**MOVIE DATABASE**

Consider the schema for Movie Database:

ACTOR(Act\_id, Act\_Name, Act\_Gender)

DIRECTOR(Dir\_id, Dir\_Name, Dir\_Phone)

MOVIES(Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id)

MOVIE\_CAST(Act\_id, Mov\_id, Role)

RATING(Mov\_id, Rev\_Stars)

CREATE TABLE ACTOR (ACT\_ID INTEGER PRIMARY KEY,ACT\_NAME VARCHAR2(20),ACT\_GENDER CHAR(1));

CREATE TABLE DIRECTOR (DIR\_ID INTEGER PRIMARY KEY,DIR\_NAME VARCHAR2(20),DIR\_PHONE NUMBER(10));

CREATE TABLE MOVIES (MOV\_ID INTEGER PRIMARY KEY,MOV\_TITLE VARCHAR2(20),MOV\_YEAR NUMBER (4),MOV\_LANG VARCHAR2(20),DIR\_ID REFERENCES DIRECTOR (DIR\_ID) ON DELETE CASCADE);

CREATE TABLE MOVIE\_CAST (ACT\_ID REFERENCES ACTOR (ACT\_ID) ON DELETE CASCADE,MOV\_ID REFERENCES MOVIES (MOV\_ID) ON DELETE CASCADE,ROLE VARCHAR2(20),PRIMARY KEY (ACT\_ID,MOV\_ID));

CREATE TABLE RATING (MOV\_ID REFERENCES MOVIES (MOV\_ID) ON DELETE CASCADE,REV\_STARS INTEGER,PRIMARY KEY (MOV\_ID));

**INSERTING VALUES**

INSERT INTO ACTOR VALUES (1,'ANUSHKA','F');

INSERT INTO ACTOR VALUES (2,'PUNEETH','M');

INSERT INTO ACTOR VALUES (3,'JOHN','M');

INSERT INTO DIRECTOR VALUES (11,'YOU',9785461230);

INSERT INTO DIRECTOR VALUES (22,'ME',4567891230);

INSERT INTO DIRECTOR VALUES (33,'US',1234567890);

INSERT INTO DIRECTOR VALUES(44,'HITCHCOCK',4567892456);

INSERT INTO DIRECTOR VALUES(55,'STEVEN SPIELBERG',4567892456);

INSERT INTO MOVIES VALUES(101,'SPIDERMAN',1997,'ENGLISH',22);

INSERT INTO MOVIES VALUES(102,'SPIDERMAN1',2004,'ENGLISH',11);

INSERT INTO MOVIES VALUES(103,'SPIDERMAN',2017,'ENGLISH',22);

INSERT INTO MOVIES VALUES(104,'SPIDERMAN5',2006,'ENGLISH',44);

INSERT INTO MOVIES VALUES(105,'SPIDER',2001,'ENGLISH',55);

INSERT INTO MOVIES VALUES(106,'SPI',2000,'ENGLISH',55);

INSERT INTO MOVIE\_CAST VALUES (1,101,'HEROINE');

INSERT INTO MOVIE\_CAST VALUES (1,102,'GUEST');

INSERT INTO MOVIE\_CAST VALUES (2,103,'HERO');

INSERT INTO RATING VALUES (101,2);

INSERT INTO RATING VALUES (102,3);

INSERT INTO RATING VALUES (103,4);

INSERT INTO RATING VALUES (104,7);

INSERT INTO RATING VALUES (105,9);

INSERT INTO RATING VALUES (106,2);

Write SQL queries to

1. List the titles of all movies directed by ‘Hitchcock’.

SELECT M.MOV\_TITLE

FROM MOVIES M,DIRECTOR D

WHERE M.DIR\_ID=D.DIR\_ID AND DIR\_NAME='HITCHCOCK';

2. Find the movie names where one or more actors acted in two or more movies.

SELECT M.MOV\_TITLE

FROM MOVIES M,MOVIE\_CAST MC

WHERE M.MOV\_ID = MC.MOV\_ID AND ACT\_ID IN (SELECT ACT\_ID FROM MOVIE\_CAST MC GROUP BY ACT\_ID HAVING COUNT (ACT\_ID) > 1);

## Result Set 17

|  |
| --- |
| **MOV\_TITLE** |
| SPIDERMAN |
| SPIDERMAN1 |

3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

SELECT A.ACT\_NAME,M.MOV\_TITLE,M.MOV\_YEAR

FROM ACTOR A

JOIN MOVIE\_CAST MC ON A.ACT\_ID= MC.ACT\_ID

JOIN MOVIES M ON MC.MOV\_ID = M.MOV\_ID

WHERE M.MOV\_YEAR NOT BETWEEN 2000 AND 2015;

## Result Set 12

|  |  |  |
| --- | --- | --- |
| **ACT\_NAME** | **MOV\_TITLE** | **MOV\_YEAR** |
| ANUSHKA | SPIDERMAN | 1997 |
| PUNEETH | SPIDERMAN | 2017 |

4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.

SELECT M.MOV\_TITLE,MAX(REV\_STARS)

FROM MOVIES M

JOIN RATING R ON M.MOV\_ID = R.MOV\_ID GROUP BY MOV\_TITLE HAVING MAX (REV\_STARS)>0

ORDER BY MOV\_TITLE;

## Result Set 14

|  |  |
| --- | --- |
| **MOV\_TITLE** | **MAX(REV\_STARS)** |
| SPI | 2 |
| SPIDER | 9 |
| SPIDERMAN | 4 |
| SPIDERMAN1 | 3 |
| SPIDERMAN5 | 7 |

5. Update rating of all movies directed by ‘Steven Spielberg’ to 5.

UPDATE RATING SET REV\_STARS=5 WHERE MOV\_ID IN (SELECT MOV\_ID FROM MOVIES M WHERE DIR\_ID IN (SELECT DIR\_ID FROM DIRECTOR WHERE DIR\_NAME= 'STEVEN SPIELBERG'));

SELECT \* FROM RATING;

## Result Set 15

|  |  |
| --- | --- |
| **MOV\_ID** | **REV\_STARS** |
| 101 | 2 |
| 102 | 3 |
| 103 | 4 |
| 104 | 7 |
| 105 | 5 |
| 106 | 5 |

**COLLEGE DATABASE**

Consider the schema for College Database:

STUDENT(USN, SName, Address, Phone, Gender)

SEMSEC(SSID, Sem, Sec)

CLASS(USN, SSID)

COURSE(Subcode, Title, Sem, Credits)

IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

CREATE TABLE STUDENT (USN VARCHAR2(20) PRIMARY KEY,SNAME VARCHAR2(20),ADDRESS VARCHAR2(20),PHONE NUMBER(10),GENDER CHAR(1));

CREATE TABLE SEMSEC (SSID VARCHAR2(20) PRIMARY KEY,SEM NUMBER(02),SEC CHAR(1));

CREATE TABLE CLASS (USN REFERENCES STUDENT (USN) ON DELETE CASCADE,SSID REFERENCES SEMSEC (SSID) ON DELETE CASCADE,PRIMARY KEY(USN,SSID));

CREATE TABLE COURSE (SUBCODE VARCHAR2(20) PRIMARY KEY,TITLE VARCHAR2(20),SEM NUMBER(20),CREDITS NUMBER(2));

CREATE TABLE IAMARKS (USN REFERENCES STUDENT (USN) ON DELETE CASCADE,SSID REFERENCES SEMSEC (SSID) ON DELETE CASCADE,SUBCODE REFERENCES COURSE (SUBCODE) ON DELETE CASCADE,TEST1 NUMBER(2),TEST2 NUMBER(2),TEST3 NUMBER(2),FINALIA NUMBER(2),PRIMARY KEY(USN,SSID,SUBCODE));

**INSERTING VALUES**

INSERT INTO STUDENT VALUES ('1BI15CS101','TEESHA','BENGALURU',4561237890,'F');

INSERT INTO STUDENT VALUES ('1BI15CS102','ESHA','MANGALURU',1245789630,'F');

INSERT INTO STUDENT VALUES ('1BI15CS103','SHA','BENGALURU',4578963145,'M');

INSERT INTO STUDENT VALUES ('1BI15CS104','HIM','MYSURU',5467953120,'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('CSE4C',4,'C');

INSERT INTO CLASS VALUES ('1BI15CS101','CSE8A');

INSERT INTO CLASS VALUES ('1BI15CS102','CSE8B');

INSERT INTO CLASS VALUES ('1BI15CS103','CSE8C');

INSERT INTO CLASS VALUES ('1BI15CS104','CSE4C');

INSERT INTO COURSE VALUES ('18CS51','ME',8,4);

INSERT INTO COURSE VALUES ('18CS52','NE',8,4);

INSERT INTO COURSE VALUES ('18CS53','DBMS',8,4);

INSERT INTO COURSE VALUES ('18CS54','AUTOMATA',4,4);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS101','18CS51','CSE8A',15,20,13);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS101','18CS52','CSE8A',11,20,20);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS102','18CS51','CSE8B',20,20,15);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS102','18CS53','CSE8B',15,20,15);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS103','18CS51','CSE8C',15,20,3);

INSERT INTO IAMARKS (USN,SUBCODE,SSID,TEST1,TEST2,TEST3) VALUES ('1BI15CS104','18CS54','CSE4C',15,20,13)

Write SQL queries to

1. List all the student details studying in fourth semester ‘C’ section.

SELECT S.\*,SS.SEM,SS.SEC

FROM STUDENT S,SEMSEC SS,CLASS C

WHERE S.USN=C.USN AND C.SSID=SS.SSID AND SEM =4 AND SEC='C';

## Result Set 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **USN** | **SNAME** | **ADDRESS** | **PHONE** | **GENDER** | **SEM** | **SEC** |
| 1BI15CS104 | HIM | MYSURU | 5467953120 | M | 4 | C |

2. Compute the total number of male and female students in each semester and in each section.

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 C.SSID=SS.SSID GROUP BY SS.SEM,SS.SEC,S.GENDER ORDER BY SEM;

## Result Set 2

|  |  |  |  |
| --- | --- | --- | --- |
| **SEM** | **SEC** | **GENDER** | **COUNT** |
| 4 | C | M | 1 |
| 8 | A | F | 1 |
| 8 | B | F | 1 |
| 8 | C | M | 1 |

3. Create a view of Test1 marks of student USN ‘1BI15CS101’ in all Courses.

CREATE VIEW STU\_TEST1 AS SELECT SUBCODE,TEST1,USN

FROM IAMARKS WHERE USN='1BI15CS101';

SELECT \* FROM STU\_TEST1;

## Result Set 3

|  |  |  |
| --- | --- | --- |
| **SUBCODE** | **TEST1** | **USN** |
| 18CS51 | 15 | 1BI15CS101 |
| 18CS52 | 11 | 1BI15CS101 |

4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.

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

SELECT \* FROM IAMARKS;

## Result Set 5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **USN** | **SSID** | **SUBCODE** | **TEST1** | **TEST2** | **TEST3** | **FINALIA** |
| 1BI15CS101 | CSE8A | 18CS51 | 15 | 20 | 13 | 18 |
| 1BI15CS101 | CSE8A | 18CS52 | 11 | 20 | 20 | 20 |
| 1BI15CS102 | CSE8B | 18CS51 | 20 | 20 | 15 | 20 |
| 1BI15CS102 | CSE8B | 18CS53 | 15 | 20 | 15 | 18 |
| 1BI15CS103 | CSE8C | 18CS51 | 15 | 20 | 3 | 18 |
| 1BI15CS104 | CSE4C | 18CS54 | 15 | 20 | 13 | 18 |

5. Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = ‘Outstanding’ If FinalIA = 12 to 16 then CAT = ‘Average’ If FinalIA< 12 then CAT = ‘Weak’ Give these details only for 8th semester A, B, and C section students.