**✅ Part 1: SQL Theory Questions (10 Marks)**

**1. What is a Database?**

A **database** is an organized collection of data that can be easily accessed, managed, and updated. It allows users to store information in a structured way for quick retrieval and manipulation.

**2. What is a Table in a Database?**

A **table** is a structure within a database that stores data in **rows** and **columns**. Each row represents a single record, and each column represents a specific field or attribute of that record.

**3. What is SQL?**

**SQL** (Structured Query Language) is a standard programming language used to manage and manipulate relational databases. It allows users to create, read, update, and delete data.

**4. What is a Primary Key?**

A **primary key** is a column (or a combination of columns) that uniquely identifies each record in a table. It ensures that no two rows have the same value in the primary key column.

**5. What is a SELECT query?**

A **SELECT** query is used to retrieve data from one or more tables in a database. Example:

sql

CopyEdit

SELECT Name FROM Students;

**6. What is a WHERE clause used for?**

The **WHERE** clause is used to filter records based on specific conditions. It returns only rows that meet the condition.  
Example:

sql

CopyEdit

SELECT \* FROM Students WHERE Age > 15;

**7. What is the purpose of the DISTINCT keyword?**

The **DISTINCT** keyword is used to return **only unique (non-duplicate)** values in the result set.  
Example:

sql

CopyEdit

SELECT DISTINCT GradeLevel FROM Students;

**8. What is an Alias in SQL?**

An **alias** is a temporary name given to a column or table in a SQL query using the AS keyword.  
Example:

sql

CopyEdit

SELECT Name AS StudentName FROM Students;

**9. What is the ORDER BY clause used for?**

The **ORDER BY** clause is used to sort the result set by one or more columns in ascending (ASC) or descending (DESC) order.  
Example:

sql

CopyEdit

SELECT \* FROM Students ORDER BY Age DESC;

**10. What are SQL Functions? Give examples of any two.**

**SQL functions** are built-in operations that perform calculations on data.

**Examples:**

* COUNT() – counts the number of rows

sql

CopyEdit

SELECT COUNT(\*) FROM Students;

* AVG() – calculates the average of a numeric column

sql

CopyEdit

SELECT AVG(Age) FROM Students;

**Part 2: SQL Practice – SQL Script File (10 Marks)**

Write the SQL queries for each question in your `.sql` file:

----1. Create a table called Students with columns as shown above.

CREATE TABLE Students (

StudentID INT PRIMARY KEY,

Name VARCHAR(100),

Age INT,

Gender VARCHAR(10),

GradeLevel VARCHAR(20),

School VARCHAR(100),

City VARCHAR(50),

AverageScore DECIMAL(5,2)

);

-----2. Insert the sample records provided into the Students table.

INSERT INTO Students (StudentID, Name, Age, Gender, GradeLevel, School, City, AverageScore)

VALUES

(1, 'Alice Njeri', 16, 'Female', 'Form 2', 'Green Hill Academy', 'Nairobi', 78.5),

(2, 'Brian Otieno', 17, 'Male', 'Form 3', 'Kisumu Boys High', 'Kisumu', 84.3),

(3, 'Cynthia Wambui', 15, 'Female', 'Form 2', 'Ridgeways Girls', 'Nairobi', 91.0),

(4, 'David Mwangi', 16, 'Male', 'Form 2', 'Alliance High', 'Kikuyu', 74.0),

(5, 'Eva Akinyi', 18, 'Female', 'Form 4', 'Moi Girls Eldoret', 'Eldoret', 88.2),

(6, 'Felix Kiptoo', 17, 'Male', 'Form 3', 'Kapsabet Boys', 'Kapsabet', 82.7),

(7, 'Grace Muthoni', 14, 'Female', 'Form 1', 'Starehe Girls', 'Thika', 77.1),

(8, 'Hassan Abdalla', 15, 'Male', 'Form 2', 'Mombasa Secondary', 'Mombasa', 69.8),

(9, 'Irene Cherono', 16, 'Female', 'Form 3', 'Kabarak High', 'Nakuru', 86.4),

(10, 'John Kamau', 17, 'Male', 'Form 4', 'Lenana School', 'Nairobi', 80.0),

(11, 'Kevin Omondi', 15, 'Male', 'Form 2', 'Maseno School', 'Kisumu', 72.5);

3. Display all records from the Students table.

SELECT\*

FROM Students

4. Display only the distinct grade levels from the Students table.

SELECT DISTINCT GradeLevel

FROM Students;

----5. Show student names and their grade levels using column aliases.

SELECT Name AS StudentName, GradeLevel AS ClassLevel

FROM Students;

----6. Display names and ages of students older than 15.

SELECT Name, Age

FROM Students

WHERE Age > 15;

----7. Display all students sorted by Age in ascending order.

SELECT \*

FROM Students

ORDER BY Age ASC;

----8. Display all female students sorted by Name in descending order.

SELECT \*

FROM Students

WHERE Gender = 'Female'

ORDER BY Name DESC;

----9. Count the number of students in the table.

SELECT COUNT(\*) AS TotalStudents

FROM Students;

----10. Show the average age of students using an aggregate function.

SELECT AVG(Age) AS AverageAge

FROM Students;