1. **How to Create Table student write an SQL Query?**

```
→CREATE TABLE student(s_id int AUTO_INCREMENT,
                      s_name varchar(100),
                       s_age varchar(100),
                       PRIMARY KEY(s_id));
```

2. How to Create a Exam table with Foreign key on roll no write a SQL Query?

```
→CREATE TABLE student(s_id int AUTO_INCREMENT,
          s_name varchar(100),
          s_age varchar(100),
          PRIMARY KEY(s_id),
          roll_no int,
          FOREIGN KEY(roll_no) REFERENCES exam(roll_no));
```

3. What is SQL Key Constraints? Write an Example of SQL Key Constraints?

---> SQL constraints are used to specify rules for the data in a table.

Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

The following constraints are commonly used in SQL:

NOT NULL - Ensures that a column cannot have a NULL value

UNIQUE - Ensures that all values in a column are different

PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

4. What is SQL View Create a View of Student Table?

```
→ select * from table_name;
```

5. How to Create a Table user write a SQL query?

```
→CREATE TABLE student(s_id int AUTO_INCREMENT,
                      s_name varchar(100),
                       s_age varchar(100),
                       PRIMARY KEY(s_id));
```

6. What is SQL and How to Create a table with Foreign Key?

→ Structured Query Language (SQL) is a standardized programming language that is used to manage relational databases and perform various operations on the data in them.

→
```
CREATE TABLE student(s_id int AUTO_INCREMENT,
            s_name varchar(100),
            s_age varchar(100),
            PRIMARY KEY(s_id),
            roll_no int,
            FOREIGN KEY(roll_no) REFERENCES exam(roll_no));
```

8. What is the Difference between DBMS and RDBMS?

DBMS stands for Database Management System, and RDBMS is the acronym for the Relational Database Management system. In DBMS, the data is stored as a file, whereas in RDBMS, data is stored in the form of tables. To know what is the difference between RDBMS and DBMS, check out the table below.

Difference between RDBMS and DBMS

RDBMS	**DBMS**
Data stored is in table format	Data stored is in the file format
Multiple data elements are accessible together	Individual access of data elements
Data in the form of a table are linked together	No connection between data

Normalisation is not achievable	There is normalisation
Support distributed database	No support for distributed database
Here, redundancy of data is reduced with the help of key and indexes in RDBMS	Data redundancy is common
RDBMS supports multiple users	DBMS supports a single user
It features multiple layers of security while handling data	There is only low security while handling data
The software and hardware requirements are higher	The software and hardware requirements are low

CHECK - Ensures that the values in a column satisfies a specific condition

DEFAULT - Sets a default value for a column if no value is specified

CREATE INDEX - Used to create and retrieve data from the database very quickly.

9. **What is Normalization?**

Normalization is a technique for organizing data in a database. It is important that a database is normalized to minimize redundancy (duplicate data) and to ensure only related data is stored in each table. It also prevents any issues stemming from database modifications such as insertions, deletions, and updates.

Data stored is in table format	Data stored is in the file format
Multiple data elements are accessible together	Individual access of data elements
Data in the form of a table are linked together	No connection between data
Normalisation is not achievable	There is normalisation

The stages of organization are called **normal forms**. In this tutorial we will be redesigning a database for a construction company and ensuring that it satisfies the three normal forms:

First Normal Form (1NF):

- Data is stored in tables with rows uniquely identified by a primary key
- Data within each table is stored in individual columns in its most reduced form
- There are no repeating groups

Second Normal Form (2NF):

- Everything from 1NF
- Only data that relates to a table’s primary key is stored in each table

Third Normal Form (3NF):

- Everything from 2NF

- There are no in-table dependencies between the columns in each table