Q1. What is a database? Differentiate between SQL and NoSQL databases.

Ans: - A database is an organized collection of data, so that it can be easily accessed and managed.

SQL or Structured Query Language is used to operate on the data stored in a database. SQL depends on relational algebra and tuple relational calculus.

The **main purpose** of the database is to operate a large amount of information by storing, retrieving, and managing data.

SQL	NoSQL
Relational database management system (RDBMS)	Non-relational or distributed database system.
These databases have fixed or static or predefined schema	They have a dynamic schema
These databases are best suited for complex queries	These databases are not so good for complex queries

Q2. What is DDL? Explain why CREATE, DROP, ALTER, and TRUNCATE are used with an example.

Ans - DDL stands for Data Definition Language. It is a subset of SQL (Structured Query Language) that is used to define and manage the structure of a database and its objects.

- CREATE : - The CREATE statement is used to create a new database object, such as a table or view. It specifies the object's name, structure, and any associated constraints. Here's an example of creating a table named "employees":

```
CREATE TABLE employees (id INT PRIMARY KEY,  name VARCHAR(100),age INT, salary DECIMAL(10,2));
```

- DROP: The DROP statement is used to remove an existing database object. It permanently deletes the object and its associated data. Here's an example of dropping a table named "employees":

```
DROP TABLE employees;
```

- ALTER: The ALTER statement is used to modify the structure of an existing database object. It allows you to add, modify, or delete columns, constraints, or indexes. Here's an example of adding a new column named "email" to the "employees" table:

```
ALTER TABLE employees
ADD COLUMN email VARCHAR(255);
```

- TRUNCATE: The TRUNCATE statement is used to delete all data from a table, while keeping the table structure intact. Unlike the DROP statement, TRUNCATE does not delete the table itself. Here's an example of truncating the "employees" table:

```
TRUNCATE TABLE employees;
```

Q3. What is DML? Explain INSERT, UPDATE, and DELETE with an example.

Ans - DML stands for Data Manipulation Language. It is a subset of SQL (Structured Query Language) that is used to manipulate data within a database. DML statements are used to insert, update, and delete records in database tables.

- INSERT: The INSERT statement is used to add new records into a table. It specifies the table name and the values to be inserted into the corresponding columns. Here's an example of inserting a new record into a table named "employees":

```
INSERT INTO employees (id, name, age, salary)
VALUES (1, 'John Doe', 30, 5000.00);
```

- UPDATE: The UPDATE statement is used to modify existing records in a table. It specifies the table name, the columns to be updated, and the new values. You can also include a WHERE clause to specify the condition for updating specific records. Here's an example of updating the salary of an employee with an ID of 1:

```
UPDATE employees
SET salary = 6000.00
WHERE id = 1;
```

- DELETE: The DELETE statement is used to remove records from a table. It specifies the table name and can include a WHERE clause to specify the condition for deleting specific records. Here's an example of deleting an employee with an ID of 1:

```
DELETE FROM employees
WHERE id = 1;
```

Q4. What is DQL? Explain SELECT with an example.

Ans - DQL stands for Data Query Language. It is a subset of SQL (Structured Query Language) that is used to retrieve data from a database. DQL statements are primarily focused on querying and fetching data from database tables.

The SELECT statement is used to retrieve data from one or more tables in a database. It allows you to specify the columns to retrieve, the table(s) to query, and any conditions or filters to apply.

```
SELECT id, name, age, salary
FROM employees;
```

- ORDER BY clause: This clause is used to sort the result set based on one or more columns. For example, to retrieve employees sorted by their age in descending order:

```
SELECT id, name, age, salary
FROM employees
ORDER BY age DESC;
```

- Joins: The SELECT statement can also be used to retrieve data from multiple tables by performing joins. Joins allow you to combine data from different tables based on related columns. Here's an example of retrieving data from two tables, "employees" and "departments," using an inner join:

```
SELECT e.id, e.name, d.department_name
FROM employees e
JOIN departments d ON e.department_id = d.id;
```

Q5. Explain Primary Key and Foreign Key.

Ans - Primary Key:

A primary key is a column or a set of columns in a database table that uniquely identifies each record within the table. It serves as a unique identifier for a particular row and ensures the integrity and uniqueness of the data.

- Uniqueness: Every value in the primary key column(s) must be unique.
- Single-value constraint: A primary key can consist of a single column or multiple columns, forming a composite primary key.
- Primary Key Constraints: Primary keys are often defined using constraints, such as the PRIMARY KEY constraint in SQL, to enforce the uniqueness and non-nullability of the key.

Foreign Key:

A foreign key is a column or a set of columns in a database table that establishes a link or relationship between two tables. It represents a reference to the primary key of another table, thereby establishing a relationship between the two tables.

- Referential Integrity: Foreign keys maintain referential integrity by ensuring that the values in the foreign key column(s) match the values of the primary key column(s) in the referenced table.
- Multiple Foreign Keys: A table can have multiple foreign keys, each representing a different relationship with other tables.

Q6. Write a python code to connect MySQL to python. Explain the cursor() and execute() method.

Ans - cursor():

- The cursor() method is used to create a cursor object in Python for interacting with the MySQL database. A cursor is like a handle or a control structure that allows you to execute SQL queries and retrieve the results. It acts as a pointer that moves through the result set returned by the database.
-

Execute():

The execute() method is used to execute SQL statements or queries on the MySQL database using the cursor object. It takes the SQL query as a parameter and sends it to the database for execution.

```
import mysql.connector

# Connect to the MySQL database
cnx = mysql.connector.connect(
    user='your_username',
    password='your_password',
    host='localhost',  # or '127.0.0.1'
    database='your_database')

# Create a cursor object to interact with the database
cursor = cnx.cursor()

# Execute a SQL query
query = "SELECT * FROM your_table"
cursor.execute(query)
```

Q7. Give the order of execution of SQL clauses in an SQL query.

Ans –

- FROM: The FROM clause specifies the table(s) from which the data will be retrieved. It is the first clause to be evaluated in the query.
- WHERE: The WHERE clause is used to filter the rows based on specified conditions. It follows the FROM clause and is executed after the FROM clause retrieves the initial set of rows.
- GROUP BY: The GROUP BY clause is used to group rows based on specified columns. It is executed after the WHERE clause filters the rows and before any aggregate functions are applied.
- HAVING: The HAVING clause is used to filter the groups generated by the GROUP BY clause. It is executed after the GROUP BY clause and before the SELECT clause.

- SELECT: The SELECT clause is used to specify the columns to be included in the result set. It is executed after the previous clauses have filtered and grouped the rows.

- ORDER BY: The ORDER BY clause is used to sort the result set based on specified columns. It is executed after the SELECT clause retrieves the result set.

- LIMIT/OFFSET: The LIMIT/OFFSET clause is used to limit the number of rows returned by the query or to skip a specified number of rows. It is executed after the ORDER BY clause.