SQL TASK – Questions

**Q.** What is SQL?

**A.** SQL is the language used to interact and manipulate data in database using special commands syntax.

**Q.** What are the different types of SQL commands?

**A.**

1. Data Query language (DQL)

SELECT statement queries and retrieves data from a database.

2. Data Modification language (DML)

DML, is a set of SQL commands that are used to manipulate data within database tables or query.

DML commands are: SELECT, INSERT, UPDATE, and DELETE.

3. Data Definition language (DDL)

DDL is responsible for Setting up and configuring the DB itself.

DDL commands are: CREATE, ALTER ,DROP, and TRUNCATE.

4. Data Control Language (DCL)

DCL is The ability to give/take from someone a DB access

DCL commands are: GRANT / REVOKE

5. Transaction Control Language (TCL)

TCL instructions are used to handle database transactions.

COMMIT: Finalizes a transaction, making all changes permanent.

ROLLBACK: Reverts the database to its state before the beginning of a transaction.

SAVEPOINT: Sets points within transactions to which you can later roll back.

**Q.** What is a primary key in SQL?

**A.** A primary key (PK) is a column in a table defined as a unique identifier for each row and cannot be empty.

**Q.** What is a foreign key in SQL?

**A.** A foreign key (FK) is simply a PK in another table, used to establish relations between tables using both these two keys.

**Q.** What is a composite key?

**A.** A composite key is made by the combination of two or more columns in a table that can be used to as a PK in a table.

**Q.** What is a database?

**A.** A **database** is an organized collection of data stored in a computer system and usually controlled by a database management system (DBMS).

**Q.** What is a constraint in SQL?

**A.** SQL constraints are used to specify rules for the data in a table. e.g., a PRIMARY KEY constraint indicates that this column is a primary key, while NOT NULL

constraint tells us this column cannot be empty.

**Q.** GROUP BY VS ORDER BY

**A.** The GROUP BY clause groups the rows specified in the expression, while the ORDER BY clause sorts data in ascending or descending order.

**Q.** Describe the difference between WHERE and HAVING in SQL.

**A.** The HAVING clause can be used with aggregates e.g. GROUP BY, while the WHERE clause cannot, it only works on rows dat**a.**

**Q.** What is a Stored Procedure?

**A.** A stored procedure (SP) in SQL is a group of SQL queries that can be saved and reused multiple times.

SP is extremely useful as it reduces the need for rewriting SQL queries. It enhances efficiency, reusability, and security in database management.

**Q.** What are aggregate functions?

**A.** Aggregate functions are functions where the values of multiple rows are grouped as input on certain criteria to form a single value result of more significant meaning. For example, calculating total records in a column, the sum of a column of type INT, or it’s average.

**Q.** Explain the different types of joins with examples.

**A.**

Ok, let us consider the two tables below as follows:

**Student:**



**StudentCourse** :



SQL Join operation combines data or rows from two or more tables based on a common field between them.

SQL INNER JOIN

The [INNER JOIN](https://www.geeksforgeeks.org/sql-inner-join) keyword selects all rows from both the tables as long as the condition is satisfied.

SQL INNER JOIN using the tables provided above:

**SELECT StudentCourse.COURSE\_ID, Student.NAME, Student.AGE FROM Student  
INNER JOIN StudentCourse  
ON Student.ROLL\_NO = StudentCourse.ROLL\_NO**

**GO**

**Output:**

****

SQL LEFT JOIN

LEFT JOIN returns all the rows of the table on the left side of the join and matches rows for the table on the right side of the join.

For the rows for which there is no matching row on the right side, the result-set will contain null.

SQL LEFT JOIN using the tables provided above:

**SELECT** Student.NAME,StudentCourse.COURSE\_ID   
**FROM** Student  
**LEFT JOIN** StudentCourse   
**ON** StudentCourse.ROLL\_NO = Student.ROLL\_NO

**GO**

**Output:**

****

SQL RIGHT JOIN

[RIGHT JOIN](https://www.geeksforgeeks.org/sql-right-join) returns all the rows of the table on the right side of the join and matching rows for the table on the left side of the join.

It is similar to LEFT JOIN For the –rows for which there is no matching row on the left side, the result-set will contain null.

SQL RIGHT JOIN using the tables provided above:

**SELECT** Student.NAME,StudentCourse.COURSE\_ID   
**FROM** Student  
**RIGHT JOIN** StudentCourse   
**ON** StudentCourse.ROLL\_NO = Student.ROLL\_NO

**GO**

**Output:**

