Assignment

Aug24/ DBT/ 011

Database Technologies

Diploma in Advance Computing

August 2024

**Sub-queries with joins.**

USE ***student\_phone, student\_address, faculty\_phone, faculty\_address, batch\_students, course\_batches, student\_qualifications, faculty\_qualifications, course\_modules, modules, faculty, student, course, student\_cards, and student\_order*** relation to solve the following queries.

|  |
| --- |
| 1. Display all student who have taken admission in more than 2 batches. |
| SELECT studentID FROM student\_batches GROUP BY studentID HAVING COUNT(batchID) > 2; |
|  |
| 1. Display the student detail who have joined the same batch of the student ‘saleel’. |
| SELECT s.\* FROM students s JOIN student\_batches sb ON s.studentID = sb.studentID WHERE sb.batchID IN (SELECT batchID FROM students JOIN student\_batches ON students.studentID = student\_batches.studentID WHERE namefirst = 'saleel'); |
|  |
| 1. Display all courses where least number of students have taken the admission. |
| SELECT courseID, COUNT(studentID) as student\_count FROM student\_courses GROUP BY courseID ORDER BY student\_count ASC LIMIT 1; |
|  |
| 1. Display student details who havenot taken the admission. |
| SELECT \* FROM students WHERE studentID NOT IN (SELECT DISTINCT studentID FROM student\_courses); |
|  |
| 1. Get all courses where no modules are defined in course\_modules table. |
| SELECT \* FROM courses WHERE courseID NOT IN (SELECT DISTINCT courseID FROM course\_modules); |
|  |
| 1. Display course*\_batches*details where student has taken the admission. |
| SELECT DISTINCT cb.\* FROM course\_batches cb JOIN student\_batches sb ON cb.batchID = sb.batchID; |
|  |
| 1. Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| SELECT sq1.\* FROM student\_qualifications sq1 WHERE sq1.qualification = 'BE' AND sq1.marks > (SELECT sq2.marks FROM student\_qualifications sq2 WHERE sq2.name = 'ULKA' AND sq2.qualification = 'BE'); |
|  |
| 1. Display all students whose marks are more than ‘saleel’ marks in 10th std. |
| SELECT sq1.\* FROM student\_qualifications sq1 WHERE sq1.qualification = '10th' AND sq1.marks > (SELECT sq2.marks FROM student\_qualifications sq2 WHERE sq2.name = 'saleel' AND sq2.qualification = '10th'); |
|  |
| 1. Display students whose DOB is as same as ‘kaushal’ |
| SELECT s.\* FROM students s WHERE s.DOB = (SELECT DOB FROM students WHERE namefirst = 'kaushal'); |
|  |
| 1. Display all student details who have three or more phone numbers. |
| SELECT s.\* FROM students s JOIN student\_phones sp ON s.studentID = sp.studentID GROUP BY s.studentID HAVING COUNT(sp.phoneID) >= 3; |
|  |
| 1. Display marks for the studentID 1 and 7 who have done ‘BE’. (Note: the marks must be displayed side by side). |
| SELECT (SELECT marks FROM student\_qualifications WHERE studentID = 1 AND qualification = 'BE') AS student1\_marks, (SELECT marks FROM student\_qualifications WHERE studentID = 7 AND qualification = 'BE') AS student7\_marks; |
|  |
| 1. Display marks for the studentID 1 and 7 who have done ‘BE’ also fine out the difference of marks between them.   (Note: the marks and difference between the marks must be displayed side by side) |
| SELECT  (SELECT marks FROM student\_qualifications WHERE studentID = 1 AND qualification = 'BE') AS student1\_marks,  (SELECT marks FROM student\_qualifications WHERE studentID = 7 AND qualification = 'BE') AS student7\_marks,  ABS((SELECT marks FROM student\_qualifications WHERE studentID = 1 AND qualification = 'BE') -  (SELECT marks FROM student\_qualifications WHERE studentID = 7 AND qualification = 'BE')) AS marks\_difference; |
|  |
| 1. Display all student who are not joined any of the batch. |
| SELECT \* FROM students WHERE studentID NOT IN (SELECT DISTINCT studentID FROM student\_batches); |
|  |
| 1. Display all course\_batches details who are starting on the same day as ‘Batch1’. |
| SELECT \* FROM course\_batches WHERE start\_date = (SELECT start\_date FROM course\_batches WHERE batch\_name = 'Batch1'); |
|  |
| 1. Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| SELECT sq1.\* FROM student\_qualifications sq1 WHERE sq1.qualification = '10th' AND sq1.marks > (SELECT sq2.marks FROM student\_qualifications sq2 WHERE sq2.name = 'Neel’ AND sq2.qualification = '10th'); |
|  |
| 1. Get all student with their qualification details who have highest marks in ‘BE’. |
| SELECT sq.\* FROM student\_qualifications sq WHERE sq.qualification = 'BE' ORDER BY sq.marks DESC LIMIT 1; |
|  |
| 1. Get all student with their qualification details who have second highest marks in ‘BE’. |
| SELECT sq.\* FROM student\_qualifications sq WHERE sq.qualification = 'BE' ORDER BY sq.marks DESC LIMIT 1 OFFSET 1; |
|  |
| 1. Display the student and student\_qualification details who have scored the maximum marks in ‘BE’ |
| SELECT s.\*, sq.\* FROM students s JOIN student\_qualifications sq ON s.studentID = sq.studentID WHERE sq.qualification = 'BE' ORDER BY sq.marks DESC LIMIT 1; |
|  |
| 1. Display the student details who have scored the maximum marks in ‘BE’ |
| SELECT s.\*  FROM students s JOIN student\_qualifications sq ON s.studentID = sq.studentID WHERE sq.qualification = 'BE’ ORDER BY sq.marks DESC LIMIT 1; |
|  |
| 1. Display the student details who have scored the minimum marks in ‘10’ std. |
| SELECT s.\* FROM students s JOIN student\_qualifications sq ON s.studentID = sq.studentID WHERE sq.qualification = '10th' ORDER BY sq.marks ASC LIMIT 1; |
|  |
| 1. Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
| SELECT s.\*, sq.\* FROM students s JOIN student\_qualifications sq ON s.studentID = sq.studentID  WHERE sq.qualification = 'BE' AND sq.marks > (SELECT marks FROM student\_qualifications WHERE name = 'RAJAN' AND qualification = 'BE'); |
|  |
| 1. Display all student who have done ‘BE’ in the same year as of studentID 16. |
| SELECT s.\* FROM students s JOIN student\_qualifications sq ON s.studentID = sq.studentID WHERE sq.qualification = 'BE'AND sq.graduation\_year = (SELECT graduation\_year FROM student\_qualifications WHERE studentID = 16 AND qualification = 'BE'); |
|  |
| 1. Display all odd records. |
| SELECT \*FROM students WHERE MOD(studentID, 2) != 0; |
|  |
| 1. Calculate the sum of marks student wise of their qualifications (i.e. 10th, 12th and BE marks) |
| SELECT studentID, SUM(marks) AS total\_marks FROM student\_qualifications WHERE qualification IN ('10th', '12th', 'BE') GROUP BY studentID; |
|  |
| 1. Display students’ details who are not having ***'Aadhaar'*** card. |
| SELECT \*FROM student WHERE Aadhaar IS NULL; |
|  |