Queries:

Sql-1

1.

```
SELECT B.BOOK_ID, B.TITLE, B.PUBLISHER_NAME, A.AUTHOR_NAME,
C.NO_OF_COPIES, L.PROGRAMME_ID FROM BOOK B, BOOK_AUTHORS
A, BOOK_COPIES
C, LIBRARY_PROGRAMME L WHERE B.BOOK_ID=A.BOOK_ID AND
B.BOOK_ID=C.BOOK_ID AND L.PROGRAMME_ID=C.PROGRAMME_ID;
```

2.

```
SELECT CARD_NO FROM BOOK_LENDING WHERE DATE_OUT
BETWEEN '2017-01-01'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 VW_PUBLICATION AS SELECT PUB_YEAR FROM BOOK;
SELECT * FROM VW_PUBLICATION;
```

5.

```
CREATE VIEW VW_BOOKS AS SELECT B.BOOK_ID, B.TITLE,
C.NO_OF_COPIES
FROM BOOK B, BOOK_COPIES C, LIBRARY_PROGRAMME L WHERE
B.BOOK_ID=C.BOOK_ID AND C.PROGRAMME_ID=L.PROGRAMME_ID;
SELECT * FROM VW_BOOKS;
```

Sql-2

1.

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

2.

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

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 NOT CITY = ANY
(SELECT
CITY FROM CUSTOMER) ORDER BY 2 DESC;
```

4.

```
CREATE VIEW VW_ELITSALESMAN 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 VW_ELITSALESMAN;
```

5.

```
DELETE FROM SALESMAN WHERE SALESMAN_ID=101;
```

Sql-3

1.

```
SELECT MOV_TITLE FROM MOVIES M, DIRECTOR D WHERE
M.DIR_ID=D.DIR_ID
AND DIR_NAME='HITCHCOCK';
```

2.

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

3.

```
SELECT ACT_NAME, MOV_TITLE, MOV_YEAR FROM ACTOR A JOIN
MOVIES_CAST C ON
A.ACT_ID=C.ACT_ID INNER 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 M ,RATING R
WHERE
M.MOV_ID=R.MOV_ID GROUP BY MOV_TITLE HAVING
MAX(REV_STARS)>0 ORDER BY
MOV_TITLE;
```

5.

```
UPDATE RATING R, MOVIES M, DIRECTOR D SET REV_STARS=5 WHERE
R.MOV_ID=M.MOV_ID AND
    M.DIR_ID=D.DIR_ID AND DIR_NAME='STEVEN SPIELBER';
```

Sql-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
    JOIN CLASS C ON S.USN = C.USN
    JOIN SEMSEC SS ON SS.SSID = C.SSID
    GROUP BY SS.SEM, SS.SEC, S.GENDER
    ORDER BY SS.SEM;
```

3.

```
 CREATE VIEW VW_STUDENT_TEST AS SELECT TEST1,SUBCODE FROM
IAMARKS WHERE USN= '4AD13CS091';
SELECT * FROM VW_STUDENT_TEST;
```

4.

```
 UPDATE IAMARKS
    SET FINALIA=GREATEST(TEST1+TEST2,TEST2+TEST3,TEST1+TEST3)/2;
 SELECT * FROM IAMARKS;
```

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;
```

Sql-5

1.

```
(SELECT DISTINCT P.PNO
    FROM PROJECT P, DEPARTMENT D, EMPLOYEE E
    WHERE E.DNO = D.DNO
    AND D.MGRSSN = E.SSN
    AND E.LNAME = 'budke')
    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 = 'budke');
```

2.

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

3.

```
SELECT SUM(E.SALARY) AS TOTAL_SALARY,  MAX(E.SALARY) AS
MAX_SALARY,        MIN(E.SALARY) AS MIN_SALARY,   AVG(E.SALARY) AS
AVG_SALARY FROM DEPARTMENT D LEFT JOIN EMPLOYEE E ON D.DNO
=
E.DNO WHERE D.DNAME = 'IT';
```

4.

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

5.

```
SELECT D.DNO, COUNT(*) FROM DEPARTMENT D, EMPLOYEE E WHERE
D.DNO=E.DNO AND E.SALARY>6000 AND D.DNO IN (SELECT E1.DNO
FROM EMPLOYEE E1  GROUP BY E1.DNO   HAVING COUNT(*)>1) GROUP
BY D.DNO;
```