

SQL Practice – 1

Keep in Mind That...

- SQL keywords are NOT case sensitive: `select` is the same as `SELECT`

The SQL SELECT Statement

The `SELECT` statement is used to select data from a database.

Syntax

```
SELECT column1, column2, ...
FROM table_name;
```

Here, `column1`, `column2`, ... are the *field names* of the table you want to select data from.

The `table_name` represents the name of the *table* you want to select data from.

Select ALL columns

If you want to return all columns, without specifying every column name, you can use the `SELECT *` syntax:

[What is the Difference Between SELECT Statement & SELECT DISTINCT Statement](#)

The SQL SELECT DISTINCT Statement

The `SELECT DISTINCT` statement is used to return only distinct (different) values.

SQL Aggregate Functions

An aggregate function is a function that performs a calculation on a set of values, and returns a single value.

Aggregate functions are often used with the `GROUP BY` clause of the `SELECT` statement. The `GROUP BY` clause splits the result-set into groups of values and the aggregate function can be used to return a single value for each group.

The most commonly used SQL aggregate functions are:

- `MIN()` - returns the smallest value within the selected column
- `MAX()` - returns the largest value within the selected column
- `COUNT()` - returns the number of rows in a set
- `SUM()` - returns the total sum of a numerical column
- `AVG()` - returns the average value of a numerical column

Aggregate functions ignore null values (except for `COUNT()`).

We will go through the aggregate functions above in the next chapters.

Some of The Most Important SQL Commands

- `SELECT` - extracts data from a database
- `UPDATE` - updates data in a database
- `DELETE` - deletes data from a database
- `INSERT INTO` - inserts new data into a database
- `CREATE DATABASE` - creates a new database
- `ALTER DATABASE` - modifies a database
- `CREATE TABLE` - creates a new table
- `ALTER TABLE` - modifies a table
- `DROP TABLE` - deletes a table
- `CREATE INDEX` - creates an index (search key)
- `DROP INDEX` - deletes an index

GROUP BY

The GROUP BY Statement in SQL is used to arrange identical data into groups with the help of some functions. If a particular column has same values in different rows then it will arrange these rows in a group.

Important:

- GROUP BY clause is used with the SELECT statement.
- In the query, GROUP BY clause is placed after the WHERE clause.
- Where clause use here only if it needed according to the given question.
- In the query, GROUP BY clause is placed before ORDER BY clause if used any.

Exercise

- a. Display the total number of modules in each course?
- b. How many students are there for each course? Rename the count as 'Number of Student'.
- c. What is the number of Modules offered by each course in each academic year?
- d. What is the number of Modules in Semester 02 only?
- e. Sort the results of Question (d.) according to the ascending order of CID.

The SQL HAVING Clause

The **HAVING** clause was added to SQL because the **WHERE** keyword cannot be used with aggregate functions.

HAVING Syntax

```
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
HAVING condition
ORDER BY column_name(s);
```

HAVING CLAUSE

The HAVING Clause enables you to specify conditions that filter which group results appear in the results.

The HAVING clause was added to SQL because the WHERE keyword could not be used with aggregate functions.

Syntax

```
SELECT Column1, Column2, aggregate_function (aggregate_ Column)
FROM table
[WHERE conditions]
GROUP BY Column1, Column2
HAVING condition;
```

Difference between WHERE clause and HAVING clause.

The main difference between WHERE and HAVING clause comes when used together with GROUP BY clause

- WHERE is used to filter rows before grouping
- HAVING is used to exclude records after grouping.

When SQL statements have both a WHERE clause and HAVING clause, keep in mind the WHERE clause is applied first, then the results grouped, and finally, the groups filtered according to the HAVING clause.

Exercise

- a) Display the number of students for each course? List the Course_ID of courses only if there are less than 10 students for the course.
- b) List the Course_ID and the number of modules offered for each course. Display only the course ids which have more than 3 modules offered in it. Sort the result according to the ascending order of the module count.
- c) Display the course id, academic year and the number of modules offered. The number of modules offered should be less than 10.
- d) List the courses that offer more than 2 modules for year 3 students?