

WORKSHEET 6 SQL

Q1 and Q2 have one or more correct answer. Choose all the correct option to answer your question.

1. Which of the following are TCL commands?
 - A. Commit
 - B. Select
 - C. Rollback
 - D. Savepoint

ANS: All are SQL commands not TCL Commands

2. Which of the following are DDL commands?
 - A. Create
 - B. Select
 - C. Drop
 - D. Alter

ANS: A, C and D

Q3 to Q10 have only one correct answer. Choose the correct option to answer your question.

3. Which of the following is a legal expression in SQL?
 - A. SELECT NULL FROM SALES;
 - B. SELECT NAME FROM SALES;
 - C. SELECT * FROM SALES WHEN PRICE = NULL;
 - D. SELECT # FROM SALES;

ANS: B

4. DCL provides commands to perform actions like-
 - A. Change the structure of Tables
 - B. Insert, Update or Delete Records and Values
 - C. Authorizing Access and other control over Database
 - D. None of the above

ANS: C

5. Which of the following should be enclosed in double quotes?
 - A. Dates
 - B. Column Alias
 - C. String
 - D. All of the mentioned

Ans: C

6. Which of the following command makes the updates performed by the transaction permanent in the database?
 - A. ROLLBACK
 - B. COMMIT
 - C. TRUNCATE
 - D. DELETE

ANS: B

7. A subquery in an SQL Select statement is enclosed in:
 - A. Parenthesis - (...).
 - B. brackets - [...].
 - C. CAPITAL LETTERS.
 - D. braces - {...}.

ANS: A

8. The result of a SQL SELECT statement is a :-
- A. FILE
 - B. REPORT
 - C. TABLE
 - D. FOR

Ans: C

9. Which of the following do you need to consider when you make a table in a SQL?
- A. Data types
 - B. Primary keys
 - C. Default values
 - D. All of the mentioned

ANS: D

10. If you don't specify ASC and DESC after a SQL ORDER BY clause, the following is used by ____?
- A. ASC
 - B. DESC
 - C. There is no default value
 - D. None of the mentioned

ANS: A

Q11 to Q15 are subjective answer type questions, Answer them briefly.

11. What is denormalization?

Denormalization is the process of intentionally adding redundant data to a database table to improve query performance or simplify data retrieval. In a normalized database, data is organized into tables to minimize data redundancy and ensure data consistency. However, this can result in more complex queries that require joining multiple tables to retrieve the required data.

In some cases, denormalization can be used to improve query performance by reducing the number of tables that need to be joined. For example, if a database frequently retrieves data from multiple related tables, denormalization can be used to create a new table that combines the data from these tables into a single table. This can make queries simpler and faster to execute, at the cost of increased data redundancy and the potential for data inconsistencies.

However, denormalization should be used judiciously and only after careful consideration of the potential benefits and drawbacks. It can make database maintenance more complex, increase the risk of data inconsistencies, and require additional storage space. Therefore, denormalization should only be used when it can provide a significant performance benefit and when the risks are carefully managed.

12. What is a database cursor?

In a database system, a cursor is a mechanism that allows the traversal of records in a result set, one record at a time. It provides a way to iterate over the results of a database query, retrieve data from the result set, and perform operations on the retrieved data.

A database cursor can be seen as a pointer to a specific record in a result set. The cursor can be moved forward or backward in the result set, and each time it moves, it points to a new record. Cursors are typically used in programming languages or database management systems that provide procedural extensions, such as PL/SQL in Oracle or T-SQL in Microsoft SQL Server.

There are two types of cursors: static and dynamic. A static cursor retrieves a snapshot of the result set when it is opened, and any changes made to the data while the cursor is open are not reflected in the result set. A

dynamic cursor, on the other hand, retrieves data from the result set as it is accessed, so changes made to the data while the cursor is open are reflected in the result set.

13. What are the different types of the queries?

In SQL, there are several types of queries that can be used to retrieve data from a database. Some of the most common types of queries are:

SELECT query: The SELECT query is the most basic type of query in SQL, and is used to retrieve data from one or more tables in a database. It can be used to retrieve all rows and columns from a table, or to retrieve a subset of rows and columns based on specific conditions.

INSERT query: The INSERT query is used to add new rows to a table in a database. It specifies the table to insert into and the values to be inserted.

UPDATE query: The UPDATE query is used to modify existing rows in a table. It specifies the table to update and the new values to be set.

DELETE query: The DELETE query is used to remove rows from a table. It specifies the table to delete from and the conditions that must be met for a row to be deleted.

JOIN query: The JOIN query is used to combine data from two or more tables in a database. It specifies the tables to be joined and the columns to be used as the join condition.

SUBQUERY or nested query: A subquery is a query within another query. It is used to retrieve data based on the results of another query.

AGGREGATE query: The AGGREGATE query is used to perform calculations on data in a table. It can be used to calculate counts, sums, averages, maximum and minimum values, and other aggregate functions.

14. Define constraint?

In SQL, a constraint is a rule or restriction that is imposed on a table column or a set of columns, which ensures data integrity and consistency in the database. Constraints can be applied to one or more columns in a table and can be used to enforce various rules on the data that is inserted, updated or deleted from the table.

Some common types of constraints are:

Primary Key constraint: A primary key constraint is used to uniquely identify each row in a table. It ensures that the values in the primary key column are unique and not null.

Foreign Key constraint: A foreign key constraint is used to enforce referential integrity between tables. It ensures that the values in a column of one table match the values in the primary key column of another table.

Unique constraint: A unique constraint is used to ensure that the values in a column or a set of columns are unique. It can be used to prevent duplicate data from being inserted into a table.

Not Null constraint: A not null constraint is used to ensure that a column does not contain null values. It can be used to enforce mandatory data entry for a column.

Check constraint: A check constraint is used to enforce a specific condition on the data that is inserted, updated or deleted from a table. It can be used to ensure that the data meets certain criteria, such as a range of values or a specific format.

Constraints play an important role in maintaining the data integrity and consistency in a database, and are an essential part of creating robust and reliable database applications.

15. What is auto increment?

Auto Increment is a feature in SQL that allows a unique numeric value to be automatically generated and assigned to a column whenever a new row is inserted into a table. It is commonly used to generate primary key values for tables where each row needs to have a unique identifier.

To enable auto-increment in a column, the column must be defined as an integer data type (e.g. INT, BIGINT) and marked as the primary key or with a unique index. Then, the AUTO_INCREMENT attribute must be set for that column.

For example, to create a table with an auto-increment primary key column, the following SQL code can be used:

```
CREATE TABLE customers (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  name VARCHAR(50),  
  email VARCHAR(50)  
);
```

In this example, the id column is defined as an INT data type with the AUTO_INCREMENT attribute, which means that each time a new row is inserted into the customers table, the id column will be automatically populated with a unique integer value.

Auto Increment is a useful feature in SQL because it eliminates the need for manually assigning primary key values to each new row, which can be time-consuming and error-prone. It also ensures that each row in a table has a unique identifier, which is important for maintaining data integrity and consistency.
