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 studentid from batch\_students group by studentid having count(\*)>2) ; |
|  |
| 1. Display the student detail who have joined the same batch of the student ‘saleel’. |
| select \* from student s join batch\_students bs on s.id=bs.studentid where batchid in(select batchid from batch\_students where studentid=1); |
|  |
| 1. Display all courses where least number of students have taken the admission. |
| select \* from ( select cs.name,count(cs.id) r1 from course cs join course\_batches cb join batch\_students bs join student s where cs.id=cb.courseid and cb.id=bs.batchid and bs.studentid=s.id group by cs.name) e order by r1 limit 2; |
|  |
| 1. Display student details who have not taken the admission. |
| select \* from student s where id not in(select studentid from batch\_students group by studentid); |
|  |
| 1. Get all courses where no modules are defined in course\_modules table. |
| select \* from course c where c.id not in(select courseid from course\_modules cm); |
|  |
| 1. Display course*\_batches* details where student has taken the admission. |
| select \* from course\_batches where id in (select batchid from batch\_students); |
|  |
| 1. Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’. |
| select \* from student s join student\_qualifications q on s.id = q.studentid where q.marks>(select marks from student\_qualifications sq join student s 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 q on s.id = q.studentid where q.name="10" and q.marks>(select marks from student\_qualifications sq join student s 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 namefirst from student where id in (select sp.studentid from student s join student\_phone sp on s.id = sp.studentid 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). |
|  |
|  |
| 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) |
|  |
|  |
| 1. Display all student who are not joined any of the batch. |
| select \* from student s where s.id not in(select s.id from student s join batch\_students bs on s.id=bs.studentid where s.id = bs.studentid order by id); |
|  |
| 1. 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") ; |
|  |
| 1. Display all students whose 10th marks is more than student ‘Neel’s 10th marks. |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where marks>(select marks from student\_qualifications sq join student s on s.id=sq.studentid where s.namefirst ="neel" and sq.name="10")and name="10"; |
|  |
| 1. Get all student with their qualification details who have highest marks in ‘BE’. |
| select \* from student s join student\_qualifications sq on s.id=sq.studentid where marks=(select max(marks) from student\_qualifications group by name having name="BE"); |
|  |
| 1. Get all student with their qualification details who have second highest marks in ‘BE’. |
| select \* from (select dense\_rank() over(order by marks desc) r ,s.\*, marks from student s join student\_qualifications sq on s.id=sq.studentid where name="be") e where r=2 ; |
|  |
| 1. Display the student and student\_qualification details who have scored the maximum marks in ‘BE’ |
| select \* from (select dense\_rank() over(order by marks desc) r ,s.\*, marks from student s join student\_qualifications sq on s.id=sq.studentid where name="be") e where r=1; |
|  |
| 1. Display the student details who have scored the maximum marks in ‘BE’ |
| select \* from (select dense\_rank() over(order by marks desc) r ,s.\*, marks from student s join student\_qualifications sq on s.id=sq.studentid where name="be") e where r=1; |
|  |
| 1. Display the student details who have scored the minimum marks in ‘10’ std. |
| select \* from (select dense\_rank() over(order by marks) r , s.\*, marks from student s join student\_qualifications sq on s.id = sq.studentid where name=10)e where r=1; |
|  |
| 1. Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’. |
| select \* from student\_qualifications sq join student s on s.id=sq.studentid where name=10 and marks > (select sq.marks from student s join student\_qualifications sq on s.id=sq.studentid where namefirst="rajan" and name="be"); |
|  |
| 1. Display all student who have done ‘BE’ in the same year as of studentID 16. |
| select s.\*,sq.\* from student s join student\_qualifications sq on s.id=sq.studentid where year=(select year from student\_qualifications where studentid = 16 and name ="be"); |
|  |
| 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. |
|  |
|  |