MIS 301 Relational Database Management System

DATABASE MANAGEMENT SYSTEM

Structured Query Language(SQL)-3

LECTURE 13 & 14

Data Query Language-Select

- The select command is the only DQL command.
- It is merged with the DML commands by some thinkers.
- The select command is used for extracting data from the database.
- The select command can be used for selection as well as projection

DATA QUERY LANGUAGE-SELECT

- Syntax:

```
select * from <tablename>;
```

→selects all fields from all records i.e. the content of the entire table. * represents all fields.

- Example:

```
select * from student;
```

DATA QUERY LANGUAGE-SELECT

- Syntax:

```
select attrib1, attrib2, …. from <tablename>;
```

→selects named fields from all records i.e. the content of specific columns from all records. This is projection.

- Example:

```
select roll_no, name  from student;
```

Data Query Language-Select

- Syntax:

  select * from <tablename> where <condition>;

→selects all fields from records satisfying the given condition i.e. all columns from particular records. This is selection.

- Example:

  select * from student where marks between 60 and 80;

Data Query Language-Select

- Syntax:

```
select attrib1, attrib2,… from <tablename>
where <condition>;
```

→selects selected fields from records satisfying the given condition i.e. particular columns from particular records. This is selection+projection.

- Example:

```
select roll_no, name from student
where marks between 60 and 80;
```

QUERY FROM MULTIPLE TABLES

Example

- Table 1→ Employees

  Attributes→emp_code, name, dept_code, dt_of_join, city_code, basic_sal
- Table 2→ Department

  Attributes→dept_code, dept_name, head_code, no_of_emps
- Table 3→ City

  Attributes→ city_code, city_name, city_status
- Table 4→ Vendors

  Attributes→vend_code,vname, vcity_code,vaddress

Query from Multiple Tables- Horizontal Joins

Example

Displaying names of employees along with their city names and department names for employees with basic salary greater than 50,000.

```
select employees.name, department.dept_name, city.city_name
from employees, department, city
where employees.dept_code=department.dept_code
and     employees.city_code=city.city_code
and     employees.basic_sal>50000;
```

Query from multiple tables-Nested Select and Vertical Join

Example

Displaying names of cities which have no employees

```
select city_name from city
where city_code in (select city_code from city
                                        minus
                            select city_code from employees);
```

Query From Multiple Tables-Nested Select And Vertical Join

Example

Displaying names of cities which have either employees or vendors

```
select city_name from city
where city_code in (select city_code from employees
                             union
                     select vcity_code from vendors);
```

Query from Multiple Tables-Nested Select and Vertical Join

Example

Displaying names of cities which have both employees and vendors

```
select city_name from city
where city_code in (select city_code from employees
                                        intersect
                              select vcity_code from vendors);
```

- Till we meet again in the next class……….