Q1: What do you understand by database?

* A database is an organized collection of structured information, or data, typically stored electronically in a computer system.
* A database is usually controlled by a database management system (DBMS).
* Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient.
* The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

Q2: What is normalization?

* Normalization is the process of organizing data in a database.
* It includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.
* It was first proposed by British computer scientist Edgar F. Codd as part of his relational model.

Q3: What is difference between DBMS and RDBMS?

|  |  |
| --- | --- |
| Database Management System | Relational Database Management System |
| DBMS stores data as file. | RDBMS stores data in tabular form. |
| Data elements need to access individually. | Multiple data elements can be accessed at the same time. |
| Normalization is not present. | Normalization is present. |
| Data fetching is slower for the large amount of data. | Data fetching is fast because of relational approach. |
| Examples: XML, Window Registry, Forxpro, dbaseIIIplus etc. | Examples: MySQL, PostgreSQL, SQL Server, Oracle, Microsoft Access etc. |

Q4: What is MF Cod Rule of RDBMS Systems?

* Codd's twelve rules are a set of thirteen rules (numbered zero to twelve) proposed by Edgar F. Codd, a pioneer of the relational model for databases, designed to define what is required from a database management system in order for it to be considered relational, i.e., a relational database management system (RDBMS).
* It uses the concept of relations to represent each and every file. Relations are Two-Dimensional Tables.
* Relational Algebra and Relational Calculus are used to process the relations manually. Many database languages are oriented toward handling relational data models.

Q5: What do you understand by Data Redundancy?

* Data Redundancy is the existence of data that is additional to the actual data and permits correction of errors in stored or transmitted data.
* The additional data can simply be a complete copy of the actual data (a type of repetition code), or only select pieces of data that allow detection of errors and reconstruction of lost or damaged data up to a certain level.

Q6: What is DDL Interpreter?

* A DDL (Data Definition Language) Interpreter is a component of a Database Management System (DBMS) that processes and executes DDL statements.
* Its primary function is to interpret DDL commands, such as CREATE TABLE, ALTER TABLE, and DROP TABLE, and generate the necessary metadata to define or modify database objects.
* In other words, the DDL Interpreter is responsible for translating high-level DDL statements into low-level, machine-understandable instructions that the DBMS can execute.
* This process enables the creation, modification, and deletion of database objects, such as tables, views, and indexes, which are essential for managing and querying data.

Q7: What is DML Compiler in SQL?

* DML (Data Manipulation Language) is a subset of SQL (Structured Query Language) statements used to modify data in a database.
* The primary DML statements in SQL are INSERT: Adds new rows to a table, UPDATE: Modifies existing rows in a table & DELETE: Deletes rows from a table.
* DML statements are contrasted with DDL (Data Definition Language) statements, which are used to define the structure of the database, such as creating or altering tables, indexes, and views.

Q8: What is SQL Key Constraints? Give an example.

* SQL key constraints are rules that define how data is inserted, updated, or deleted in a relational database to ensure data integrity and accuracy.
* Following are some examples of Key Constraints:
  + Primary Key: Ensures that each row in a table has a unique and non-null value.
  + Foreign Key: Enforces a link between the data in two tables, ensuring that the value in one table matches a value in another.
  + Unique Key: Ensures that all values in a column or a group of columns are unique across the table.
  + Not Null: Ensures that a column cannot have a NULL value.

Q9: What is save point? How to create a save point query?

* A SAVEPOINT in SQL is a way to set a point within a transaction to which you can later roll back. This allows for more granular control over transactions, giving you the ability to undo parts of the transaction without affecting the entire transaction.
* The SAVEPOINT command is used to create a savepoint. The ROLLBACK TO SAVEPOINT command is used to roll back to that savepoint, and the RELEASE SAVEPOINT command is used to remove a savepoint.

Q10: What is trigger and how to create a trigger in SQL?

* A trigger in SQL is a special kind of stored procedure that automatically executes or "fires" when certain events occur in a database, such as insertions, updates, or deletions.
* Triggers can be used to enforce business rules, validate data, maintain audit trails, and synchronize tables.
* Example – CREATE TRIGGER TRIGGER\_NAME [BEFORE/AFTER/INSTEAD OF] [INSERT/UPDATE/DELETE] ON TABLE\_NAME [FOR EACH ROW] [WHEN(condition)] BEGIN [Your trigger logic] END;