**Experiment 1:**

1)select B.book\_id, B.title, B.publisher\_name, A.author\_name, C.no\_of\_copies, C.branch\_id from Book B, BOOK\_AUTHORS A, BOOK\_COPIES C, LIBRARY\_BRANCH L where B.book\_id = A.book\_id and B.book\_id = C.book\_id and L.branch\_id = C.branch\_id;

2)select card\_no BOOK\_LENDING where date\_out between '201701-01' And '2017-07-01' Group by card\_no Having count(\*) > 3;

3)delete from BOOK where book\_id = 3;

select \* from BOOK;

4)create view v\_pub as select distinct(pub\_year) from BOOK;

select \* from v\_pub;

5)create view V\_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 V\_books;

**Experiment 2:**

1)select grade , count(customer\_id) from CUSTOMER1 group by grade having grade > (select avg(grade) from CUSTOMER1 where city = 'BANGALORE');

2) select s.salesman\_id, s.name from SALESMAN s where salesman\_id IN (select salesman\_id from CUSTOMER1 c1 group by c1.salesman\_id having count(\*)>1);

3) select s.salesman\_id, s.name, c.cust\_name from SALESMAN s, CUSTOMER1 c where s.city = c.city and s.salesman\_id = c.salesman\_id UNION select s.salesman\_id, s.name , c.cust\_name from SALESMAN s, CUSTOMER1 c where s.city!=c.city and s.salesman\_id = c.salesman\_id;

4)create view ess\_salesman as select b.ord\_date, s.salesman\_id, s.name from ORDERS b, SALESMAN s where s.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 ess\_salesman;

5) delete from SALESMAN where salesman\_id = 1000;

select \* from SALESMAN;

**Experiment 3:**

1. select mov\_title from MOVIES where dir\_id IN (select dir\_id from DIRECTOR where dir\_name = 'HITCHCOCK');
2. select mov\_title from MOVIES M , MOVIE\_CAST MC where M.mov\_id = MC.mov\_id AND MC.act\_id IN (select act\_id from MOVIE\_CAST group by act\_id having count(act\_id)>1);
3. select act\_name, mov\_title, mov\_year from ACTOR A JOIN movie\_cast C on A.act\_id = C.act\_id JOIN MOVIES M ON C.mov\_id = M.mov\_id where M.mov\_year NOT BETWEEN 2000 and 2015;
4. 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;
5. 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;

**Experiment 4:**

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';
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 ss.sem;
3. create view STU\_TEST\_MARKS as select test1,subcode from IAMARKS where usn='1RN13CS091';

select \* from STU\_TEST\_MARKS;

1. Update IAMARKS set FINALIA=GREATEST (TEST1+TEST2, TEST2+TEST3, TEST1+TEST3)/2; SELECT \* FROM IAMARKS;
2. 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, iamarks ia, subject sub, semsec ss where s.usn = ia.usn and ss.ssid = ia.ssid and sub.subcode = ia.subcode and sub.sem = 8;