**Database Management Systems**

**Course code: CS/AI 3103**

**Assignment -4**

**Note:** Consider the database or tables you have created in assignment-1 to solve this assignment.

**Instructions to open mysql and using created database**

1. **Start mysql using**

$ mysql -u root –p

(enter password if prompted)

1. **Use database inside which you have created and store the tables given in the assignment-1**

use studentdb;

1. **Use the following table created in assignment-1 to solve this assignment**

Student (snum, sname, major, standing, age)

Faculty (fid, fname, deptid)

Class (name, meets, room, fid)

Enrolled (snum, cname)

1. **Create a table employee in studentdb database**

Create table employee(empid numeric(9,0),empname varchar(20), department varchar(20),salary numeric(10,0), bonus numeric(10,0));

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| empid | empname | department | salary | bonus |
| 1 | Mark | HR | 30000 | 6000 |
| 2 | Miraj | Admin | 25000 | 5000 |
| 3 | Chinaya | HR | 50000 | 9000 |
| 4 | Dilkush | Admin | 25000 | 6000 |
| 5 | Eimanual | Account | 10000 | 4000 |
| 6 | Fisal | Account | 12000 | 7000 |
| 7 | Girkshan | HR | 55000 | 12000 |
| 8 | Milka shingh | HR | 58000 | 10000 |

1. **Exercise on BETWEEN query:**

**Syntax: SELECT column\_name1, column\_name2 FROM table\_name**

**WHERE column\_name1 BETWEEN value1 AND value2;**

**Syntax: SELECT column\_name1, column\_name2 FROM table\_name**

**WHERE column\_name1 NOT BETWEEN value1 AND value2;**

1. Write the sql query to display all records from the student table whose age between 20 and 23.

**Solution: select \* FROM student WHERE age BETWEEN 20 and 23;**

1. Write the sql query to fetch snum, sname, major and age columns from the student table whose age is between 20 and 23.

**Solution: select snum, sname, major, age FROM student WHERE age BETWEEN 20 and 23;**

1. Write the sql query to fetch snum, sname, major and age columns from the student table whose age is not between 22 and 24.

**Solution: select snum, sname, major, age FROM student WHERE age NOT BETWEEN 22 and 24;**

1. Write the sql query to fetch snum, sname, major and age columns from the student table whose age is not between 22 and 24 and major not in English.

**Solution: SELECT snum, sname, major, age FROM student where age BETWEEN 20 AND 25 AND major NOT IN ('English');**

1. **Exercise on IN and Not IN**

**Syntax: SELECT column\_name1, column\_name2 FROM table\_name WHERE column\_name1 IN (value1, value2, etc);**

**Syntax: SELECT column\_name1, column\_name2 FROM table\_name WHERE column\_name1 NOT IN (value1, value2, etc);**

1. Write the sql query to fetch names, number and age from student table whose ages are in 22,23.

**Solution: SELECT snum, sname, age FROM student WHERE age IN (22,23);**

1. Write the sql query to fetch names, number and age from student table whose ages are in 22,25 and major not in English and Architecture.

**Solution: SELECT snum, sname, major, age FROM student where age IN (22,25) AND major NOT IN ('English','Architecture');**

1. **Exercise on LIKE**

**Syntax: SELECT column 1, column 2 FROM table\_name WHERE column Like Pattern;**

1. Write the sql query to fetch the records of employees whose name starts with ‘Do’.

**Solution: SELECT empid, empname FROM employee WHERE empname LIKE 'Do%';**

1. Write the sql query to fetch the records of employees whose name end with ‘on’.

**Solution: SELECT empid, empname FROM employee WHERE empname LIKE '%on';**

1. Write the sql query to fetch the records of employees whose name consists of with ‘rk’.

**Solution: SELECT empid, empname FROM employee WHERE empname LIKE '%rk%';**

1. Write the sql query to fetch the records of employees whose name starts with ‘M’ but not starts with ‘MA’.

**Solution: Select \* from employee where empname like 'M%' and empname not like 'MA%';**

1. Write the sql query to fetch the records of employees whose name starts with M and are atleast two character in length.

**Solution: Select \* from employee where empname like 'M\_%';**

1. Write the sql query to fetch the records of employees whose name starts with M and ends with j.

**Solution: Select \* from employee where empname like 'M%j';**

1. **Exercise on Exists**

Consider the following tables.



**Syntax:**

**SELECT *column\_name(s)* FROM *table\_name* WHERE EXISTS  
(SELECT *column\_name*FROM *table\_name* WHERE *condition*);**

1. Write the sql query to fetch the details of the student who have enrolled in the course of ‘database systems’;

**Solution:** Select \* from student where exists (select \* from enrolled where student.snum=enrolled.snum and cname = 'Database Systems');

1. **Exercise on ANY**

**Syntax:**

SELECT *column\_name(s)* FROM *table\_name* WHERE *column\_name operator* ANY   (SELECT *column\_name*  
FROM *table\_name*WHERE *condition*);

1. Write the SQL query to find details of the students if any student enrolled for course patent law.

**Solution:** Select \* from student where snum=any(select snum from enrolled where cname='Patent Law');

1. **Exercise on ALL**

**Syntax:**

SELECT column\_name(s) FROM table\_name  
WHERE column\_name operator ALL   (SELECT column\_name  
  FROM table\_name   WHERE condition);

1. Write the SQL query to find details of the students for all the students who have enrolled for database systems.

**Solution:** select \* from student where snum=ALL(select snum from enrolled where cname='Database Systems');

**Note:** The above SQL statement lists the student details if ALL the records in the enrolled table has the cname=’database Systems’.

1. **Aggregate functions**

**MIN ( )**

Syntax:

SELECT MIN(*column\_name*) FROM *table\_name* WHERE *condition*;

1. Write an SQL query to find the Minimum salary from employee table.

**Solution: select min(salary) from employee;**

**MAX ( )**

Syntax:

SELECT MIN(*column\_name*) FROM *table\_name* WHERE *condition*;

Write an SQL query to find the Maximum salary from employee table.

**Solution: select max(salary) from employee;**

**COUNT ( )**

Syntax:

SELECT COUNT(*column\_name*) FROM *table\_name*  
WHERE *condition*;

* + - 1. Write an SQL query to count number of employee names from employee table.

**Solution: select count(empname) from employee;**

**AVG ( )**

Syntax:

SELECT AVG(*column\_name*) FROM *table\_name* WHERE *condition*;

Write an SQL query to Display the Average salary from Employee table.

**Solution: select avg(salary) from employee;**

**SUM ( )**

Syntax:

SELECT SUM(*column\_name*) FROM *table\_name*  
WHERE *condition*;

* + - 1. Write an SQL query to Display the total salary from employee table.

**Solution: select sum(salary) from employee;**

**STDDEV( )**

Syntax: Select stddev(column-name) from table-name where condition;

Write a sql query to display the standard deviation of the salaries in the employee table:

**Solution: select stddev(salary) from employee where salary> 7000;**

**Variance()**

**Syntax:** Select variance(column-name) from table-name where condition;

1. **Write a sql query to display the variance in the employee salary for less than 9000.**

**Solution: select variance(salary) from employee where salary< 9000;**