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.

|  |
| --- |
| * Display all student who have taken admission in more than 2 batches. |
| select \* from student s, ( select studentid, count(batchid) Count from batch\_students group by studentid having count(batchid)>2)e where s.id=e.studentid ; |
|  |
| * Display the student detail who have joined the same batch of the student ‘saleel’. |
| select s.namefirst, s.namelast,s.DOB,s.emailID from student s join batch\_students bs on s.id= bs.studentid where bs.batchid in (select course\_batches.id from student join batch\_students on student.id= batch\_students.studentid join course\_batches on batch\_students.batchid= course\_batches.id where student.namefirst='saleel'); //  select s.id ,s.namefirst, s.namelast,e.batchid from student s, (select studentid, batchid from batch\_students having batchid in(1,6))e where s.id=e.studentid order by s.id; |
|  |
| * 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; |
|  |
| * Display student details who have not taken the admission. |
| select \* from student where id not in (select studentid from batch\_students);//  select \* from student where student.id not in (select s.id from student s join batch\_students bs on s.id=bs.studentid); |
|  |
| * Get all courses where no modules are defined in course\_modules table. |
| select id,name from course where id not in(select courseid from course\_modules); |
|  |
| * Display course*\_batches* details where student has taken the admission. |
| select distinct batchid, name from batch\_students bs join course\_batches cb on cb.id=bs.batchid ; //  select s.id, s.namefirst, s.namelast,cb.name from student s join batch\_students bs on s.id=bs.studentid join course\_batches cb on bs.batchid=cb.id where cb.id in (select id from course\_batches where id in(select distinct batchid from batch\_students)); |
|  |
| * Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| select s.namefirst, s.namelast,sq.name,sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='BE' and marks >(select marks from student\_qualifications where name='BE' and studentid=(select id from student where namefirst='ulka')); |
|  |
| * Display all students whose marks are more than ‘saleel’ marks in 10th std. |
| select s.namefirst,s.namelast, sq.name,sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and marks >(select marks from student\_qualifications where name='10' and studentid=(select id from student where namefirst='saleel')); |
|  |
| * Display students whose DOB is as same as ‘kaushal’ |
| select namefirst, namelast, DOB from student where DOB = (select DOB from student where namefirst='kaushal'); |
|  |
| * Display all student details who have three or more phone numbers. |
| select id,namefirst,namelast from student where id in(select studentid from student\_phone group by studentid having count(studentid)>2); |
|  |
| * Display marks for the studentID 1 and 7 who have done ‘BE’. (Note: the marks must be displayed side by side). |
| select t1.marks"studentID 1",t2.marks"studentID 2" from ((select marks from student\_qualifications where name='BE' and studentid=1)t1 join(select marks from student\_qualifications where name='BE' and studentid=7)t2); |
|  |
| * 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"studentID 1",t2.marks"studentID 2",t2.marks-t1.marks "Marks Difference" from ((select marks from student\_qualifications where name='BE' and studentid=1)t1 join(select marks from student\_qualifications where name='BE' and studentid=7)t2); |
|  |
| * Display all student who are not joined any of the batch. |
| select id,namefirst,namelast from student where id not in(select distinct studentid from batch\_students); |
|  |
| * Display all course\_batches details who are starting on the same day as ‘Batch1’. |
| select name from course\_batches where starton=(select starton from course\_batches where name="Batch1"); |
|  |
| * Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| select s.id,s.namefirst,s.namelast,sq.name,sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where sq.name='10' and sq.marks>(select marks from student\_qualifications sq join student s on s.id=sq.studentid where sq.name='10' and s.namefirst='neel'); |
|  |
| * Get all student with their qualification details who have highest marks in ‘BE’. |
| select s.id,namefirst, namelast,name,college,university,marks,year from student s join student\_qualifications sq on s.id=sq.studentid where s.id in(select studentid from student\_qualifications where name='BE' and marks=(select max(marks) from student\_qualifications where name='BE')); |
|  |
| * Get all student with their qualification details who have second highest marks in ‘BE’. |
| select s.id,namefirst, namelast,name,college,university,marks,year from student s join student\_qualifications sq on s.id=sq.studentid where s.id in(select studentid from student\_qualifications where name='BE' and marks=(select max(marks) from student\_qualifications where name='BE' and marks<(select max(marks) from student\_qualifications where name='BE'))); |
|  |
| * Display the student and student\_qualification details who have scored the maximum marks in ‘BE’ |
| |  | | --- | | select s.id,namefirst, namelast,name,college,university,marks,year from student s join student\_qualifications sq on s.id=sq.studentid where s.id in(select studentid from student\_qualifications where name='BE' and marks=(select max(marks) from student\_qualifications where name='BE')); | |
|  |
| * Display the student details who have scored the maximum marks in ‘BE’ |
| select id,namefirst, namelast from student where id in(select studentid from student\_qualifications where name='BE' and marks=(select max(marks) from student\_qualifications where name='BE')); |
|  |
| * 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 name='10' and marks=(select min(marks) from student\_qualifications where name='10')); |
|  |
| * Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
| select s.id,namefirst, namelast,name,college,university,marks,year from student s join student\_qualifications sq on s.id=sq.studentid where sq.name="BE" and sq.marks>(select marks from student\_qualifications where name="BE" and studentid=(select id from student where namefirst="RAJAN")); |
|  |
| * Display all student who have done ‘BE’ in the same year as of studentID 16. |
| select s.id,namefirst, namelast,name,year from student s join student\_qualifications sq on s.id=sq.studentid where sq.name="BE" and sq.year=(select year from student\_qualifications where name="BE" and studentID=16); |
|  |
| * Display all odd records. |
| select \* from student where id in(select id from student where id%2=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; |
|  |
| * Display students’ details who are not having ***'Aadhaar'*** card. |
| select \* from student where id not in(select studentid from student\_cards where name="aadhaar"); |
|  |