SQL

- Instructor

  Dr. Sanam Shahla Rizvi

  PhD in Information and Communication from Ajou University, Korea

SELECT Statement

Retrieving records from database

Basic SELECT Statement

- Syntax:

```
SELECT * |{ [DISTINCT] column
   | expression [alias], ...} FROM table;
```

- For example:
  - Select all columns in the table:

    ```
    SELECT * FROM departments;
    ```

  - Specify the column names to select

    ```
    SELECT did, dname
      FROM departments;
    ```

Select with Arithmetic Expression

- You can include *, /, +, - in the SELECT statements

  -
    ```
    SELECT last_name, CGPA, CGPA + 2.0
       FROM students;
    ```

  -
    ```
    SELECT last_name, CGPA, CGPA / 4
       FROM students;
    ```

ALIAS

- Change the displayed name of a column

  -
    ```
    SELECT last_name AS "name"
      FROM students;
    ```

- Write a SELECT statement to select a column called "Quarter CGPA" that is computed as CGPA /4.

Concatenation

- Link the value of columns to other Columns
  ```
  SELECT first_name | | last_name
     AS “Full Name”
     FROM students;
  ```

- Add a string to the concatenation
  ```
  SELECT last_name | | ' studies in ' | |
  department_id AS “Description”
     FROM students;
  ```

DISTINCT

- Select unique records only

  -
    ```
    SELECT department_id
      FROM students;
    ```

  -
    ```
    SELECT DISTINCT department_id
      FROM students;
    ```

WHERE Clause

Specifying which rows to retrieve

WHERE Clause

- Syntax:
  - SELECT * | { [DISTINCT] column | expression [alias], …}

    FROM table

    WHERE conditions;
- Retrieve the information about CS department:
  - SELECT *

    FROM departments

    WHERE department_id = ‘CS’;
- Here the string in condition clause is *case sensitive*

Comparison Operator

=, >, >=, <, <=, <>

- Examples:
  - SELECT * FROM students WHERE CGPA >= 3.3;
  - SELECT * FROM students WHERE CGPA < 2.0;
- Try other comparison operators by yourself

More Conditions

- BETWEEN
  - E.g. WHERE cgpa BETWEEN 3.5 AND 4.0
- IN
  - E.g. WHERE department_id
    IN (‘EE’, ‘CS’)
- LIKE
  - E.g. WHERE first_name LIKE ‘%i%’
  - E.g. WHERE first_name LIKE ‘_i%’
  - %: zero or more
  - _: exactly one
- IS NULL
  - E.g. WHERE last_name IS NULL

Logical Conditions

- AND
  - E.g. WHERE CGPA >=3.5
    AND department_id = ‘CS’
- OR
  - E.g. WHERE CGPA >=3.5
    OR department_id = ‘CS’
- NOT
  - E.g. WHERE department_id
    NOT IN (‘CS’, ‘EE’)

Using Parentheses

- Try the following:

  -
    ```
    SELECT * FROM students
     WHERE department_id = ‘CS’
     OR department_id = ‘EE’
     AND CGPA > 3.5;
    ```

  -
    ```
    SELECT * FROM students
     WHERE (department_id = ‘CS’
     OR department_id = ‘EE’)
     AND CGA > 3.5;
    ```
- The results are different, because AND has a higher precedence than OR.

ORDER BY Clause

Sort the result by the specific column

ORDER BY Clause

- Sort the result by the specific column
  - ASC (default): ascending
  - DESC: descending

- Examples:
  - SELECT * FROM students
    ORDER BY CGPA;

  - SELECT * FROM students
    ORDER BY CGPA DESC;

More about ORDER BY Clause

- Sort by alias column name

  ```
  SELECT first_name,CGPA*0.8 AS wCGPA
  FROM students
  ORDER BY wCGPA
  ```

- Sort by Multiple columns

  ```
  SELECT * FROM students
  ORDER BY CGPA ASC, student_id DESC
  ```

This ASC can be omitted

Exercise

- Create queries for the following:
  - Display the first_name and email of students from the CS department.
  - Display “Quarter CGPA” that is computed as CGPA /4.
  - Display first_name and CGPA for all students whose CGPA is in the range of 3 and 4 sorted by cgpa.
  - Show all information of students whose department is CS or EE and CGPA is not 3.34 or 3.9.

Solutions

- SELECT first_name, email
- FROM students
- WHERE department = ‘CS’;

- SELECT CGPA/4 AS “Quarter CGPA”
- FROM students;

- SELECT first_name, CGPA
- FROM students
- WHERE CGPA BETWEEN 3.0 AND 4.0;

- SELECT * FROM students
- WHERE (department_id = ‘CS’
- OR department_id = ‘EE’)
- AND CGPA NOT IN (3.34, 3.9);