Name-Karthik Deepak

USN-1BM19CS200

Subject – DBMS

PROGRAM 4: STUDENT FACULTY DATABASE

Consider the following database for student enrollment for course :

STUDENT(snum: integer, sname: string, major: string, lvl: string, age:integer) CLASS(cname: string, meets at: time, room: string, fid: integer)

ENROLLED(snum: integer, cname: string)

FACULTY(fid: integer, fname: string, deptid: integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level(lvl) is a two character code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL. No duplicates should be printed in any of the answers.

i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by

ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.

iii. Find the names of all students who are enrolled in two classes that meet at the same time.

iv. Find the names of faculty members who teach in every room in which some class is taught.

v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

vi. Find the names of students who are not enrolled in any class.

vii. For each age value that appears in Students, find the level value that appears most often. For example, if there are more FR level students aged 18 than SR, JR, or SO students aged 18, you should print the pair (18,FR).

|  |
| --- |
| CREATE DATABASE student\_faculty; |
|  | USE student\_faculty; |
|  | CREATE TABLE student( |
|  | snum INT, |
|  | sname VARCHAR(10), |
|  | major VARCHAR(2), |
|  | lvl VARCHAR(2), |
|  | age INT, primary key(snum)); |
|  |  |
|  | CREATE TABLE faculty( |
|  | fid INT,fname VARCHAR(20), |
|  | deptid INT, |
|  | PRIMARY KEY(fid)); |
|  |  |
|  | CREATE TABLE class( |
|  | cname VARCHAR(20), |
|  | metts\_at TIMESTAMP, |
|  | room VARCHAR(10), |
|  | fid INT, |
|  | PRIMARY KEY(cname), |
|  | FOREIGN KEY(fid) REFERENCES faculty(fid)); |
|  |  |
|  | CREATE TABLE enrolled( |
|  | snum INT, |
|  | cname VARCHAR(20), |
|  | PRIMARY KEY(snum,cname), |
|  | FOREIGN KEY(snum) REFERENCES student(snum), |
|  | FOREIGN KEY(cname) REFERENCES class(cname)); |
|  |  |
|  | INSERT INTO STUDENT VALUES(1, 'jhon', 'CS', 'Sr', 19); |
|  | INSERT INTO STUDENT VALUES(2, 'Smith', 'CS', 'Jr', 20); |
|  | INSERT INTO STUDENT VALUES(3 , 'Jacob', 'CV', 'Sr', 20); |
|  | INSERT INTO STUDENT VALUES(4, 'Tom ', 'CS', 'Jr', 20); |
|  | INSERT INTO STUDENT VALUES(5, 'Rahul', 'CS', 'Jr', 20); |
|  | INSERT INTO STUDENT VALUES(6, 'Rita', 'CS', 'Sr', 21); |
|  |  |
|  | INSERT INTO FACULTY VALUES(11, 'Harish', 1000); |
|  | INSERT INTO FACULTY VALUES(12, 'MV', 1000); |
|  | INSERT INTO FACULTY VALUES(13 , 'Mira', 1001); |
|  | INSERT INTO FACULTY VALUES(14, 'Shiva', 1002); |
|  | INSERT INTO FACULTY VALUES(15, 'Nupur', 1000); |
|  |  |
|  | insert into class values('class1', '12/11/15 10:15:16', 'R1', 14); |
|  | insert into class values('class10', '12/11/15 10:15:16', 'R128', 14); |
|  | insert into class values('class2', '12/11/15 10:15:20', 'R2', 12); |
|  | insert into class values('class3', '12/11/15 10:15:25', 'R3', 11); |
|  | insert into class values('class4', '12/11/15 20:15:20', 'R4', 14); |
|  | insert into class values('class5', '12/11/15 20:15:20', 'R3', 15); |
|  | insert into class values('class6', '12/11/15 13:20:20', 'R2', 14); |
|  | insert into class values('class7', '12/11/15 10:10:10', 'R3', 14); |
|  |  |
|  | insert into enrolled values(1, 'class1'); |
|  | insert into enrolled values(2, 'class1'); |
|  | insert into enrolled values(3, 'class3'); |
|  | insert into enrolled values(4, 'class3'); |
|  | insert into enrolled values(5, 'class4'); |
|  | insert into enrolled values(1, 'class5'); |
|  | insert into enrolled values(2, 'class5'); |
|  | insert into enrolled values(3, 'class5'); |
|  | insert into enrolled values(4, 'class5'); |
|  | insert into enrolled values(5, 'class5'); |
|  |  |
|  |  |
|  | -- Query 1 |
|  | SELECT DISTINCT S.Sname |
|  | FROM Student S, Class C, Enrolled E, Faculty F |
|  | WHERE S.snum = E.snum AND E.cname = C.cname AND C.fid = F.fid AND |
|  | F.fname = 'Harish' AND S.lvl = 'Jr'; |
|  |  |
|  | -- Query 2 |
|  | SELECT DISTINCT cname |
|  | FROM class |
|  | WHERE room='R128' |
|  | OR |
|  | cname IN (SELECT e.cname FROM enrolled e GROUP BY e.cname HAVING COUNT(\*)>=5); |
|  |  |
|  |  |
|  |  |
|  | -- Query 3 |
|  | SELECT DISTINCT S.sname |
|  | FROM Student S |
|  | WHERE S.snum IN (SELECT E1.snum |
|  | FROM Enrolled E1, Enrolled E2, Class C1, Class C2 |
|  | WHERE E1.snum = E2.snum AND E1.cname <> E2.cname |
|  | AND E1.cname = C1.cname |
|  | AND E2.cname = C2.cname AND C1.metts\_at = C2.metts\_at); |
|  |  |
|  | -- Query 4 |
|  | SELECT f.fname,f.fid |
|  | FROM faculty f |
|  | WHERE f.fid in ( SELECT fid FROM class |
|  | GROUP BY fid HAVING COUNT(\*)=(SELECT COUNT(DISTINCT room) FROM class) ); |
|  |  |
|  | -- Query 5 |
|  | SELECT DISTINCT F.fname |
|  | FROM Faculty F |
|  | WHERE 5 > (SELECT COUNT(E.snum) |
|  | FROM Class C, Enrolled E |
|  | WHERE C.cname = E.cname |
|  | AND C.fid = F.fid); |
|  |  |
|  | -- Query 6 |
|  | SELECT DISTINCT S.sname |
|  | FROM Student S |
|  | WHERE S.snum NOT IN (SELECT E.snum |
|  | FROM Enrolled E ); |
|  |  |
|  | -- Query 7 |
|  | SELECT S.age, S.lvl |
|  | FROM STUDENT S |
|  | GROUP BY S.age, S.lvl |
|  | HAVING S.lvl IN(SELECT S1.lvl |
|  | FROM STUDENT S1 |
|  | WHERE S1.age=S.age |
|  | GROUP BY S1.age, S1.lvl |
|  | HAVING COUNT(\*) >= ALL (SELECT COUNT(\*) |
|  | FROM STUDENT S2 |
|  | WHERE S1.age=S2.age |
|  | GROUP BY S2.lvl, S2.age)) |
|  | ORDER BY S.age; |

Output:

