Assignment

Aug24/ DBT/ 007

Database Technologies

Diploma in Advance Computing

August 2024

**DML commands: Select data with WHERE, GROUP BY, HAVING, ORDER BY and LIMIT clause.**

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. List all student. |
| **select \* from student;** |
|  |
| 1. List namefirst,namelast of all student. |
| **select namefirst,namelast from student;** |
|  |
| 1. Display student information of the student whosestudent *ID* is 10. |
| **select id,namefirst,namelast,DOB,emailid from student where id=1;** |
|  |
| 1. List of various faculties available from faculty table. |
| **select \* from faculty;** |
|  |
| 1. List all student having ‘A’ as second letter in their namefirst. |
| **SELECT \* FROM students WHERE SUBSTRING(first\_name, 2, 1) = 'A';** |
|  |
| 1. List all student having letter ‘A’ in their namefirst. |
| SELECT \* FROM student WHERE POSITION('A' IN namefirst) > 0; |
|  |
| 1. Display the details of the student whoseDOBis '1986-12-14'. |
| **SELECT \* FROM student WHERE DOB = '1986-12-14';** |
|  |
| 1. List all student having ‘R’ as first letter in their namefirst. |
| SELECT \* FROM student WHERE LEFT(namefirst, 1) = 'R'; |
|  |
| 1. Display the *namefirst, lastname* from student relation with Customized column headings. |
| **SELECT namefirst AS "First Name", namelast AS "Last Name" FROM student;** |
|  |
| 1. Display all students in ascending order of their DOB. |
| **select \* from student order by DOB asc;** |
|  |
| 1. Display two records of student whose name starts with the letter ‘S’. |
| **SELECT \* FROM student WHERE LEFT(namefirst, 1) = 'S' LIMIT 2;** |
|  |
| 1. Display the student detail whose DOB is ‘1986-12-14’. |
| **SELECT \* FROM student WHERE DOB = '1986-12-14';** |
|  |
| 1. Display all modules whose module duration is 1 (use modules table). |
| **select \* from modules where duration='1';** |
|  |
| 1. Display all batches whose sitting capacity is 80 students (use course\_batches table). |
| **select \* from course\_batches where capacity='80';** |
|  |
| 1. Display all student qualification who have done’ BE’ and secured marks more than 70. (use student\_qualifications table). |
| **select \* from student\_qualifications where name='BE'having marks>70;** |
|  |
| 1. Display all student qualification who have done’ BE’ and graduated in the year 2017. (use student\_qualifications table). |
| **select \* from student\_qualifications where name='BE'having year='2017';** |
|  |
| 1. Display all student qualification who have done’ BE’ and graduated in the year 2017 and scored marks more than 80. (use student\_qualifications table). |
| **select \* from student\_qualifications where year='2017'having marks>'80';** |
|  |
| 1. Display faculty qualification who have done ‘BE’ from ‘Harvard University’(use faculty\_qualifications table) |
| **select \* from student\_qualifications where name = 'be' and university = 'harvard university';** |
|  |
| 1. Display all courses whose course duration is 6 months.(use course table) |
| **select \* from course where duration='6';** |
|  |
| 1. Display module details whose module duration is between 1 and 2, arrange the data in ascending order of module duration. (use module table) |
| **select \* from modules where duration between 1 and 2 order by duration asc;** |
|  |
| 1. Display all student with their voting rights, if the student is below 1980 then print the message “\*The student can vote” else print “The student cannot vote”. |
| **select id,namefirst,dob,case when dob < 1980 then '\*the student can vote'else 'the student cannot vote' end as voting\_rights from student;** |
|  |
| 1. Display all distinct universities from student\_qualifications table. |
| **select distinct university from student\_qualifications;** |
|  |
| 1. Display the second highest marks scored by any student in ‘BE’. |
| **select max(marks) as second\_highest\_marks from student\_qualifications where name = 'be' and marks < (select max(marks) from student\_qualifications where name = 'be');** |
|  |
| 1. Display the second lowest marks scored by any student in ‘BE’. |
| **select min(marks) as second\_lowest\_marks from student\_qualifications where name = 'be' and marks > ( select min(marks) from student\_qualifications where name = 'be');** |
|  |
| 1. Display last 7 student. |
| **select \* from student order by id desc limit 7;** |
|  |