Program 6

```
create
schema
orderdb;
                create table salesman(
                salesmanid int,
                name varchar(200),
                city varchar(200),
                commision char(50),
                primary key(salesmanid)
                );


                create table customer(
                customerid int,
                cust_name varchar(200),
                city varchar(200),
                grade int,
                salesmanid int,
                primary key(customerid),
                foreign key (salesmanid) references salesman(salesmanid)
                );


                create table orders(
                orderno int,
                purch_amt int,
                orderdate date,
                customerid int,
                salesmanid int,
                primary key(orderno),
                foreign key (salesmanid) references salesman(salesmanid),
                foreign key (customerid) references customer(customerid)
                );
```

```
insert into salesman(salesmanid,name,city,commision) values
('1000','john','bangalore','25%'),
('2000','ravi','bangalore','20%'),
('3000','kumar','mysore','15%'),
('4000','smith','dehli','30%'),
('5000','harsha','hyderabad','15%');

insert into customer(customerid,cust_name,city,grade,salesmanid)
values
('10','preethi','bangalore','100','1000'),
('11','vivek','mangalore','300','1000'),
('12','bhaskar','chennai','400','2000'),
('13','chetan','bangalore','200','2000'),
('14','mamatha','bangalore','400','3000');

insert into
orders(orderno,purch_amt,orderdate,customerid,salesmanid)values
('50','5000','04-05-17','10','1000'),
('51','450','20-01-17','10','2000'),
('52','1000','23-02-17','13','2000'),
('53','3500','13-04-17','14','3000'),
('54','550','19-03-17','12','2000');
```

1. Count the customers with grades above Bangalore’s average.

```
SELECT grade, count(DISTINCT customerid)
FROM customer
GROUP BY grade
HAVING grade > (SELECT AVG(grade)
FROM customer
WHERE city='bangalore');
```

```
54 •  SELECT grade, count(DISTINCT customerid)
55    FROM customer
56    GROUP BY grade
57    HAVING grade > (SELECT AVG(grade)
58    FROM customer
59    WHERE city='bangalore');
```

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

grade	count(DISTINCT customerid)
300	1
400	2

2. Find the name and numbers of all salesmen who had more than one customer.

```
SELECT salesmanid, NAME
FROM salesman a
WHERE 1 < (SELECT count(*)
FROM customer
WHERE salesmanid=a.salesmanid);
```

```
63 •  SELECT salesmanid, NAME
64    FROM salesman a
65    WHERE 1 < (SELECT count(*)
66    FROM customer
67    WHERE salesmanid=a.salesmanid);
68
```

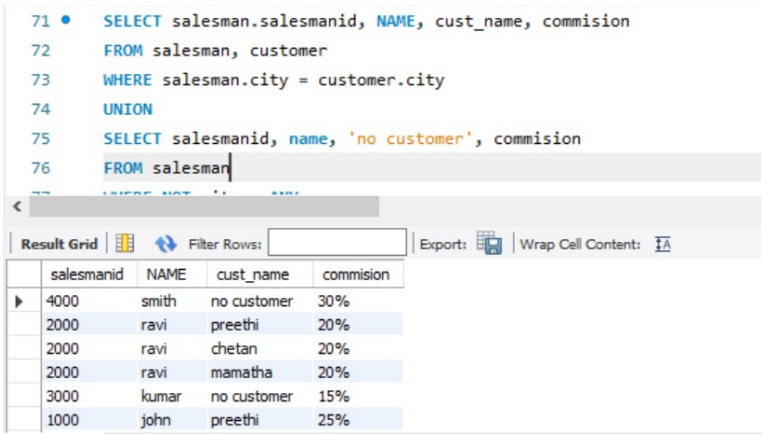
Result Grid | Filter Rows: | Edit:

salesmanid	NAME
1000	john
2000	ravi
NULL	NULL

3. List all salesmen and indicate those who have and don’t have customers in their cities (Use UNION operation.)

```
SELECT salesman.salesmanid, NAME, cust_name, commision
FROM salesman, customer
```

```
WHERE salesman.city = customer.city

UNION

SELECT salesmanid, name, 'no customer', commision

FROM salesman

WHERE NOT city = ANY

(SELECT city

FROM customer)

ORDER BY 2 DESC;
```



```
71 •  SELECT salesman.salesmanid, NAME, cust_name, commision
72    FROM salesman, customer
73    WHERE salesman.city = customer.city
74    UNION
75    SELECT salesmanid, name, 'no customer', commision
76    FROM salesman
```

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

salesmanid	NAME	cust_name	commision
4000	smith	no customer	30%
2000	ravi	preethi	20%
2000	ravi	chetan	20%
2000	ravi	mamatha	20%
3000	kumar	no customer	15%
1000	john	preethi	25%

4. Create a view that finds the salesman who has the customer with the highest order of a day.

```
CREATE VIEW highsalesman AS

SELECT b.orderdate, a.salesmanid, a.name

FROM salesman a, orders b

WHERE a.salesmanid = b.salesmanid

AND b.purch_amt=(SELECT max(purch_amt)

FROM orders c

WHERE c.orderdate = b.orderdate);

SELECT * FROM highsalesman;
```

Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted

```
DELETE FROM salesman
```

```
 WHERE salesmanid=1000;
```

Program 7

```
create
database
Lab7;
              create table publisher (
                     name varchar (20) primary key,
                     phone integer,
                     address varchar (20)
              );
              desc publisher;


              create table book (
                     book_id integer primary key,
                     title varchar (20),
                     pub_year varchar (20),
                     publisher_name varchar (20),
                     foreign key (publisher_name) references publisher (name) on
              delete cascade
              );
              desc book;


              create table book_authors (
                     author_name varchar (20),
                     book_id integer,
                     foreign key (book_id) references book (book_id) on delete
              cascade,
                     primary key (book_id, author_name)
              );
              desc book_authors;
```

```
create table library_branch (
       branch_id integer primary key,
       branch_name varchar (50),
       address varchar (50)
);
desc library_branch;


create table book_copies (
       no_of_copies integer,
       book_id integer,
       branch_id integer,
       foreign key (book_id) references book (book_id) on delete
cascade,
       foreign key (branch_id) references library_branch (branch_id)
on delete cascade,
       primary key (book_id, branch_id)
);
desc book_copies;


create table card (
       card_no integer primary key
);
desc card;


create table book_lending (
       date_out date,
       due_date date,
   book_id integer,
       branch_id integer,
   card_no integer,
```

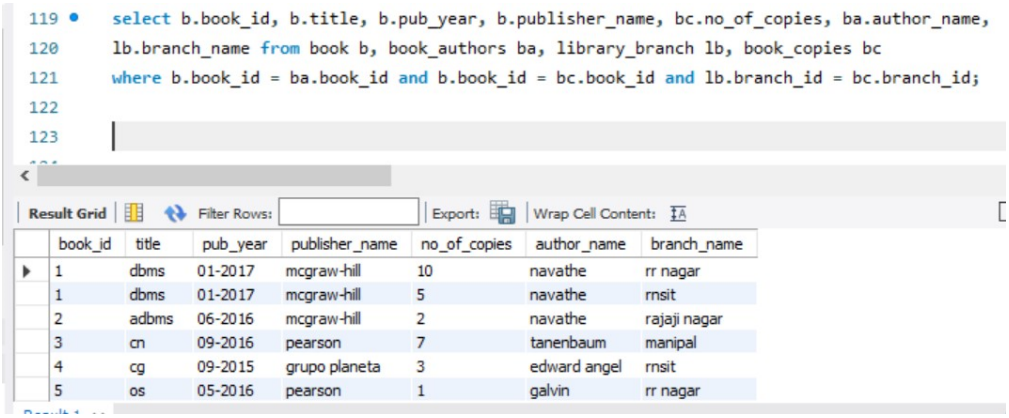
```
      foreign key (book_id) references book (book_id) on delete
cascade,
      foreign key (branch_id) references library_branch (branch_id)
on delete cascade,
      foreign key (card_no) references card (card_no) on delete
cascade,
      primary key (book_id, branch_id, card_no)
);
desc book_lending;


insert into publisher(name,phone,address) values
('mcgraw-hill', 99890, 'bangalore'),
('pearson', 98890, 'newdelhi'),
('random house', 74556, 'hyderabad'),
('hachette livre', 897086, 'chenai'),
('grupo planeta', 77561, 'bangalore');
select * from publisher;


insert into book (book_id,title,pub_year,publisher_name)values
(1,'dbms','01-2017', 'mcgraw-hill'),
(2,'adbms','06-2016', 'mcgraw-hill'),
(3,'cn','09-2016', 'pearson'),
(4,'cg','09-2015', 'grupo planeta'),
(5,'os','05-2016', 'pearson');
select * from book;


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

```
select * from book_authors;


insert into library_branch (branch_id,branch_name,address)values
(10,'rr nagar','bangalore'),
(11,'rnsit','bangalore'),
(12,'rajaji nagar', 'bangalore'),
(13,'nitte','mangalore'),
(14,'manipal','udupi');
select * from library_branch;


insert into book_copies(no_of_copies,book_id,branch_id) values
(10, 1, 10),
(5, 1, 11),
(2, 2, 12),
(7, 3, 14),
(1, 5, 10),
(3, 4, 11);
select * from book_copies;


insert into card(card_no) values
(100),
(101),
(102),
(103),
(104);
select * from card;


insert into book_lending(date_out,due_date,book_id,branch_id,card_no)
values
('01-01-17','01-06-17', 1, 10, 101),
```

```
('11-01-17','11-03-17', 3, 14, 101),
('21-02-17','21-04-17', 2, 13, 101),
('15-03-17','15-07-17', 4, 11, 101),
('12-08-17','12-08-17', 1, 11, 104);
select * from book_lending;
```

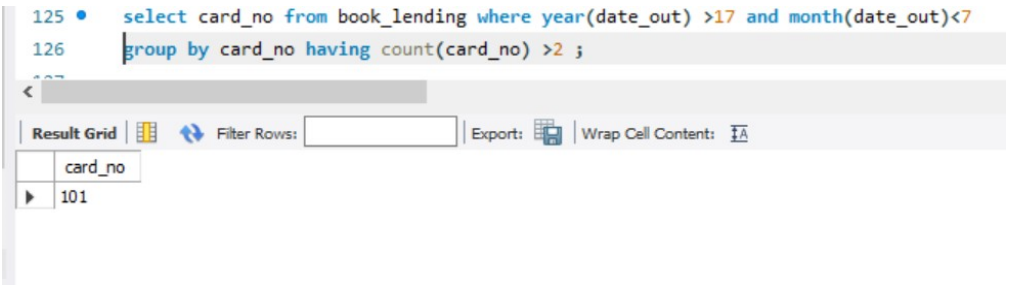
Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc.

```
select b.book_id, b.title, b.pub_year, b.publisher_name,
bc.no_of_copies, ba.author_name,
lb.branch_name from book b, book_authors ba, library_branch lb,
book_copies bc
where b.book_id = ba.book_id and b.book_id = bc.book_id and
lb.branch_id = bc.branch_id;
```



Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017

```
select card_no from book_lending where year(date_out) >17 and
month(date_out)<7
group by card_no having count(card_no) >2 ;
```

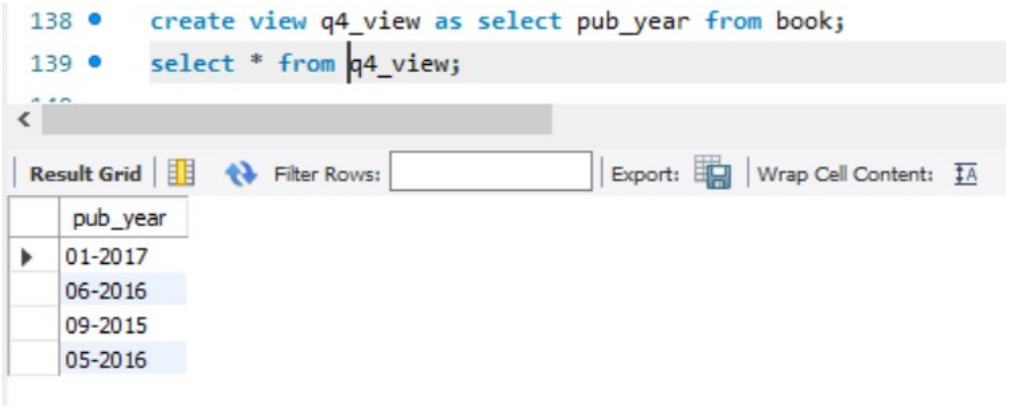


Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.

```
delete from book where book_id = 3;

select * from book;

select * from book_authors;

select * from book_copies;

select * from book_lending;
```

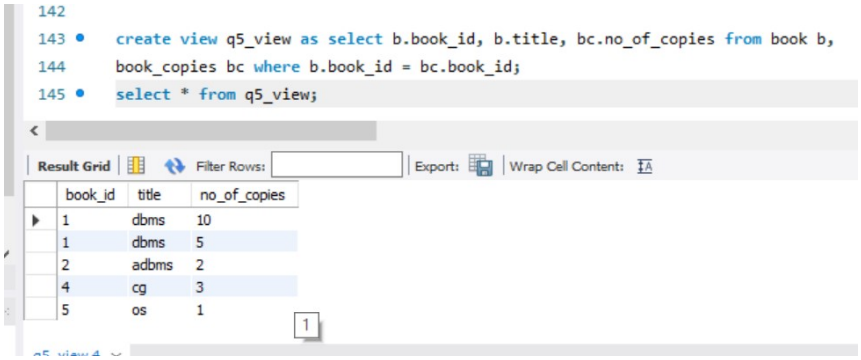
Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.

```
create view q4_view as select pub_year from book;

select * from q4_view;
```



Create a view of all books and its number of copies that are currently available in the Library.

```
create view q5_view as select b.book_id, b.title, bc.no_of_copies
from book b,
book_copies bc where b.book_id = bc.book_id;

select * from q5_view;
```



```
142
143 • create view q5_view as select b.book_id, b.title, bc.no_of_copies from book b,
144   book_copies bc where b.book_id = bc.book_id;
145 • select * from q5_view;
```

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

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

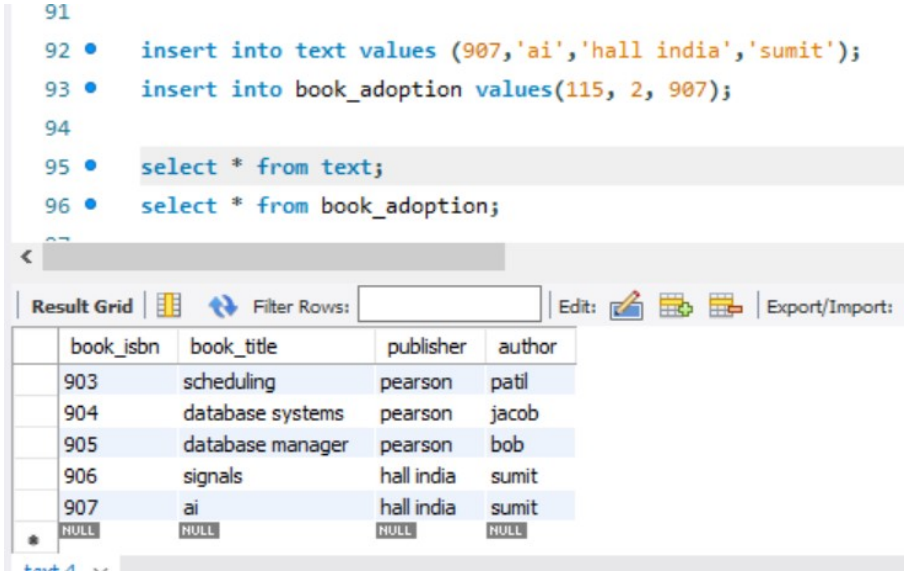
Program 8

```
create database
textcourse;
                    use textcourse;


                    create table student(
                        regno varchar(15),
                        name varchar(20),
                        major varchar(20),
                        bdate date,
                        primary key (regno)
                        );
                    desc student;
                    create table course(
                        courseno int,
                        cname varchar(20),
                        dept varchar(20),
                        primary key (courseno)
                        );
                    desc course;
                    create table enroll(
                        regno varchar(15),
                        courseno int,
```

```
    sem int(3),
    marks int(4),
    primary key (regno,courseno),
    foreign key (regno) references student (regno),
    foreign key (courseno) references course (courseno)
    );
desc enroll;
create table text(
    book_isbn int(5),
    book_title varchar(20),
    publisher varchar(20),
    author varchar(20),
    primary key (book_isbn)
    );
desc text;


create table book_adoption(
    courseno int,
    sem int(3),
    book_isbn int(5),
    primary key (courseno,book_isbn),
    foreign key (courseno) references course (courseno),
    foreign key (book_isbn) references text(book_isbn)
    );
desc book_adoption;
insert into student (regno,name,major,bdate) values
    ('1pe11cs002','b','sr','19930924'),
    ('1pe11cs003','c','sr','19931127'),
    ('1pe11cs004','d','sr','19930413'),
    ('1pe11cs005','e','jr','19940824');
select * from student;
```

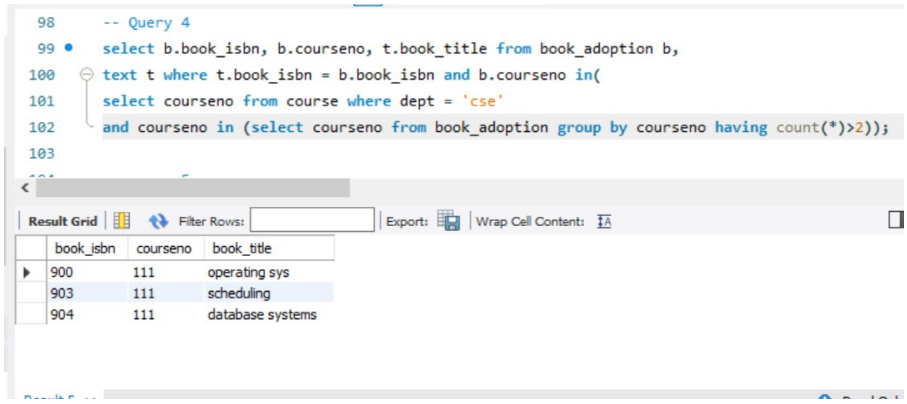
```
insert into course values (111,'os','cse'),
    (112,'ec','cse'),
    (113,'ss','ise'),
    (114,'dbms','cse'),
    (115,'signals','ece');
select * from course;
insert into text values
(book_isbn,book_title,publisher,author),
    (10,'database systems','pearson','schield'),
    (900,'operating sys','pearson','leland'),
    (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');
select * from text;


insert into enroll (regno,courseno,sem,marks) values
    ('1pe11cs002',114,5,100),
    ('1pe11cs003',113,5,100),
    ('1pe11cs004',111,5,100),
    ('1pe11cs005',112,3,100);
select * from enroll;


insert into book_adoption (courseno,sem,book_isbn) 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 book_adoption;

-- Query 3

insert into text values (907,'ai','hall india','sumit');
insert into book_adoption values(115, 2, 907);

select * from text;
select * from book_adoption;
```

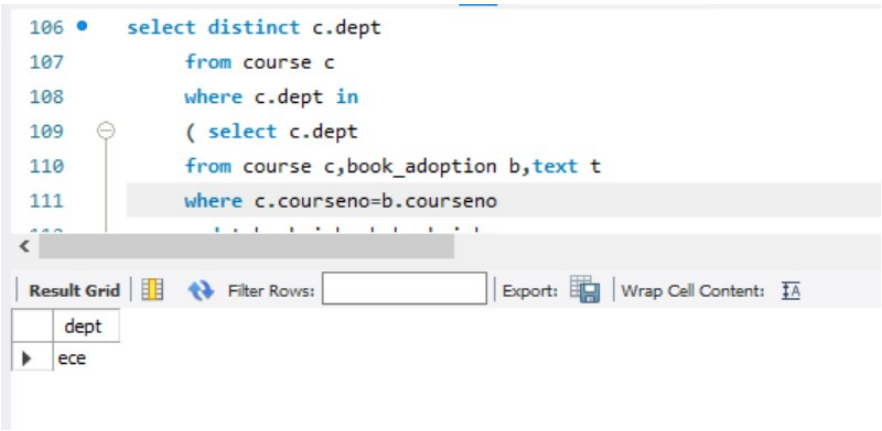



```
-- Query 4

select b.book_isbn, b.courseno, t.book_title from
book_adoption b,
text t where t.book_isbn = b.book_isbn and b.courseno in(
```

```
select courseno from course where dept = 'cse'
and courseno in (select courseno from book_adoption group by
courseno having count(*)>2));
```




```
-- query 5


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='hall india')
    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 != 'hall india');
```

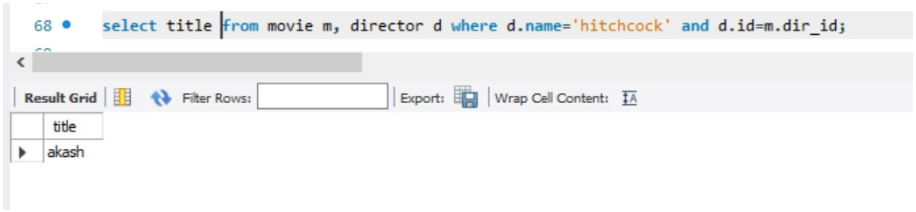


Program 9

```
create database
moviesdb;
                    use moviesdb;


                    create table actor (
                           id int primary key,
                           name varchar(10),
                           gender varchar(1)
                    );
                    desc actor;
                    create table director (
                           id int primary key,
                           name varchar(10),
                           phone varchar(10)
                    );
                    desc director;
                    create table movie (
                           id int primary key,
                           title varchar(20),
                           year int,
                           lang varchar(10),
```

```
	dir_id int,
	foreign key(dir_id) references director(id)
);
desc movie;
create table cast (
	act_id int,
	mov_id int,
	role varchar(20),
	primary key(act_id, mov_id),
	foreign key(act_id) references actor(id),
	foreign key(mov_id) references movie(id)
);
desc cast;
create table rating(
	mov_id int,
	stars int,
	foreign key(mov_id) references movie(id),
	check (stars > 0 and stars <= 5)
);
desc rating;
insert into actor values (301,'anushka','f'),
(302,'prabhas','m'),
(303,'punith','m'),
(304,'jermy','m');
select * from actor;
insert into director values (60,'rajamouli',8751611001),
(61,'hitchcock',7766138911),
(62,'faran',9986776531),
(63,'spielberg',8989776530);
select * from director;
insert into movie values (1001,'bahubali-2',2017,'telugu',60),
(1002,'bahubali-1',2015,'telugu',60),
```

```
(1003,'akash',2008,'kannada',61),

(1004,'war horse',2011,'english',63);

select * from movie;

insert into cast values (301,1002, 'heroine'),

(301,1001,'heroine'),(303,1003,'hero'),(303,1002,'guest'),(304,
1004,'hero');
select * from cast;

insert into rating values

(1001,4),

(1002,2),

(1003,5),

(1004,4);

select * from rating;


-- Query 1


select title from movie m, director d where d.name='hitchcock'
and d.id=m.dir_id;
```




```
-- Query 2


select distinct title from movie m, cast c where m.id=c.mov_id

    and c.act_id in (select act_id from cast group by act_id
having count(mov_id) > 1);
```

```
72 •  select distinct title from movie m, cast c where m.id=c.mov_id
73        and c.act_id in (select act_id from cast group by act_id having count(mov_id) > 1);
```

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

title
bahubali-2
bahubali-1
akash

```
-- Query 3

select distinct name from actor a inner join

  cast c on a.id=c.act_id and c.mov_id in

              (select id from movie m where year not between
2000 and 2015);
```

```
80 •  select distinct name from actor a inner join
81        cast c on a.id=c.act_id and c.mov_id in
82             (select id from movie m where year not between 2000 and 2015);
```

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

name
anushka

```
-- Query 4


select title, stars from movie m

   inner join (select mov_id, max(stars) as stars from rating
group by mov_id) r
      on m.id=r.mov_id order by title;
```

```
select title, stars from movie m
    inner join (select mov_id, max(stars) as stars from rating group by mov_id) r
      on m.id=r.mov_id order by title;
```

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

title	stars
akash	5
bahubali-1	2
bahubali-2	4
war horse	4

```
-- Query 5
update rating set stars=5 where mov_id in
(select m.id from movie m, director d where m.dir_id = d.id and
d.name='spielberg');
select * from rating;
```

```
update rating set stars=5 where mov_id in
(select m.id from movie m, director d where m.dir_id = d.id and d.name='spielberg');
select * from rating;
```

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

mov_id	stars
1001	4
1002	2
1003	5
1004	5

Program 10

```
create
database
lab10;
                use lab10;


                create table student(
                 usn varchar(30),
                 sname varchar(30),
                 address varchar(30),
                 phone real,
```

```
 gender varchar(30),
 primary key(usn)
 );
desc student;
create table semsec(
 ssid varchar(30),
 sem int,
 sec varchar(30),
 primary key(ssid)
 );
desc semsec;
create table class(
       usn varchar(30),
       ssid varchar(30),
       primary key(usn,ssid),
       foreign key(usn) REFERENCES student(usn),
       foreign key(ssid) REFERENCES semsec(ssid)
);
desc class;
create table subject(
  code varchar(30),
  title varchar(30),
  sem int,
  credits int,
  primary key(code)
);
desc subject;
create table marks(
       usn varchar(30),code varchar(30),
       ssid varchar(30),
       test1 real, test2 real, test3 real, final real,
       primary key(usn,code,ssid),
```

```
       foreign key(usn) REFERENCES student(usn),
       foreign key(code) REFERENCES subject(code),
       foreign key(ssid) REFERENCES semsec(ssid)
);
desc marks;
insert into student
values('1RN13CS020','akshay','belagavi',8877881122,'m'),
('1RN13CS062','sandhya','bengaluru',7722829912,'f'),
('1RN13CS091','teesha','bengaluru',7712312312,'f'),
('1RN13CS066','supriya','mangaluru',8877881122,'f'),
('1RN14CS010','abhay','bengaluru',9900211201,'m'),
('1RN14CS032','bhaskar','bengaluru',9923211099,'m'),
('1RN14CS025','asmi','bengaluru',7894737377,'f'),
('1RN15CS011','ajay','tumkur',98545091341,'m'),
('1RN15CS029','chitra','davangere',7696772121,'f'),
('1RN15CS045','jeeva','bellary',9944850121,'m'),
('1RN15CS091','santosh','mangaluru',8812332201,'m'),
('1RN16CS045','ismail','kalburgi',9900232201,'m'),
('1RN16CS088','sameera','shimoga',9905542212,'f'),
('1RN16CS122','vinayaka','chikamagaluru',8800880011,'m');


insert into semsec values('CSE8A',8,'A'),
('CSE8B',8,'B'),('CSE8C',8,'C'),
('CSE7A',7,'A'),('CSE7B',7,'B'),('CSE7C',7,'C'),
('CSE6A',6,'A'),('CSE6B',6,'B'),('CSE6C',6,'C'),
('CSE5A',5,'A'),('CSE5B',5,'B'),('CSE5C',5,'C'),
('CSE4A',4,'A'),('CSE4B',4,'B'),('CSE4C',4,'C'),
('CSE3A',3,'A'),('CSE3B',3,'B'),('CSE3C',3,'C'),
('CSE2A',2,'A'),('CSE2B',2,'B'),('CSE2C',2,'C'),
('CSE1A',1,'A'),('CSE1B',1,'B'),('CSE1C',1,'C');
```

```
insert into class values('1RN13CS020','CSE8A'),
('1RN13CS062','CSE8A'),('1RN13CS066','CSE8B'),('1RN13CS091','CSE8C
'),
('1RN14CS010','CSE7A'),('1RN14CS025','CSE7A'),('1RN14CS032','CSE7A
'),
('1RN15CS011','CSE4A'),('1RN15CS029','CSE4A'),('1RN15CS045','CSE4B
'),
('1RN15CS091','CSE4C'),('1RN16CS045','CSE3A'),('1RN16CS088','CSE3B
'),
('1RN16CS122','CSE3C');


insert into subject values('10CS81','ACA',8,4),
('10CS82','SSM',8,4),('10CS83','NM',8,4),
('10CS84','CC',8,4),('10CS85','PW',8,4),
('10CS71','OOAD',7,4),('10CS72','ECS',7,4),
('10CS73','PTW',7,4),('10CS74','DWDM',7,4),
('10CS75','JAVA',7,4),('10CS76','SAN',7,4),
('10CS51','ME',5,4),('10CS52','CN',5,4),
('10CS53','DBMS',5,4),('10CS54','ATC',5,4),
('10CS55','JAVA',5,3),('10CS56','AI',5,3),
('10CS41','M4',4,4),('10CS42','SE',4,4),
('10CS43','DAA',4,4),('10CS44','MPMC',4,4),
('10CS45','OOC',4,3),('10CS46','DC',4,3),
('10CS31','M3',3,4),('10CS32','ADE',3,4),
('10CS33','DSA',3,4),('10CS34','CO',3,4),
('10CS35','USp',3,3),('10CS36','DMS',3,3);


insert into marks(usn,code,ssid,test1,test2,test3)
values('1RN13CS091','10CS81','CSE8C',15,16,18),
('1RN13CS091','10CS82','CSE8C',12,19,14),('1RN13CS091','10CS83','C
SE8C',19,15,20),
('1RN13CS091','10CS84','CSE8C',20,16,19),('1RN13CS091','10CS85','C
SE8C',15,15,12);
```

```
select * from student;

select * from class;

select * from subject;

select * from semsec;

select * from marks;
```

```
-- Query 1

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';
```

```
104     -- Query 1
105 •   select S.*, SS.sem, SS.sec
106     from student S, semsec SS, class C
107     where S.usn = C.usn AND SS.ssid = C.ssid AND SS.sem = 4 AND SS.sec = 'C';
108
```

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

usn	sname	address	phone	gender	sem	sec
1RN15CS091	santosh	mangaluru	8812332201	m	4	C

```
-- Query 2

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;
```

```
110 •  select SS.sem, SS.sec, S.gender, count(S.gender) as COUNT
111    from student S, semsec SS, class C
112    where S.usn = C.usn AND SS.ssid = C.ssid
113    group by SS.sem, SS.sec, S.gender ORDER by sem;
```

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

```
-- Query 3

create view STU_test1_marks_view as

select test1, code

from marks

where usn = '1RN13CS091';

select * from STU_test1_marks_view;
```

```
116 •  create view STU_test1_marks_view as
117    select test1, code
118    from marks
119    where usn = '1RN13CS091';
120 •  select * from STU_test1_marks_view;
```

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

test1	code
15	10CS81
12	10CS82
19	10CS83
20	10CS84
15	10CS85

```
-- Query 4

select S.usn, S.sname, S.address, S.phone, S.gender,
```

```
(CASE
when IA.final between 17 and 20 then 'outstanding'
when IA.final between 12 and 16 then 'average'
else 'weak' end) AS CAT
from student S, semsec SS, marks IA, subject sub
where S.usn = IA.usn AND SS.ssid = IA.ssid AND sub.code = IA.code
AND sub.sem = 8;
```