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 e.studentid,namefirst,namelast from (select studentid StudentID,count(\*) num\_of\_batches from batch\_students bs group by studentid having count(\*)>2)e,student s where s.id = e.StudentID; |
|  |
| 1. Display the student detail who have joined the same batch of the student ‘saleel’. |
| select \* from student where id in (select studentid from batch\_students where batchid in (select batchid from batch\_students where studentid in (select id from student where namefirst='saleel'))); |
|  |
| 1. Display all courses where least number of students have taken the admission. |
| select distinct course.name from course join(select \* from course\_batches join (Select batchid from (Select batchid,count(\*) count from batch\_students group by batchid) a join (select min(b.count) ct from (Select batchid,count(\*) count from batch\_students group by batchid) b) c where a.count=c.ct) t1 where course\_batches.id=t1.batchid) t2 where course.id=t2.courseid; |
|  |
| 1. Display student details who have not taken the admission. |
| select \* from student where student.id not in(select studentid from batch\_students); |
|  |
| 1. Get all courses where no modules are defined in course\_modules table. |
| select course.name from course where course.id not in (select courseid from course\_modules); |
|  |
| 1. Display course*\_batches* details where student has taken the admission. |
| select \* from course\_batches cb where cb.id in (select batchid from batch\_students); |
|  |
| 1. Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| select s.namefirst from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='be' and sq.marks>(select sq.marks from student\_qualifications sq where sq.studentid=(select id from student s where s.namefirst='ulka') and sq.name='be'); |
|  |
| 1. Display all students whose marks are more than ‘saleel’ marks in 10th std. |
| select s.id,s.namefirst from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and sq.marks>(select sq.marks from student\_qualifications sq where sq.studentid=(select id from student s where s.namefirst='saleel') and sq.name='10'); |
|  |
| 1. Display students whose DOB is as same as ‘kaushal’ |
| select namefirst,dob from student where dob=(select dob from student where namefirst='kaushal'); |
|  |
| 1. Display all student details who have three or more phone numbers. |
| select namefirst from student where student.id in (select studentid from student\_phone group by studentid having count(number)>=3); |
|  |
| 1. Display marks for the studentID 1 and 7 who have done ‘BE’. (Note: the marks must be displayed side by side). |
| select t1.marks as "stdent id 1" ,t2.marks as "student id 7" from ((select marks from student\_qualifications where studentid =1 and name='be')t1 ,(select marks from student\_qualifications where studentid =7 and name='be') t2); |
|  |
| 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 t1.marks as "stdent id 1" ,t2.marks as "student id 7",t2.marks-t1.marks as "marks diff" from ((select marks from student\_qualifications where studentid =1 and name='be')t1 , (select marks from student\_qualifications where studentid =7 and name='be') t2); |
|  |
| 1. Display all student who are not joined any of the batch. |
| select namefirst 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 course\_batches.name from course\_batches where starton=(select starton from course\_batches where course\_batches.name='batch1'); |
|  |
| 1. Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| select s.id,s.namefirst from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and sq.marks>(select sq.marks from student\_qualifications sq where sq.studentid=(select id from student s where s.namefirst='neel') and sq.name='10'); |
|  |
| 1. Get all student with their qualification details who have highest marks in ‘BE’. |
| select name,studentid,marks from student\_qualifications where marks in (select max(marks) from student\_qualifications where name='be')and name='be'; |
|  |
| 1. Get all student with their qualification details who have second highest marks in ‘BE’. |
| select \* from student where id in (select studentid from student\_qualifications where marks=(select distinct marks from student\_qualifications where name = 'be' order by marks desc limit 1,1) and name = 'be'); |
|  |
| 1. Display the student and student\_qualification details who have scored the maximum marks in ‘BE’ |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where s.id in (select studentid from student\_qualifications where marks in (select max(marks) from student\_qualifications where name = 'be') and name ='be') and sq.name ='be'; |
|  |
| 1. Display the student details who have scored the maximum marks in ‘BE’ |
| select \* from student where id in (select studentid from student\_qualifications where marks in (select max(marks) from student\_qualifications where name = 'be') and name ='be'); |
|  |
| 1. Display the student details who have scored the minimum marks in ‘10’ std. |
| select \* from student where id in (select studentid from student\_qualifications where marks in (select min(marks) from student\_qualifications where name = '10') and name ='10'); |
|  |
| 1. Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where s.id in (select studentid from student\_qualifications where marks>(select marks from student\_qualifications where studentid in (select studentid from student\_qualifications where studentid in (select id from student where namefirst='rajan') and name = 'be') and name = 'be') and name = 'be') and sq.name = 'be'; |
|  |
| 1. Display all student who have done ‘BE’ in the same year as of studentID 16. |
| select \* from student where id in (select studentid from student\_qualifications where year in (select year from student\_qualifications where studentid = 16 and name = 'be') and name = 'be'); |
|  |
| 1. Display all odd records. |
| select \* from student where id in (select id from student where id%2!=0); |
|  |
| 1. Calculate the sum of marks student wise of their qualifications (i.e. 10th, 12th and BE marks) |
| select studentid,sum(marks) from student\_qualifications group by studentid; |
|  |
| 1. Display students’ details who are not having ***'Aadhaar'*** card. |
| select \* from student where id not in (select studentid from student\_cards where name = 'aadhaar'); |
|  |