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 student where id in(select batchid from batch\_students group by batchid having count(batchid)>2); |
|  |
| 1. Display the student detail who have joined the same batch of the student ‘saleel’. |
| select \* from student s join batch\_students bs where s.id=bs.studentid and batchid in (select cb.id from student s join batch\_students bs join course\_batches cb where s.id=bs.studentid and bs.batchid=cb.id and namefirst='saleel') ; |
|  |
| 1. Display all courses where least number of students have taken the admission. |
| select \* from(select c.name,count(c.id)r1 from course c join course\_batches cb on c.id=cb.courseid join batch\_students bs on cb.id=bs.batchid join student s on bs.studentid=s.id group by c.name)e order by r1 limit 2; |
|  |
| 1. Display student details who have not taken the admission. |
| select \* from student where id not in(select studentid from batch\_students); |
|  |
| 1. Get all courses where no modules are defined in course\_modules table. |
| select \* from course where id not in(select courseid from course\_modules); |
|  |
| 1. Display course*\_batches* details where student has taken the admission. |
| select \* from course\_batches where id in(select studentid from batch\_students); |
|  |
| 1. Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| 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='be' and s.namefirst='ulka') and sq.name='be';  select \* from (select s.namefirst,sq.marks r,sq.name from student s join student\_qualifications sq where s.id=sq.studentid and sq.name='be') e where r>67; |
|  |
| 1. Display all students whose marks are more than ‘saleel’ marks in 10th std. |
| 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='saleel')and sq.name='10'; |
|  |
| 1. Display students whose DOB is as same as ‘kaushal’ |
| select \* 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 student where id in (select studentid from student\_phone group by studentid having count(number)>2); |
|  |
| 1. Display marks for the studentID 1 and 7 who have done ‘BE’. (Note: the marks must be displayed side by side). |
| select student1,student7 from (select marks as Student1 from student\_qualifications sq where sq.studentid=1 and sq.name='BE')r1, (select marks as Student7 from student\_qualifications sq where sq.studentid=7 and sq.name='BE')r2; |
|  |
| 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 student1,student7,abs(student1-student7) as 'Marks Difference' from (select marks as Student1 from student\_qualifications sq where sq.studentid=1 and sq.name='BE')r1, (select marks as Student7 from student\_qualifications sq where sq.studentid=7 and sq.name='BE')r2; |
|  |
| 1. Display all student who are not joined any of the batch. |
| select \* from student where id not in(select studentid from batch\_students); |
|  |
| 1. Display all course\_batches details who are starting on the same day as ‘Batch1’. |
| select \* from course\_batches where starton=(select starton from course\_batches where name="batch1"); |
|  |
| 1. Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| select s.\*,sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name=10 and sq.marks>( select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where s.namefirst="neel" and sq.name=10); |
|  |
| 1. Get all student with their qualification details who have highest marks in ‘BE’. |
| select s.\*,sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.marks=(select max(marks) from student\_qualifications sq where sq.name="BE"); |
|  |
| 1. Get all student with their qualification details who have second highest marks in ‘BE’. |
| select \* from student,student\_qualifications where student.id=student\_qualifications.studentid and name = 'BE' and student.id in (select studentid from (select \*,rank() over(order by marks desc) R1 from student\_qualifications where name = 'BE' and marks < (select max(marks) from student\_qualifications where name='BE' order by marks desc)) t where R1=1 ); |
|  |
| 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 on s.id=sq.studentid where sq.marks=(select max(marks) from student\_qualifications sq where 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 on s.id=sq.studentid where sq.marks=(select max(marks) from student\_qualifications sq where sq.name="BE"); |
|  |
| 1. Display the student details who have scored the minimum marks in ‘10’ std. |
| select s.\* from student s join student\_qualifications sq on s.id=sq.studentid where sq.marks=(select min(marks) from student\_qualifications sq where sq.name="10"); |
|  |
| 1. Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
| select \* from (select s.namefirst,sq.marks r,sq.name from student s join student\_qualifications sq where s.id=sq.studentid and sq.name='be') e where r>68; |
|  |
| 1. Display all student who have done ‘BE’ in the same year as of studentID 16. |
| select s.\*,sq.year,sq.name from student s join student\_qualifications sq where s.id=sq.studentid and sq.year=(select sq.year from student\_qualifications sq where sq.studentid=16 and name='be') and sq.name='be'; |
|  |
| 1. Display all odd records. |
| select \* from(select \* from student where id%2=1)e; |
|  |
| 1. Calculate the sum of marks student wise of their qualifications (i.e. 10th, 12th and BE marks) |
| select \* from (select s.\*,sum(sq.marks) from student s join student\_qualifications sq where s.id=sq.studentid group by sq.studentid)k; |
|  |
| 1. Display students’ details who are not having ***'Aadhaar'*** card. |
| select \*,sa.name from student student\_cards sa where id in (select studentid from student\_cards where name="aadhaar"); |
|  |