Assignment

Sept23/ DBT/ 011

Database Technologies

Diploma in Advance Computing

September 2023

**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 \* from (select s.namefirst,count(cb.name) batch\_cnt from student s join course\_batches cb join batch\_students bs on s.id=bs.studentid and cb.id=bs.batchid group by s.namefirst)e where(batch\_cnt)>2; |
|  |
| 1. Display the student detail who have joined the same batch of the student ‘saleel’. |
| select \* from (select s.namefirst sname,cb.name batch from student s join course\_batches cb join batch\_students bs on s.id=bs.studentid and cb.id=bs.batchid)e where batch in(select distinct cb.name from student s join batch\_students bs join course\_batches cb on s.id=bs.studentid and cb.id=bs.batchid where namefirst='saleel'); |
|  |
| 1. Display all courses where least number of students have taken the admission. |
| select cn,cnt from (select c.name cn ,count(s.id) cnt from student s join batch\_students bs join course\_batches cb join course c on s.id=bs.studentid and cb.id=bs.batchid and c.id=cb.courseid group by c.name order by cnt)e having cnt=min(cnt); |
|  |
| 1. Display student details who have not taken the admission. |
| Select \* from student where id not in (Select distinct studentid from student s join batch\_students bs on s.id=bs.studentid); |
|  |
| 1. Get all courses where no modules are defined in course\_modules table. |
| Select name from course where id not in (select distinct cm.courseid from course c join course\_modules cm on c.id=cm.courseid); |
|  |
| 1. Display course*\_batches* details where student has taken the admission. |
| select \* from course\_batches where id in (select distinct bs.batchid from course\_batches cb join student s join batch\_students bs on cb.id=bs.batchid and s.id=bs.studentid); |
|  |
| 1. Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='BE' and marks>(select sq.marks from student s inner join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.namefirst='ulka'); |
|  |
| 1. Display all students whose marks are more than ‘saleel’ marks in 10th std. |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and marks>(select sq.marks from student s inner join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and s.namefirst='saleel'); |
|  |
| 1. Display students whose DOB is as same as ‘kaushal’ |
| select namefirst from student where dob =(Select dob from student where namefirst= 'Kaushal'); |
|  |
| 1. Display all student details who have three or more phone numbers. |
| Select \* from (Select namefirst ,count(number) no from student s join student\_phone sp on s.id=sp.studentid group by namefirst)e having no>=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 sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=1) StudentID1,(select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=7) StudentID7; |
|  |
| 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 sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=1) As StudentID1,(select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=7) As StudentID7,abs(((select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=1)-(select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and s.id=7))) As "Marks Difference";  select (select sq.marks from student s join student\_qualifications sq where s.id=sq.studentid and sq.name='be' and s.id=1) "studentid1" , (select sq.marks from student s join student\_qualifications sq where s.id=sq.studentid and sq.name='be' and s.id=7) "studentid7" , (select studentid7-studentid1) 'mark difference' |
|  |
| 1. Display all student who are not joined any of the batch. |
| select s.\* from student s where s.id not in(select s.id from student s join batch\_students bs join course\_batches cb where s.id=bs.studentid and bs.batchid=cb.id); |
|  |
| 1. Display all course\_batches details who are starting on the same day as ‘Batch1’. |
| select \* from course\_batches cb where cb.starton=(select cb.starton from course\_batches cb where cb.name="batch1"); |
|  |
| 1. Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| select s.namefirst,sq.marks,sq.name from student s join student\_qualifications sq where s.id=sq.studentid and marks>(select sq.marks from student s join student\_qualifications sq where s.id =sq.studentid and sq.name='10' and s.namefirst='neel') and sq.name='10'; |
|  |
| 1. Get all student with their qualification details who have highest marks in ‘BE’. |
| select s.\* from student s join student\_qualifications sq where s.id=sq.studentid and marks =(select max(sq.marks) from student s join student\_qualifications sq where s.id = sq.studentid and sq.name='be'); |
|  |
| 1. Get all student with their qualification details who have second highest marks in ‘BE’. |
|  |
|  |
| 1. Display the student and student\_qualification details who have scored the maximum marks in ‘BE’ |
| select s.\* ,sq.\* from student s join student\_qualifications sq where s.id=sq.studentid and marks =(select max(sq.marks) from student s join student\_qualifications sq where s.id = sq.studentid and sq.name='be'); |
|  |
| 1. Display the student details who have scored the maximum marks in ‘BE’ |
| select s.\* from student s join student\_qualifications sq where s.id=sq.studentid and marks =(select max(sq.marks) from student s join student\_qualifications sq where s.id = sq.studentid and sq.name='be'); |
|  |
| 1. Display the student details who have scored the minimum marks in ‘10’ std. |
|  |
|  |
| 1. Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
|  |
|  |
| 1. Display all student who have done ‘BE’ in the same year as of studentID 16. |
|  |
|  |
| 1. Display all odd records. |
|  |
|  |
| 1. Calculate the sum of marks student wise of their qualifications (i.e. 10th, 12th and BE marks) |
|  |
|  |
| 1. Display students’ details who are not having ***'Aadhaar'*** card. |
|  |
|  |