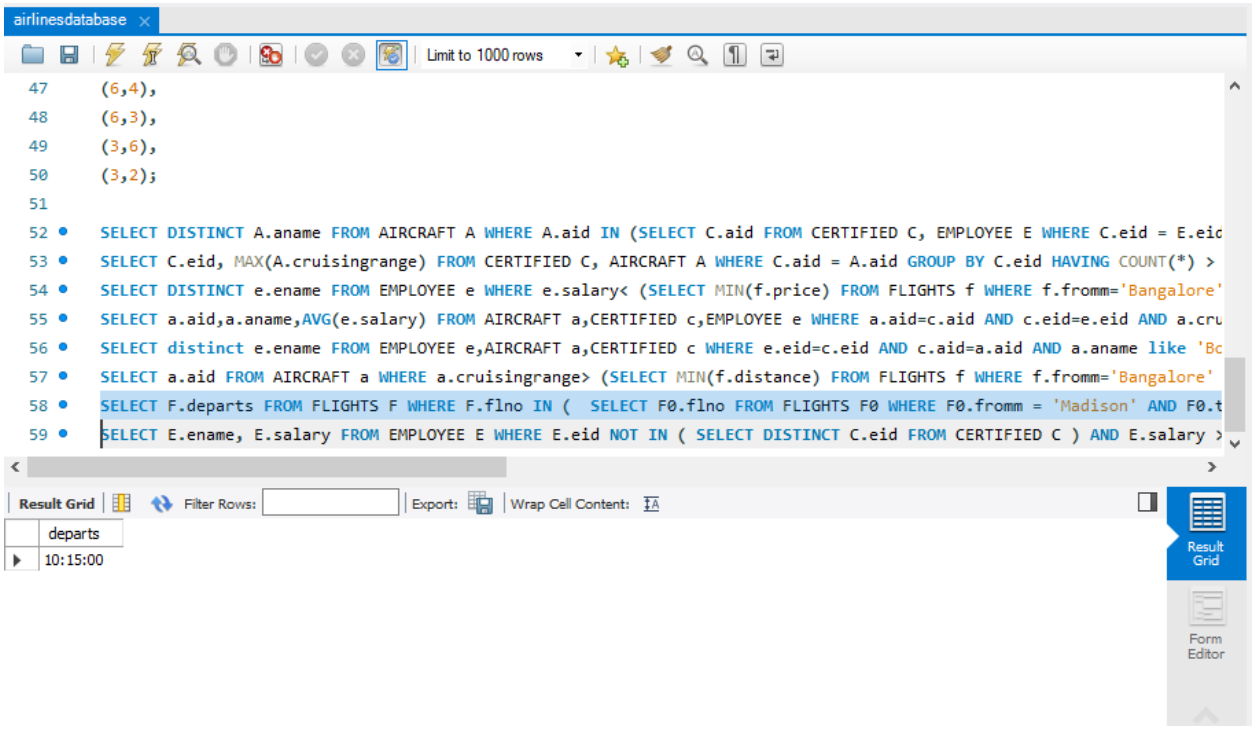

airlinesdatabase
Limit to 1000 rows
(5,4),
(5,3),
(1,2),
(2,6),
(2,5),
(4,5),
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eid
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
aname
Airbus 380
Boeing 737
Lockheed
Airbus A220
Result Grid
Form Editor

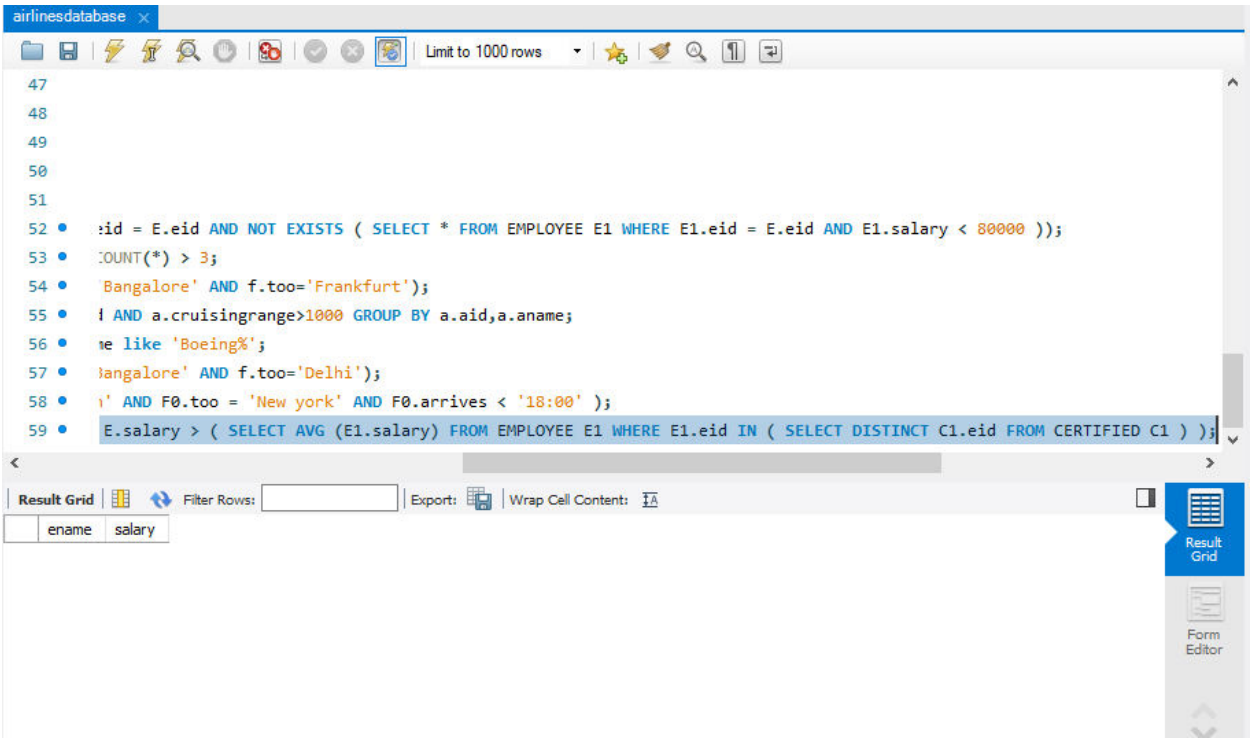

airlinesdatabase
Limit to 1000 rows
(5,3),
(1,2),
(2,6),
(2,5),
(4,5),
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eid
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
eid
MAX(A.cruisingrange)
1
9500
Result Grid
Form Editor

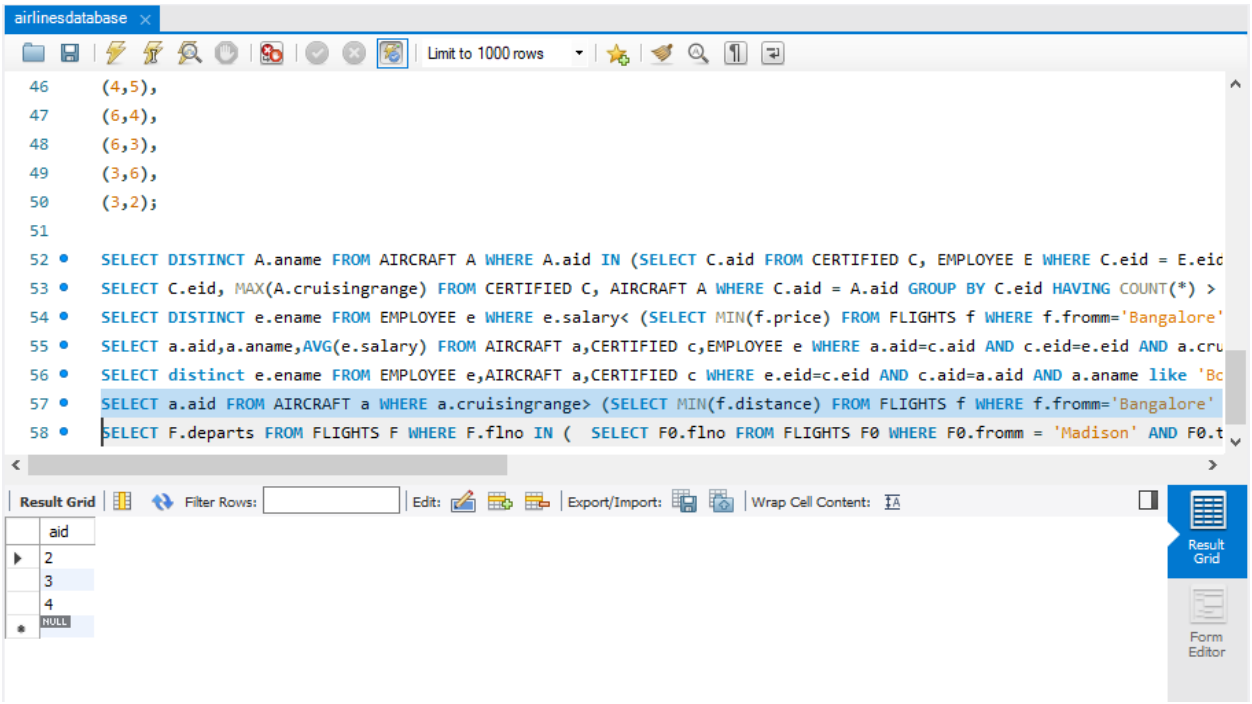

airlinesdatabase
Limit to 1000 rows
(1,2),
(2,6),
(2,5),
(4,5),
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eic
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cru
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
ename
Zoya
Akshay
Niveditha
Safan
Peter
Ajay
Result Grid
Form Editor


airlinesdatabase
Limit to 1000 rows
(2,6),
(2,5),
(4,5),
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eic
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cru
SELECT distinct e.ename FROM EMPLOYEE e,AIRCRAFT a,CERTIFIED c WHERE e.eid=c.eid AND c.aid=a.aid AND a.aname like 'Bo
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
aid
aname
AVG(e.salary)
2
Boeing 737
82500.0000
3
Lockheed
96666.6667
4
Airbus A220
96666.6667
Result Grid
Form Editor


airlinesdatabase
Limit to 1000 rows
(2,5),
(4,5),
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eic
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cru
SELECT distinct e.ename FROM EMPLOYEE e,AIRCRAFT a,CERTIFIED c WHERE e.eid=c.eid AND c.aid=a.aid AND a.aname like 'Bo
SELECT a.aid FROM AIRCRAFT a WHERE a.cruisingrange> (SELECT MIN(f.distance) FROM FLIGHTS f WHERE f.fromm='Bangalore'
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
ename
Zoya
Niveditha
Akshay
Safan
Result Grid
Form Editor


airlinesdatabase
Limit to 1000 rows
(6,4),
(6,3),
(3,6),
(3,2);
SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eic
SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cru
SELECT distinct e.ename FROM EMPLOYEE e,AIRCRAFT a,CERTIFIED c WHERE e.eid=c.eid AND c.aid=a.aid AND a.aname like 'Bo
SELECT a.aid FROM AIRCRAFT a WHERE a.cruisingrange> (SELECT MIN(f.distance) FROM FLIGHTS f WHERE f.fromm='Bangalore'
SELECT F.departs FROM FLIGHTS F WHERE F.flno IN (SELECT F0.flno FROM FLIGHTS F0 WHERE F0.fromm = 'Madison' AND F0.t
SELECT E.ename, E.salary FROM EMPLOYEE E WHERE E.eid NOT IN (SELECT DISTINCT C.eid FROM CERTIFIED C) AND E.salary >
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
departs
10:15:00
Result Grid
Form Editor


airlinesdatabase
Limit to 1000 rows
47
48
49
50
51
52 :id = E.eid AND NOT EXISTS (SELECT * FROM EMPLOYEE E1 WHERE E1.eid = E.eid AND E1.salary < 80000));
53 :OUNT(*) > 3;
54 'Bangalore' AND f.too='Frankfurt');
55 I AND a.cruisingrange>1000 GROUP BY a.aid,a.aname;
56 ie like 'Boeing%';
57 Bangalore' AND f.too='Delhi');
58 i' AND F0.too = 'New york' AND F0.arrives < '18:00');
59 E.salary > (SELECT AVG (E1.salary) FROM EMPLOYEE E1 WHERE E1.eid IN (SELECT DISTINCT C1.eid FROM CERTIFIED C1));
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
ename
salary
Result Grid
Form Editor


airlinesdatabase
Limit to 1000 rows
46 (4,5),
47 (6,4),
48 (6,3),
49 (3,6),
50 (3,2);
51
52 SELECT DISTINCT A.aname FROM AIRCRAFT A WHERE A.aid IN (SELECT C.aid FROM CERTIFIED C, EMPLOYEE E WHERE C.eid = E.eic
53 SELECT C.eid, MAX(A.cruisingrange) FROM CERTIFIED C, AIRCRAFT A WHERE C.aid = A.aid GROUP BY C.eid HAVING COUNT(*) >
54 SELECT DISTINCT e.ename FROM EMPLOYEE e WHERE e.salary< (SELECT MIN(f.price) FROM FLIGHTS f WHERE f.fromm='Bangalore'
55 SELECT a.aid,a.aname,AVG(e.salary) FROM AIRCRAFT a,CERTIFIED c,EMPLOYEE e WHERE a.aid=c.aid AND c.eid=e.eid AND a.cru
56 SELECT distinct e.ename FROM EMPLOYEE e,AIRCRAFT a,CERTIFIED c WHERE e.eid=c.eid AND c.aid=a.aid AND a.aname like 'Bc
57 SELECT a.aid FROM AIRCRAFT a WHERE a.cruisingrange> (SELECT MIN(f.distance) FROM FLIGHTS f WHERE f.fromm='Bangalore'
58 SELECT F.departs FROM FLIGHTS F WHERE F.flno IN (SELECT F0.flno FROM FLIGHTS F0 WHERE F0.fromm = 'Madison' AND F0.t
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
aid
2
3
4
NULL
Result Grid
Form Editor

LAB 8

SUPPLIER Database

```
use mallika;

create table Suppliers
(sid int,
sname varchar(30),
address varchar(30),
primary key(sid));

create table Parts
(pid int,
pname varchar(30),
color varchar(15),
primary key(pid));

create table Catalogs
(sid int,
pid int,
cost float,
primary key(sid,pid),
foreign key(sid) references suppliers(sid),
foreign key(pid) references parts(pid));
```

```
insert into Suppliers
values(10001,"Acme Widget","Bangalore"),
            (10002,"Johns","Kolkata"),
    (10003,"Vimal","Mumbai"),
    (10004,"Reliance","Delhi");
select * from suppliers;

insert into Parts
values(20001,"Book","Red"),
    (20002,"Pen","Red"),
    (20003,"Pencil","Green"),
    (20004,"Mobile","Green"),
    (20005,"Charger","Black");
select * from parts;

insert into Catalogs
values(10001,20001,10),
    (10001,20002,10),
    (10001,20003,30),
    (10001,20004,10),
    (10001,20005,10),
    (10002,20001,10),
    (10002,20002,20),
    (10003,20003,30),
```

```
    (10004,20003,40);
select * from catalogs;
/* 1 - FIND THE PNAMES OF PARTS FOR WHICH THERE IS SOME SUPPLIER. */
SELECT DISTINCT P.PNAME
 FROM PARTS P, CATALOGS C
 WHERE P.PID = C.PID;

/* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY PART */
SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (SELECT P.PID FROM
PARTS P WHERE NOT EXISTS (SELECT C.SID FROM CATALOGS C WHERE C.SID =
S.SID AND C.PID = P.PID));

/* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY RED PART. */
SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (SELECT P.PID FROM
PARTS P WHERE P.COLOR = 'RED' AND (NOT EXISTS (SELECT C.SID FROM
CATALOGS C WHERE C.SID = S.SID AND C.PID = P.PID)));

/* FIND THE PNAMES OF PARTS SUPPLIED BY ACME WIDGET SUPPLIERS AND BY NO ONE ELSE */
SELECT P.PNAME FROM PARTS P, CATALOGS C, SUPPLIERS S WHERE P.PID
= C.PID AND C.SID = S.SID AND S.SNAME = 'ACME WIDGET' AND NOT EXISTS
(SELECT * FROM CATALOGS C1, SUPPLIERS S1 WHERE P.PID = C1.PID AND
C1.SID = S1.SID AND S1.SNAME <> 'ACME WIDGET');

/* FIND THE SIDS OF SUPPLIERS WHO CHARGE MORE FOR SOME PART THAN THE AVERAGE COST OF
THAT PART (AVERAGED OVER
ALL THE SUPPLIERS WHO SUPPLY THAT PART).
```

```
 */
SELECT DISTINCT C.SID FROM CATALOGS C
WHERE C.COST > ( SELECT AVG (C1.COST)
FROM CATALOGS C1
WHERE C1.PID = C.PID );

/* FOR EACH PART, FIND THE SNAME OF THE SUPPLIER WHO CHARGES THE MOST FOR THAT PART.*/
SELECT P.PID, S.SNAME
FROM PARTS P, SUPPLIERS S, CATALOGS C
WHERE C.PID = P.PID
AND C.SID = S.SID
AND C.COST = (SELECT MAX(C1.COST)
FROM CATALOGS C1
WHERE C1.PID = P.PID);

/* FIND THE SIDS OF SUPPLIERS WHO SUPPLY ONLY RED PARTS.*/
SELECT DISTINCT C.SID
FROM CATALOGS C
WHERE NOT EXISTS ( SELECT *
FROM PARTS P
WHERE P.PID = C.PID AND P.COLOR <> 'RED' );
```

```
22
23 •  insert into Suppliers
24    values(10001,"Acme Widget","Bangalore"),
25          (10002,"Johns","Kolkata"),
26          (10003,"Vimal","Mumbai"),
27          (10004,"Reliance","Delhi");
28 •  select * from suppliers;
29
30 •  insert into Parts
31    values(20001,"Book","Red"),
32          (20002,"Pen","Red"),
33          (20003,"Pencil","Green"),
34          (20004,"Mobile","Green"),
35          (20005,"Charger","Black");
36 •  select * from parts;
37
38 •  insert into Catalogs
39    values(10001,20001,10),
40          (10001,20002,10),
41          (10001,20003,30),
42          (10001,20004,10),
43          (10001,20005,10),
44          (10002,20001,10),
45          (10002,20002,20),
46          (10003,20003,30),
```

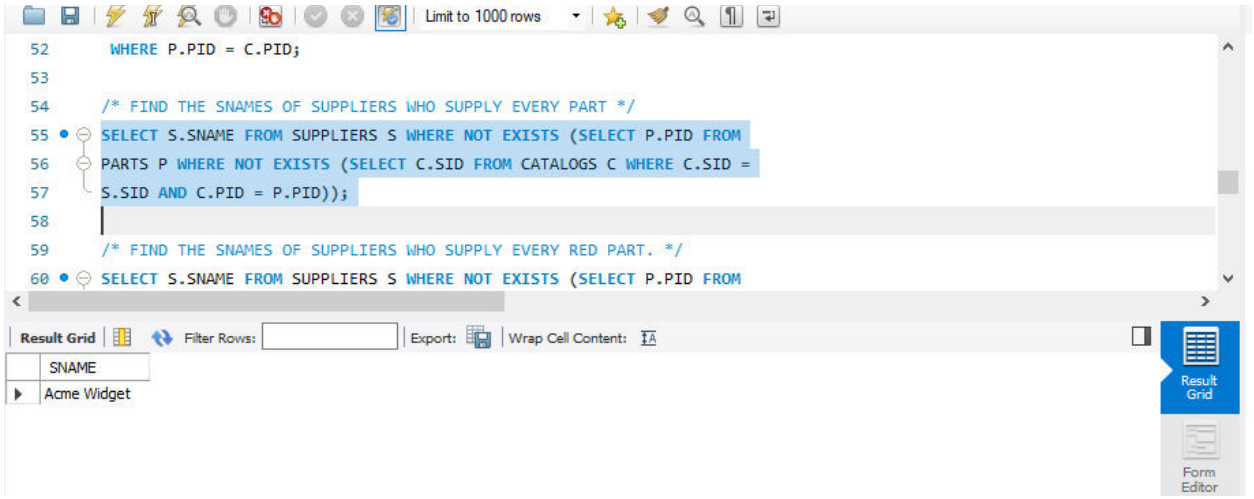
```
46          (10003,20003,30),
47          (10004,20003,40);
48 •  select * from catalogs;
49    /* 1 - FIND THE PNAMES OF PARTS FOR WHICH THERE IS SOME SUPPLIER. */
50 •  SELECT DISTINCT P.PNAME
51     FROM PARTS P, CATALOGS C
52     WHERE P.PID = C.PID;
53
54    /* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY PART */
```

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

PNAME
Book
Pen
Pencil
Mobile
Charger

Result Grid

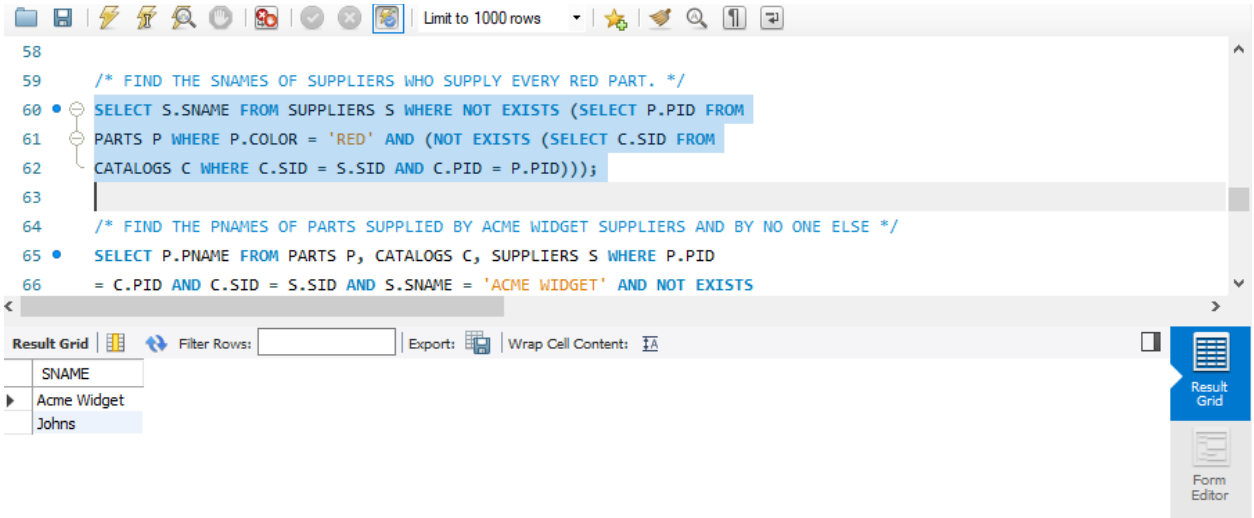
Form Editor



```
52      WHERE P.PID = C.PID;
53
54      /* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY PART */
55      SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (SELECT P.PID FROM
56      PARTS P WHERE NOT EXISTS (SELECT C.SID FROM CATALOGS C WHERE C.SID =
57      S.SID AND C.PID = P.PID));
58
59      /* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY RED PART. */
60      SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (SELECT P.PID FROM
```

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

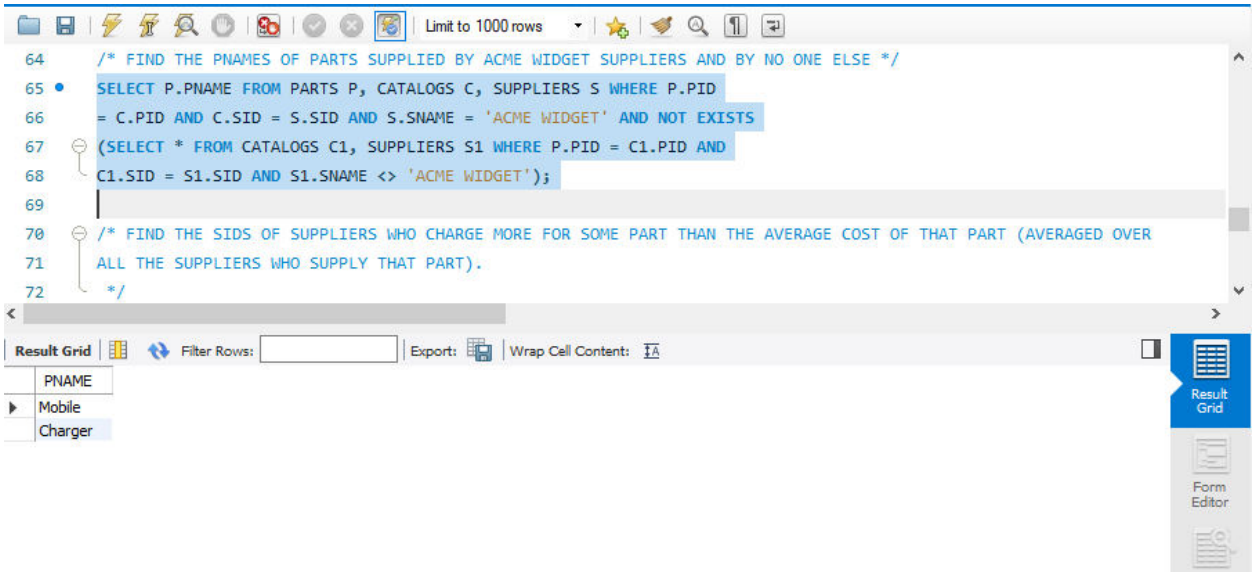
SNAME
Acme Widget



```
58
59      /* FIND THE SNAMES OF SUPPLIERS WHO SUPPLY EVERY RED PART. */
60      SELECT S.SNAME FROM SUPPLIERS S WHERE NOT EXISTS (SELECT P.PID FROM
61      PARTS P WHERE P.COLOR = 'RED' AND (NOT EXISTS (SELECT C.SID FROM
62      CATALOGS C WHERE C.SID = S.SID AND C.PID = P.PID)));
63
64      /* FIND THE PNAMES OF PARTS SUPPLIED BY ACME WIDGET SUPPLIERS AND BY NO ONE ELSE */
65      SELECT P.PNAME FROM PARTS P, CATALOGS C, SUPPLIERS S WHERE P.PID
66      = C.PID AND C.SID = S.SID AND S.SNAME = 'ACME WIDGET' AND NOT EXISTS
```

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

SNAME
Acme Widget
Johns



```
64      /* FIND THE PNAMES OF PARTS SUPPLIED BY ACME WIDGET SUPPLIERS AND BY NO ONE ELSE */
65      SELECT P.PNAME FROM PARTS P, CATALOGS C, SUPPLIERS S WHERE P.PID
66      = C.PID AND C.SID = S.SID AND S.SNAME = 'ACME WIDGET' AND NOT EXISTS
67      (SELECT * FROM CATALOGS C1, SUPPLIERS S1 WHERE P.PID = C1.PID AND
68      C1.SID = S1.SID AND S1.SNAME <> 'ACME WIDGET');
69
70      /* FIND THE SIDS OF SUPPLIERS WHO CHARGE MORE FOR SOME PART THAN THE AVERAGE COST OF THAT PART (AVERAGED OVER
71      ALL THE SUPPLIERS WHO SUPPLY THAT PART).
72       */
```

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

PNAME
Mobile
Charger

```
/* FIND THE SIDS OF SUPPLIERS WHO CHARGE MORE FOR SOME PART THAN THE AVERAGE COST OF THAT PART (AVERAGED OVER
ALL THE SUPPLIERS WHO SUPPLY THAT PART).
 */
SELECT DISTINCT C.SID FROM CATALOGS C
WHERE C.COST > ( SELECT AVG (C1.COST)
FROM CATALOGS C1
WHERE C1.PID = C.PID );

/* FOR EACH PART, FIND THE SNAME OF THE SUPPLIER WHO CHARGES THE MOST FOR THAT PART.*/
```

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

SID
10002
10004

Limit to 1000 rows

```
SELECT P.PID, S.SNAME
FROM PARTS P, SUPPLIERS S, CATALOGS C
WHERE C.PID = P.PID
AND C.SID = S.SID
AND C.COST = (SELECT MAX(C1.COST)
FROM CATALOGS C1
WHERE C1.PID = P.PID);

/* FIND THE SIDS OF SUPPLIERS WHO SUPPLY ONLY RED PARTS.*/
```

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

PID	SNAME
20001	Acme Widget
20004	Acme Widget
20005	Acme Widget
20001	Johns
20002	Johns
20003	Reliance

Limit to 1000 rows

```
WHERE C1.PID = P.PID);

/* FIND THE SIDS OF SUPPLIERS WHO SUPPLY ONLY RED PARTS.*/
SELECT DISTINCT C.SID
FROM CATALOGS C
WHERE NOT EXISTS ( SELECT *
FROM PARTS P
WHERE P.PID = C.PID AND P.COLOR <> 'RED' );
```

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

SID
10001
10002

Result Grid

Form

LAB 9

STUDENT-FACULTY Database

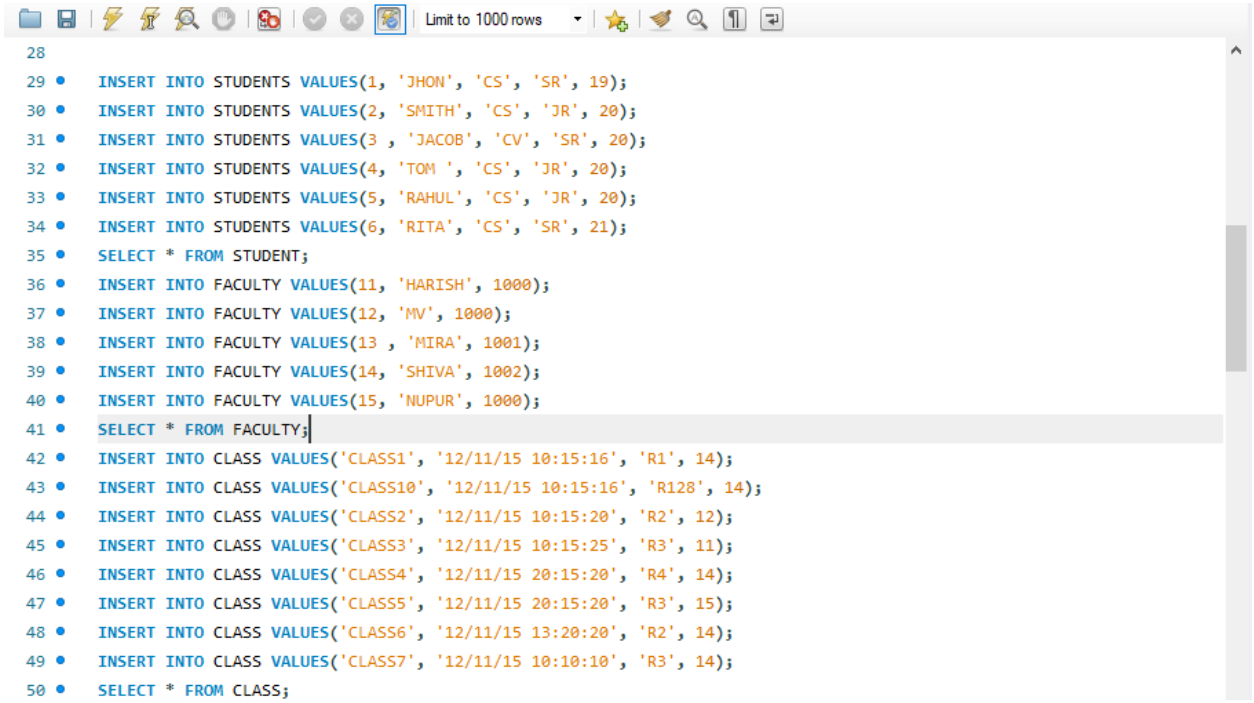
```
USE mallika;
CREATE TABLE STUDENTS(
SNUM INT,
SNAME VARCHAR(10),
MAJOR VARCHAR(2),
LVL VARCHAR(2),
AGE INT, PRIMARY KEY(SNUM));

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 STUDENTS(SNUM),
FOREIGN KEY(CNAME) REFERENCES CLASS(CNAME));

INSERT INTO STUDENTS VALUES(1, 'JHON', 'CS', 'SR', 19);
INSERT INTO STUDENTS VALUES(2, 'SMITH', 'CS', 'JR', 20);
INSERT INTO STUDENTS VALUES(3 , 'JACOB', 'CV', 'SR', 20);
INSERT INTO STUDENTS VALUES(4, 'TOM ', 'CS', 'JR', 20);
INSERT INTO STUDENTS VALUES(5, 'RAHUL', 'CS', 'JR', 20);
INSERT INTO STUDENTS VALUES(6, 'RITA', 'CS', 'SR', 21);
SELECT * FROM STUDENT;
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);
SELECT * FROM FACULTY;
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);

SELECT * FROM CLASS;

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 ENROLLED;

-- QUERY 1

SELECT DISTINCT S.SNAME

FROM STUDENTS 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='ROOM128'

OR
```

```
CNAME IN (SELECT E.CNAME FROM ENROLLED E GROUP BY E.CNAME HAVING COUNT(*)>=5);

-- QUERY 3

SELECT DISTINCT S.SNAME

FROM STUDENTS 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 STUDENTS S

WHERE S.SNUM NOT IN (SELECT E.SNUM
```

```
FROM ENROLLED E );

-- QUERY 7

SELECT S.AGE, S.LVL

FROM STUDENTS S

GROUP BY S.AGE, S.LVL

HAVING S.LVL IN (SELECT S1.LVL

FROM STUDENTS S1

WHERE S1.AGE=S.AGE

GROUP BY S1.AGE, S1.LVL

HAVING COUNT(*) >= ALL (SELECT COUNT(*)

FROM STUDENTS S2

WHERE S1.AGE=S2.AGE

GROUP BY S2.LVL, S2.AGE))

ORDER BY S.AGE;
```



```
INSERT INTO CLASS VALUES('CLASS7', '12/11/15 10:10:10', 'R3', 14);
SELECT * FROM CLASS;
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 ENROLLED;
```

```
INSERT INTO ENROLLED VALUES(5, 'CLASS5');
SELECT * FROM ENROLLED;
-- QUERY 1
SELECT DISTINCT S.SNAME
FROM STUDENTS 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='ROOM128'
```

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

SNAME
TOM

```
-- QUERY 2
SELECT DISTINCT CNAME
FROM CLASS
WHERE ROOM='ROOM128'
OR
CNAME IN (SELECT E.CNAME FROM ENROLLED E GROUP BY E.CNAME HAVING COUNT(*)>=5);
-- QUERY 3
SELECT DISTINCT S.SNAME
FROM STUDENTS S
WHERE S.SNUM IN (SELECT E1.SNUM
```

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

CNAME
CLASS5
NULL

```
81    -- QUERY 4
82    SELECT F.FNAME,F.FID
83    FROM FACULTY F
84    WHERE F.FID IN ( SELECT FID FROM CLASS
85    GROUP BY FID HAVING COUNT(*)=(SELECT COUNT(DISTINCT ROOM) FROM CLASS) );
86    -- QUERY 5
```

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

FNAME	FID
SHIVA	14
NULL	NULL

```
85    GROUP BY FID HAVING COUNT(*)=(SELECT COUNT(DISTINCT ROOM) FROM CLASS) );
86    -- QUERY 5
87    SELECT DISTINCT F.FNAME
88    FROM FACULTY F
89    WHERE 5 > (SELECT COUNT(E.SNUM)
90    FROM CLASS C, ENROLLED E
91    WHERE C.CNAME = E.CNAME
92    AND C.FID = F.FID);
93    -- QUERY 6
```

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

FNAME
HARISH
MV
MIRA
SHIVA

```
72    CNAME IN (SELECT E.CNAME FROM ENROLLED E GROUP BY E.CNAME HAVING COUNT(*)>=5);
73    -- QUERY 3
74    SELECT DISTINCT S.SNAME
75    FROM STUDENTS S
76    WHERE S.SNUM IN (SELECT E1.SNUM
77    FROM ENROLLED E1, ENROLLED E2, CLASS C1, CLASS C2
78    WHERE E1.SNUM = E2.SNUM AND E1.CNAME <> E2.CNAME
79    AND E1.CNAME = C1.CNAME
80    AND E2.CNAME = C2.CNAME AND C1.METTS_AT = C2.METTS_AT);
81    -- QUERY 4
82    SELECT F.FNAME,F.FID
```

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

SNAME
RAHUL

```
93      -- QUERY 6
94      SELECT DISTINCT S.SNAME
95      FROM STUDENTS S
96      WHERE S.SNUM NOT IN (SELECT E.SNUM
97      FROM ENROLLED E );
98      -- QUERY 7
```

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

SNAME
RITA

```
98      -- QUERY 7
99      SELECT S.AGE, S.LVL
100     FROM STUDENTS S
101     GROUP BY S.AGE, S.LVL
102     HAVING S.LVL IN (SELECT S1.LVL
103     FROM STUDENTS S1
104     WHERE S1.AGE=S.AGE
105     GROUP BY S1.AGE, S1.LVL
106     HAVING COUNT(*) >= ALL (SELECT COUNT(*)
107     FROM STUDENTS S2
108     WHERE S1.AGE=S2.AGE
109     GROUP BY S2.LVL, S2.AGE))
110     ORDER BY S.AGE;
```

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

AGE	LVL
19	SR
20	JR
21	SR

LAB 10

COLLEGE Database

```
create database college;

use college;

CREATE TABLE STUDENT (

USN VARCHAR (10) PRIMARY KEY,

SNAME VARCHAR (25),

ADDRESS VARCHAR (25),

PHONE real,

GENDER CHAR (1));

CREATE TABLE SEMSEC (

SSID VARCHAR (5) PRIMARY KEY,

SEM INT ,

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,
```

```
CREDITS INT,
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));
INSERT INTO STUDENT VALUES('1RN13CS020','AKSHAY','BELAGAVI',8877881122,'M');
INSERT INTO STUDENT VALUES('1RN13CS062','SANDHYA','BENGALURU',7722829912,'F');
INSERT INTO STUDENT VALUES('1RN13CS091','TEESHA','BENGALURU',7712312312,'F');
INSERT INTO STUDENT VALUES('1RN13CS066','SUPRIYA','MANGALURU',8877881122,'F');
INSERT INTO STUDENT VALUES('1RN14CS010','ABHAY','BENGALURU',9900211201,'M');
INSERT INTO STUDENT VALUES('1RN14CS032','BHASKAR','BENGALURU',9923211099,'M');
INSERT INTO STUDENT VALUES ('1RN14CS025','ASMI','BENGALURU', 7894737377,'F');
INSERT INTO STUDENT VALUES ('1RN15CS011','AJAY','TUMKUR', 9845091341,'M');
INSERT INTO STUDENT VALUES ('1RN15CS029','CHITRA','DAVANGERE',7696772121,'F');
INSERT INTO STUDENT VALUES ('1RN15CS045','JEEVA','BELLARY', 9944850121,'M');
INSERT INTO STUDENT VALUES ('1RN15CS091','SANTOSH','MANGALURU',8812332201,'M');
```

```
INSERT INTO STUDENT VALUES('1RN16CS045','ISMAIL','KALBURGI',9900232201,'M');
INSERT INTO STUDENT VALUES ('1RN16CS088','SAMEERA','SHIMOGA',9905542212,'F');
INSERT INTO STUDENT VALUES ('1RN16CS122','VINAYAKA','CHIKAMAGALUR',8800880011,'M');

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,'C');
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');


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');

INSERT INTO CLASS VALUES('1RN15CS045','CSE4B');

INSERT INTO CLASS VALUES('1RN15CS091','CSE4C');

INSERT INTO CLASS VALUES('1RN16CS045','CSE3A');

INSERT INTO CLASS VALUES('1RN16CS088','CSE3B');

INSERT INTO CLASS VALUES('1RN16CS122','CSE3C');


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);

INSERT INTO SUBJECT VALUES ('10CS75','JAVA', 7, 4);

INSERT INTO SUBJECT VALUES ('10CS76','SAN', 7, 4);

INSERT INTO SUBJECT VALUES ('15CS51', 'ME', 5, 4);

INSERT INTO SUBJECT VALUES ('15CS52','CN', 5, 4);

INSERT INTO SUBJECT VALUES ('15CS53','DBMS', 5, 4);

INSERT INTO SUBJECT VALUES ('15CS54','ATC', 5, 4);

INSERT INTO SUBJECT VALUES ('15CS55','JAVA', 5, 3);

INSERT INTO SUBJECT VALUES ('15CS56','AI', 5, 3);

INSERT INTO SUBJECT VALUES ('15CS41','M4', 4, 4);

INSERT INTO SUBJECT VALUES ('15CS42','SE', 4, 4);

INSERT INTO SUBJECT VALUES ('15CS43','DAA', 4, 4);

INSERT INTO SUBJECT VALUES ('15CS44','MPMC', 4, 4);

INSERT INTO SUBJECT VALUES ('15CS45','OOC', 4, 3);

INSERT INTO SUBJECT VALUES ('15CS46','DC', 4, 3);

INSERT INTO SUBJECT VALUES ('15CS31','M3', 3, 4);

INSERT INTO SUBJECT VALUES ('15CS32','ADE', 3, 4);

INSERT INTO SUBJECT VALUES ('15CS33','DSA', 3, 4);

INSERT INTO SUBJECT VALUES ('15CS34','CO', 3, 4);

INSERT INTO SUBJECT VALUES ('15CS35','USP', 3, 3);

INSERT INTO SUBJECT VALUES ('15CS36','DMS', 3, 3);


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 STUDENT;

SELECT * FROM SEMSEC;

SELECT * FROM CLASS;

SELECT * FROM SUBJECT;

SELECT * FROM IAMARKS;


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='C';


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 VIEW STU_TEST1_MARKS_VIEW

AS

SELECT TEST1, SUBCODE

FROM IAMARKS

WHERE USN = '1RN13CS091';


-- QUERY 4


DELIMITER //


CREATE PROCEDURE AVG_MARKS()

BEGIN

DECLARE C_A INTEGER;

DECLARE C_B INTEGER;

DECLARE C_C INTEGER;

DECLARE C_SUM INTEGER;

DECLARE C_AVG INTEGER;

DECLARE C_USN VARCHAR(10);

DECLARE C_SUBCODE VARCHAR(8);

DECLARE C_SSID VARCHAR(5);

DECLARE C_IAMARKS CURSOR FOR

SELECT GREATEST(TEST1,TEST2) AS A, GREATEST(TEST1,TEST3) AS B, GREATEST(TEST3,TEST2) AS C, USN,
SUBCODE, SSID

FROM IAMARKS

WHERE FINALIA IS NULL
```

```
FOR UPDATE;
OPEN C_IAMARKS;
LOOP
FETCH C_IAMARKS INTO C_A, C_B, C_C, C_USN, C_SUBCODE, C_SSID;
IF (C_A != C_B) THEN
 SET C_SUM=C_A+C_B;
ELSE
 SET C_SUM=C_A+C_C;
END IF;
SET C_AVG=C_SUM/2;
UPDATE IAMARKS SET FINALIA = C_AVG
WHERE USN = C_USN AND SUBCODE = C_SUBCODE AND SSID = C_SSID;
END LOOP;
CLOSE C_IAMARKS;
END;
//

CALL AVG_MARKS();

SELECT * FROM IAMARKS;
-- QUERY 5

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;

```
34      FOREIGN KEY (USN) REFERENCES STUDENT (USN),
35      FOREIGN KEY (SUBCODE) REFERENCES SUBJECT (SUBCODE),
36      FOREIGN KEY (SSID) REFERENCES SEMSEC (SSID));
37 •    INSERT INTO STUDENT VALUES('1RN13CS020','AKSHAY','BELAGAVI',8877881122,'M');
38 •    INSERT INTO STUDENT VALUES('1RN13CS062','SANDHYA','BENGALURU',7722829912,'F');
39 •    INSERT INTO STUDENT VALUES('1RN13CS091','TEESHA','BENGALURU',7712312312,'F');
40 •    INSERT INTO STUDENT VALUES('1RN13CS066','SUPRIYA','MANGALURU',8877881122,'F');
41 •    INSERT INTO STUDENT VALUES('1RN14CS010','ABHAY','BENGALURU',9900211201,'M');
42 •    INSERT INTO STUDENT VALUES('1RN14CS032','BHASKAR','BENGALURU',9923211099,'M');
43 •    INSERT INTO STUDENT VALUES ('1RN14CS025','ASMI','BENGALURU', 7894737377,'F');
44 •    INSERT INTO STUDENT VALUES ('1RN15CS011','AJAY','TUMKUR', 9845091341,'M');
45 •    INSERT INTO STUDENT VALUES ('1RN15CS029','CHITRA','DAVANGERE',7696772121,'F');
46 •    INSERT INTO STUDENT VALUES ('1RN15CS045','JEEVA','BELLARY', 9944850121,'M');
47 •    INSERT INTO STUDENT VALUES ('1RN15CS091','SANTOSH','MANGALURU',8812332201,'M');
48 •    INSERT INTO STUDENT VALUES('1RN16CS045','ISMAIL','KALBURGI',9900232201,'M');
49 •    INSERT INTO STUDENT VALUES ('1RN16CS088','SAMEERA','SHIMOGA',9905542212,'F');
50 •    INSERT INTO STUDENT VALUES ('1RN16CS122','VINAYAKA','CHIKAMAGALUR',8800880011,'M');
51
52 •    INSERT INTO SEMSEC VALUES ('CSE8A', 8,'A');
53 •    INSERT INTO SEMSEC VALUES ('CSE8B', 8,'B');
54 •    INSERT INTO SEMSEC VALUES ('CSE8C',8,'C');
55 •    INSERT INTO SEMSEC VALUES ('CSE7A',7,'A');
56 •    INSERT INTO SEMSEC VALUES ('CSE7B',7,'B');
57 •    INSERT INTO SEMSEC VALUES ('CSE7C',7,'C');
```

```
54 •  INSERT INTO SEMSEC VALUES ('CSE8C',8,'C');
55 •  INSERT INTO SEMSEC VALUES ('CSE7A',7,'A');
56 •  INSERT INTO SEMSEC VALUES ('CSE7B',7,'B');
57 •  INSERT INTO SEMSEC VALUES ('CSE7C',7,'C');
58 •  INSERT INTO SEMSEC VALUES ('CSE6A',6,'A');
59 •  INSERT INTO SEMSEC VALUES ('CSE6B', 6,'B');
60 •  INSERT INTO SEMSEC VALUES ('CSE6C', 6,'C');
61 •  INSERT INTO SEMSEC VALUES ('CSE5A', 5,'A');
62 •  INSERT INTO SEMSEC VALUES ('CSE5B', 5,'B');
63 •  INSERT INTO SEMSEC VALUES ('CSE5C', 5,'C');
64 •  INSERT INTO SEMSEC VALUES ('CSE4A',4,'A');
65 •  INSERT INTO SEMSEC VALUES ('CSE4B', 4,'B');
66 •  INSERT INTO SEMSEC VALUES('CSE4C',4,'C');
67 •  INSERT INTO SEMSEC VALUES ('CSE3A', 3,'A');
68 •  INSERT INTO SEMSEC VALUES ('CSE3B', 3,'B');
69 •  INSERT INTO SEMSEC VALUES('CSE3C',3,'C');
70 •  INSERT INTO SEMSEC VALUES ('CSE2A', 2,'C');
71 •  INSERT INTO SEMSEC VALUES ('CSE2B', 2,'B');
72 •  INSERT INTO SEMSEC VALUES ('CSE2C', 2,'C');
73 •  INSERT INTO SEMSEC VALUES ('CSE1A', 1,'A');
74 •  INSERT INTO SEMSEC VALUES ('CSE1B', 1,'B');
75 •  INSERT INTO SEMSEC VALUES ('CSE1C', 1,'C');
```


```
78 •  INSERT INTO CLASS VALUES('1RN13CS062','CSE8A');
79 •  INSERT INTO CLASS VALUES('1RN13CS066','CSE8B');
80 •  INSERT INTO CLASS VALUES('1RN13CS091','CSE8C');
81 •  INSERT INTO CLASS VALUES('1RN14CS010','CSE7A');
82 •  INSERT INTO CLASS VALUES('1RN14CS025','CSE7A');
83 •  INSERT INTO CLASS VALUES('1RN14CS032','CSE7A');
84 •  INSERT INTO CLASS VALUES('1RN15CS011','CSE4A');
85 •  INSERT INTO CLASS VALUES('1RN15CS029','CSE4A');
86 •  INSERT INTO CLASS VALUES('1RN15CS045','CSE4B');
87 •  INSERT INTO CLASS VALUES('1RN15CS091','CSE4C');
88 •  INSERT INTO CLASS VALUES('1RN16CS045','CSE3A');
89 •  INSERT INTO CLASS VALUES('1RN16CS088','CSE3B');
90 •  INSERT INTO CLASS VALUES('1RN16CS122','CSE3C');
91
92 •  INSERT INTO SUBJECT VALUES ('10CS81','ACA', 8, 4);
93 •  INSERT INTO SUBJECT VALUES ('10CS82','SSM', 8, 4);
94 •  INSERT INTO SUBJECT VALUES ('10CS83','NM', 8, 4);
95 •  INSERT INTO SUBJECT VALUES ('10CS84','CC', 8, 4);
96 •  INSERT INTO SUBJECT VALUES ('10CS85','PW', 8, 4);
97 •  INSERT INTO SUBJECT VALUES ('10CS71','OOAD', 7, 4);
98 •  INSERT INTO SUBJECT VALUES ('10CS72','ECS', 7, 4);
99 •  INSERT INTO SUBJECT VALUES ('10CS73','PTW', 7, 4);
100 •  INSERT INTO SUBJECT VALUES ('10CS74','DWDM', 7, 4);
101 •  INSERT INTO SUBJECT VALUES ('10CS75','JAVA', 7, 4);
102 •  INSERT INTO SUBJECT VALUES ('10CS76','SAN', 7, 4);
```

```
99  INSERT INTO SUBJECT VALUES ('10CS73','PTW', 7, 4);
100 INSERT INTO SUBJECT VALUES ('10CS74','DWDM', 7, 4);
101 INSERT INTO SUBJECT VALUES ('10CS75','JAVA', 7, 4);
102 INSERT INTO SUBJECT VALUES ('10CS76','SAN', 7, 4);
103 INSERT INTO SUBJECT VALUES ('15CS51', 'ME', 5, 4);
104 INSERT INTO SUBJECT VALUES ('15CS52','CN', 5, 4);
105 INSERT INTO SUBJECT VALUES ('15CS53','DBMS', 5, 4);
106 INSERT INTO SUBJECT VALUES ('15CS54','ATC', 5, 4);
107 INSERT INTO SUBJECT VALUES ('15CS55','JAVA', 5, 3);
108 INSERT INTO SUBJECT VALUES ('15CS56','AI', 5, 3);
109 INSERT INTO SUBJECT VALUES ('15CS41','M4', 4, 4);
110 INSERT INTO SUBJECT VALUES ('15CS42','SE', 4, 4);
111 INSERT INTO SUBJECT VALUES ('15CS43','DAA', 4, 4);
112 INSERT INTO SUBJECT VALUES ('15CS44','MPMC', 4, 4);
113 INSERT INTO SUBJECT VALUES ('15CS45','OOC', 4, 3);
114 INSERT INTO SUBJECT VALUES ('15CS46','DC', 4, 3);
115 INSERT INTO SUBJECT VALUES ('15CS31','M3', 3, 4);
116 INSERT INTO SUBJECT VALUES ('15CS32','ADE', 3, 4);
117 INSERT INTO SUBJECT VALUES ('15CS33','DSA', 3, 4);
118 INSERT INTO SUBJECT VALUES ('15CS34','CO', 3, 4);
119 INSERT INTO SUBJECT VALUES ('15CS35','USP', 3, 3);
120 INSERT INTO SUBJECT VALUES ('15CS36','DMS', 3, 3);
```

```
21
22 INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)VALUES ('1RN13CS091','10CS81','CSE8C', 15, 16,18);
23 INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)VALUES ('1RN13CS091','10CS82','CSE8C', 12, 19,14);
24 INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)VALUES ('1RN13CS091','10CS83','CSE8C', 19, 15,20);
25 INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)VALUES ('1RN13CS091','10CS84','CSE8C', 20, 16,19);
26 INSERT INTO IAMARKS (USN, SUBCODE, SSID, TEST1, TEST2, TEST3)VALUES ('1RN13CS091','10CS85','CSE8C', 15, 15,12);
27 SELECT * FROM STUDENT;
28 SELECT * FROM SEMSEC;
29 SELECT * FROM CLASS;
30 SELECT * FROM SUBJECT;
31 SELECT * FROM IAMARKS;
```

```
132
133 SELECT S.*, SS.SEM, SS.SEC
134 FROM STUDENT S, SEMSEC SS, CLASS C
135 WHERE S.USN = C.USN AND
136 SS.SSID = C.SSID AND
137 SS.SEM = 4 AND
138 SS.SEc='C';
```

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

USN	SNAME	ADDRESS	PHONE	GENDER	SEM	SEC
1RN15CS091	SANTOSH	MANGALURU	8812332201	M	4	C

Result Grid

```
140 •  SELECT SS.SEM, SS.SEC, S.GENDER, COUNT(S.GENDER) AS COUNT
141    FROM STUDENT S, SEMSEC SS, CLASS C
142    WHERE S.USN = C.USN AND
143    SS.SSID = C.SSID
144    GROUP BY SS.SEM, SS.SEC, S.GENDER
145    ORDER BY SEM;
146
147 •  CREATE VIEW STU_TEST1_MARKS_VIEW
```

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

SEM	SEC	GENDER	COUNT
3	A	M	1
3	B	F	1
3	C	M	1
4	A	F	1
4	A	M	1
4	B	M	1
4	C	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

```
192
193    SELECT S.USN,S.SNAME,S.ADDRESS,S.PHONE,S.GENDER,
194    (CASE
195    WHEN IA.FINALIA BETWEEN 17 AND 20 THEN 'OUTSTANDING'
196    WHEN IA.FINALIA BETWEEN 12 AND 16 THEN 'AVERAGE'
197    ELSE 'WEAK'
198    END) AS CAT
199    FROM STUDENT S, SEMSEC SS, IAMARKS IA, SUBJECT SUB
200    WHERE S.USN = IA.USN AND
```

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

USN	SNAME	ADDRESS	PHONE	GENDER	CAT
1RN13CS091	TEESHA	BENGALURU	7712312312	F	WEAK
1RN13CS091	TEESHA	BENGALURU	7712312312	F	WEAK
1RN13CS091	TEESHA	BENGALURU	7712312312	F	WEAK
1RN13CS091	TEESHA	BENGALURU	7712312312	F	WEAK
1RN13CS091	TEESHA	BENGALURU	7712312312	F	WEAK