SQL PROGRAMMING QUERIES

1.LIBRARY DATABASE:

QUERY 1:

```
select b.book_id,b.title,b.pub_name,a.author_name,c.no_of_copies,c.branch_id
from book b,book_author a,book_copies c
where b.book_id=a.book_id and
b.book_id=c.book_id;
```

QUERY 2:

```
select card_no ,DATE_OUT,DUE_DATE from book_lending
where date_out between '2019-01-01' and '2019-06-30'
group by card_no
having count(*)>3;
```

QUERY 3:

```
delete from book where book_id=1001;
```

QUERY 4:

```
MariaDB [lib]> CREATE VIEW V_PUBLICATION AS SELECT PUB_YEAR FROM BOOK;
MariaDB [lib]> select * from v_publication;
```

QUERY 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;
```

2.SALES ORDER DATABASE:

QUERY 1:

```
SELECT GRADE, COUNT(DISTINCT CUSTOMER_ID) FROM CUSTOMER
GROUP BY GRADE
HAVING GRADE > (SELECT AVG(GRADE)FROM CUSTOMER WHERE CITY='BANGALORE');
```

QUERY 2:

```
SELECT S.SALESMAN_ID, S.NAME FROM SALESMAN S
WHERE 1< (SELECT COUNT(*) FROM CUSTOMER C WHERE C.SALESMAN_ID=S.SALESMAN_ID);
```

QUERY 3:

```
SELECT S.SALESMAN_ID, NAME, CUST_NAME, COMMISSION FROM SALESMAN S, CUSTOMER C
WHERE S.CITY=C.CITY
UNION
SELECT SALESMAN_ID,NAME,'NO MATCH',COMMISSION
FROM SALESMAN WHERE CITY NOT IN (SELECT CITY FROM CUSTOMER);
```

QUERY 4:

```
CREATE VIEW HIGHEST_ORDER AS
SELECT O.ORD_DATE,S.SALESMAN_ID,S.NAME
FROM SALESMAN S,ORDERS O
WHERE S.SALESMAN_ID=O.SALESMAN_ID
AND
 O.PURCHASE_AMT=(SELECT MAX(PURCHASE_AMT) FROM ORDERS OD  WHERE
O.ORD_DATE=OD.ORD_DATE);
```

QUERY 5:

```
DELETE FROM SALESMAN WHERE SALESMAN_ID=1000;
```

3.MOVIE DATABASE:

QUERY 1:

```
SELECT MOV_TITLE
FROM MOVIES
WHERE DIR_ID IN (SELECT DIR_ID FROM DIRECTOR WHERE DIR_NAME = 'ABCD');
```

QUERY 2:

```
SELECT MOV_TITLE
FROM MOVIES M, MOVIE_CAST C
WHERE M.MOV_ID=C.MOV_ID AND C.ACT_ID IN (SELECT ACT_ID
FROM MOVIE_CAST
GROUP BY ACT_ID
HAVING COUNT(act_ID)>0)
GROUP BY MOV_TITLE
HAVING COUNT(*)>1;
```

QUERY 3:

```
SELECT A.ACT_NAME
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<2000
AND EXISTS(
SELECT A.ACT_NAME
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>2015);
```

QUERY 4:

```
SELECT M.MOV_TITLE, MAX(R.REV_STARS) AS HIGHEST_RATING
FROM MOVIES M,RATING R
WHERE M.MOV_ID=R.MOV_ID
GROUP BY M.MOV_TITLE
HAVING COUNT(R.REV_STARS)>0
ORDER BY M.MOV_TITLE;
```

QUERY 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 = 'XYZ'));
```

4.COLLEGE DATABASE:

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';
```

QUERY 2:

```
SELECT SS.SEM, SS.SEC, S.GENDER, COUNT(S.GENDER) AS GENDER_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;
```

QUERY 3:

```
CREATE VIEW STU_TEST1 AS
SELECT TEST1, SUBCODE
FROM IAMARKS
 WHERE USN='11XX1234';
```

QUERY 4:

```
update iamarks set finalia=(test1+test2+test3-least(test1,test2,test3))/2;
select * from iamarks;
```

QUERY 5:

```
SELECT S.USN,S.SNAME,SS.SEM,SS.SEC,SUB.TITLE,IA.TEST1,IA.TEST2,IA.TEST3,IA.FINALIA,
(
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,SUBJECT SUB, IAMARKS IA
 WHERE S.USN=IA.USN AND SS.SSID=IA.SSID AND SUB.SUBCODE=IA.SUBCODE AND SS.SEM=8  AND
SS.SEC IN('A','B','C');
```

5.COMPANY DATABASE:

QUERY 1:

```
(SELECT DISTINCT P.PNO
FROM PROJECT P, DEPARTMENT D, EMPLOYEE E
WHERE P.DNO=D.DNO
AND D.MGRSSN=E.SSN
AND E.LNAME='SOCTT')
UNION
(SELECT DISTINCT P1.PNO
 FROM PROJECT P1, WORKS_ON W, EMPLOYEE E1
 WHERE P1.PNO=W.PNO
 AND E1.SSN=W.SSN
AND E1.LNAME='SOCTT');
```

QUERY 2:

```
SELECT E.FNAME, E.LNAME, 1.1*E.SALARY AS INCR_SAL FROM
EMPLOYEE E, WORKS_ON W, PROJECT P
WHERE E.SSN=W.SSN
AND W.PNO=P.PNO
AND P.PNAME='IOT';
```

QUERY 3:

```
SELECT SUM(E.SALARY)AS TOTAL_SALARY, MAX(E.SALARY)AS MAXIMUM, MIN(E.SALARY) AS
MINIMUM, AVG(E.SALARY) AS AVERAGE
FROM EMPLOYEE E, DEPARTMENT D
WHERE E.DNO=D.DNO
AND D.DNAME='ACCOUNTS';
```

QUERY 4:

```
SELECT E.FNAME, E.LNAME
FROM EMPLOYEE E
 WHERE NOT EXISTS
 (SELECT PNO FROM PROJECT P WHERE DNO=5
 AND P.PNO NOT IN(SELECT PNO FROM WORKS_ON WHERE E.SSN=SSN));
```

QUERY 5:

```
 SELECT E.DNO, COUNT(*) AS NO_OF_EMPLOYEE
 FROM  EMPLOYEE E
WHERE E.SALARY>600000
AND E.DNO IN (SELECT E1.DNO
FROM EMPLOYEE E1
 GROUP BY E1.DNO
HAVING COUNT(*)>5)
 GROUP BY E.DNO;
```