**Assignment**

**Feb25/ DBT/ 011**

**Database Technologies**

**Diploma in Advance Computing**

**February 2025**

**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 r1.namefirst,count(\*) from (select s.\* ,cb.name from student s join batch\_students bs on s.id=bs.studentid join course\_batches cb on cb.id=bs.batchid join course cs on cb.courseid=cs.id ) r1 group by r1.id having count(\*)>2;** |
|  |
| 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=(select id from student where nam**  **efirst='saleel')));** |
|  |
| 1. **Display all courses where least number of students have taken the admission.** |
| **select r2.name,r1 from( select c.name,count(\*) r1 from batch\_students bs join course\_batches cb on bs.batchid=cb.id join course c on c.id=cb.courseid group by name) r2 where r1<=( select min(s1) from (select c.name,count(\*) s1 from batch\_students bs join course\_batches cb on bs.batchid=cb.id join course c on c.id=cb.courseid group by name) s);** |
|  |
| 1. **Display student details who have not taken the admission.** |
| **select \*from student where id not in( select studentid from batch\_students where batchid in (select id from course\_batches));** |
|  |
| 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 student where id in( select studentid from batch\_students);** |
|  |
| 1. **Display all students whose marks of ‘BE’ is more than ‘ULKA’ marks in ‘BE’.** |
| **select \* from student where id in ( select studentid from student\_qualifications where name='be' and marks>(select marks from student\_qualifications where name='be' and id in(select id from student where namefirst='ulka')));** |
|  |
| 1. **Display all students whose marks are more than ‘saleel’ marks in 10th std.** |
| **select \* from student where id in ( select studentid from student\_qualifications where name='10' and marks>(select marks from student\_qualifications where name='10' and id in(select id from student where namefirst='saleel')));** |
|  |
| 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 (select studentid,count(\*) x from student\_phone group by studentid) r1 where x>=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 marks from student\_qualifications where name='be' and**  **studentid=1) studentID\_1 ,(select marks from student\_qualifications where na**  **me='be' and studentid=7) studentID\_7;** |
|  |
| 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 marks from student\_qualifications where name='be' and studentid=1) studentID\_1 ,(select marks from student\_qualifications where name='be' and studentid=7) studentID\_7 , abs((select marks from student\_qualifications where name='be' and studentid=1)-(select marks from student\_qualifications where name='be' and studentid=7))Marks\_diffrence;** |
|  |
| 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 \* from student where id in(select studentid from student\_qualifications where name='10' and marks>(select marks from student\_qualifications where name='10' and studentid=(select id from student where namefirst='neel')));** |
|  |
| 1. **Get all student with their qualification details who have highest marks in ‘BE’.** |
| **select s.\*,sq.\* from student s join student\_qualifications sq on s.id=sq.stu**  **dentid where s.id in(select studentid from student\_qualifications where name='be' and marks=(select max(marks) from student\_qualifications where name='be'));** |
|  |
| 1. **Get all student with their qualification details who have second highest marks in ‘BE’.** |
| **select s.\*,sq.\* 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 distinct(marks) from student\_qualifications where name='be' order by marks desc limit 1,1));** |
|  |
| 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.stu**  **dentid where s.id in(select studentid from student\_qualifications where name='be' and marks=(select max(marks) from student\_qualifications where 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 name='be' and marks=(select max(marks) from student\_qualifications where name='be'));** |
|  |
| 1. **Display the student details who have scored the minimum marks in ‘10’ std.** |
| **select \* from student where id=(select studentid from student\_qualifications where name='10' and marks=(select distinct marks from student\_qualifications where name='10' order by marks limit 1));** |
|  |
| 1. **Display all student and student\_qualification details of those students who have scored marks more than ‘RAJAN’ in ‘BE’.** |
| **select s.\*,sq.\* 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 marks from student\_qualifications where name='be' and studentid=(select id from student where namefirst='rajan')));** |
|  |
| 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 name='be' and year=(select year from student\_qualifications where studentid=16 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 where name='10' or n**  **ame='12' or name='be' 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');** |
|  |