**DATABASE INTERVIEW QUESTIONS**

**What do you understand by ‘Database’?**

**Answer:** Database is an organized collection of related data where the data is stored and organized to serve some specific purpose.

**For Example**, A librarian maintain a database of all the information related to the books that are available in the library.

**Define DBMS.**

**Answer:**DBMS stands for Database Management System. It is a collection of application programs which allow the user to organize, restore and retrieve information about data efficiently and as effectively as possible.

Some of the popular DBMS’s are MySql, Oracle, Sybase, etc.

**Define RDBMS.**

**Answer:**Relational Database Management System(RDBMS) is based on a relational model of data that is stored in databases in separate tables and they are related to the use of a common column. Data can be accessed easily from the relational database using Structured Query Language (SQL).

**Enlist the advantages of DBMS.**

**Answer: The advantages of DBMS includes:**

* Data is stored in a structured way and hence redundancy is controlled.
* Validates the data entered and provide restrictions on unauthorized access to the database.
* Provides backup and recovery of the data when required.
* It provides multiple user interfaces.

**What is an index?**

**Definition:** An index is a special data structure associated with a table that improves the speed of data retrieval operations on the table.

**Explanation:** Indexes are used to quickly look up information in a database without having to search every row. They contain pointers to the table data. Indexes improve the speed of query operations but slow down updates, inserts and deletes.

**Example:** A indexes the values in the age column of a customer table to allow fast searches based on age.

**What is a foreign key?**

**Definition:** A foreign key is a column in a table that contains the primary key values from another table.

**Explanation:** Foreign keys are used to link data between tables. They are used to enforce referential integrity - to prevent invalid data in related tables.

**Example:** In an orders table, the customer\_id column is a foreign key that links to the ID primary key in the customers table.

**What is SQL?**

**Definition:** SQL stands for Structured Query Language. It is a language used to communicate with relational databases.

**Explanation:** SQL is the standard language for manipulating and querying data in relational databases. It allows you to create tables, insert, select, update and delete data. SQL queries are used to retrieve data from databases.

**Example:** Some common SQL queries are:

CREATE TABLE table\_name (column\_name TYPE);

INSERT INTO table\_name (column\_1, column\_2) VALUES (value\_1, value\_2);

SELECT \* FROM table\_name;

**What is a table?**

**Definition:** A table is a collection of related data held in a structured format within a database.

**Explanation:** Tables allow you to organize data into rows and columns. Each row represents a single record and each column represents a field in the record.

**Example:** A customers table with columns for name, age, address, etc. Each row represents one customer.

**What is normalization?**

**Definition:** Normalization is the process of organizing columns and tables to reduce redundancy and dependency in a relational database.

**Explanation:** The goal of normalization is to have each cell in a table contain a single value, and have each table contain data about only one entity. There are 3 normal forms - 1NF, 2NF, 3NF.

**Example:** In 1NF, there should be no repeating columns in a table. In 2NF, all columns depend on the primary key. In 3NF, columns should depend only on the primary key and not other non-key columns.

**What do you understand by Data Redundancy?**

**Answer:**Duplication of data in the database is known as data redundancy. As a result of data redundancy, duplicated data is present at multiple locations, hence it leads to wastage of the storage space and the integrity of the database is destroyed.

**What are the various types of relationships in Database? Define them.**

**Answer: There are 3 types of relationships in Database:**

* **One-to-one:** One table has a relationship with another table having the similar kind of column. Each primary key relates to only one or no record in the related table.
* **One-to-many:** One table has a relationship with another table that has primary and foreign key relations. The primary key table contains only one record that relates to none, one or many records in the related table.
* **Many-to-many:** Each record in both the tables can relate to many numbers of records in another table.

**Explain Normalization and De-Normalization.**

**Answer:**

**Normalization** is the process of removing redundant data from the database by splitting the table in a well-defined manner in order to maintain data integrity. This process saves much of the storage space.

**De-normalization** is the process of adding up redundant data on the table in order to speed up the complex queries and thus achieve better performance.

**What are the different types of Normalization?**

**Answer: Different types of Normalization are:**

* **First Normal Form (1NF):** A relation is said to be in 1NF only when all the entities of the table contain unique or atomic values.
* **Second Normal Form (2NF):**A relation is said to be in 2NF only if it is in 1NF and all the non-key attribute of the table is fully dependent on the primary key.
* **Third Normal Form (3NF):**A relation is said to be in 3NF only if it is in 2NF and every non-key attribute of the table is not transitively dependent on the primary key.

**What is BCNF?**

**Answer:**BCNF is the Boyce Code Normal form. It is the higher version of 3Nf which does not have any multiple overlapping candidate keys.

**Q #10) What is SQL?**

**Answer:**Structured Query language, SQL is an ANSI(American National Standard Institute) standard programming language that is designed specifically for storing and managing the data in the relational database management system (RDBMS) using all kinds of data operations.

**How many SQL statements are used? Define them.**

**Answer:**SQL statements are basically divided into three categories, DDL, DML, and DCL.

**They can be defined as:**

* **Data Definition Language (DDL)**commands are used to define the structure that holds the data. These commands are auto-committed i.e. changes done by the DDL commands on the database are saved permanently.
* **Data Manipulation Language (DML)** commands are used to manipulate the data of the database. These commands are not auto-committed and can be rolled back.
* **Data Control Language (DCL)**commands are used to control the visibility of the data in the database like revoke access permission for using data in the database.

**Enlist some commands of DDL, DML, and DCL.**

**Answer: Data Definition Language (DDL) commands:**

* CREATE to create a new table or database.
* ALTER for alteration.
* TRUNCATE to delete data from the table.
* DROP to drop a table.
* RENAME to rename a table.

**Data Manipulation Language (DML) commands:**

* INSERT to insert a new row.
* UPDATE to update an existing row.
* DELETE to delete a row.
* MERGE for merging two rows or two tables.

**Data Control Language (DCL) commands:**

* COMMIT to permanently save.
* ROLLBACK to undo the change.
* SAVEPOINT to save temporarily.

**Define DML Compiler.**

**Answer:**DML compiler translates DML statements in a query language into a low-level instruction and the generated instruction can be understood by Query Evaluation Engine.

**What is DDL interpreter?**

**Answer:**DDL Interpreter interprets the DDL statements and records the generated statements in the table containing metadata.

**Enlist the advantages of SQL.**

**Answer: Advantages of SQL are:**

* Simple SQL queries can be used to retrieve a large amount of data from the database very quickly and efficiently.
* SQL is easy to learn and almost every DBMS supports SQL.
* It is easier to manage the database using SQL as no large amount of coding is required.

**Explain the terms ‘Record’, ‘Field’ and ‘Table’ in terms of database.**

**Answer:**

**Record:** Record is a collection of values or fields of a specific entity. **For Example,** An employee, Salary account, etc.

**Field:** A field refers to an area within a record that is reserved for specific data. **For Example,** Employee ID.

**Table:** Table is the collection of records of specific types. **For Example,** the Employee table is a collection of records related to all the employees.

**What do you understand by Data Independence? What are its two types?**

**Answer:**Data Independence refers to the ability to modify the schema definition in one level in such a way that it does not affect the schema definition in the next higher level.

**The 2 types of Data Independence are:**

* **Physical Data Independence**: It modifies the schema at the physical level without affecting the schema at the conceptual level.
* **Logical Data Independence:**It modifies the schema at the conceptual level without affecting or causing changes in the schema at the view level.

**Define the relationship between ‘View’ and ‘Data Independence’.**

**Answer:**View is a virtual table that does not have its data on its own rather the data is defined from one or more underlying base tables.

Views account for logical data independence as the growth and restructuring of base tables are not reflected in views.

**What are the advantages and disadvantages of views in the database?**

**Answer: Advantages of Views:**

* As there is no physical location where the data in the view is stored, it generates output without wasting resources.
* Data access is restricted as it does not allow commands like insertion, updation, and deletion.

**Disadvantages of Views:**

* The view becomes irrelevant if we drop a table related to that view.
* Much memory space is occupied when the view is created for large tables.

**Define Entity, Entity type, and Entity set.**

**Answer:**

**Entity** can be anything, be it a place, class or object which has an independent existence in the real world.

**Entity Type** represents a set of entities that have similar attributes.

**Entity Set** in the database represents a collection of entities having a particular entity type.

**Explain the terms ‘Attribute’ and ‘Relations’**

**Answer:**

**Attribute** is described as the properties or characteristics of an entity. **For Example**, Employee ID, Employee Name, Age, etc., can be attributes of the entity Employee.

**Relation** is a two-dimensional table containing a number of rows and columns where every row represents a record of the relation. Here, rows are also known as ‘Tuples’ and columns are known as ‘Attributes’.

**What do you understand by Join?**

**Answer:**Join is the process of deriving the relationship between different tables by combining columns from one or more tables having common values in each. When a table joins with itself, it is known as Self Join.

**Define Join types.**

**Answer:**Given below are the types of Join, which are explained with respect to the tables as an **Example.**

**employee table:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employee-table.jpg)

**employee\_info table:**

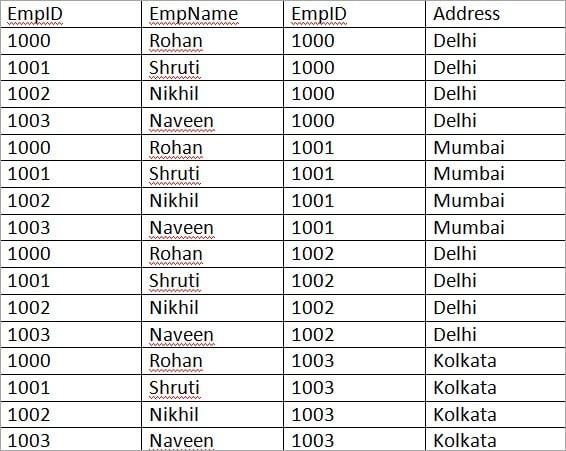
[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employee_info-table.jpg)

**a) Inner JOIN:** Inner JOIN is also known as a simple JOIN. This SQL query returns results from both the tables having a common value in rows.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee, employee\_info **WHERE** employee.EmpID = employee\_info.EmpID ; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2018/02/Inner-Join-Example.jpg)

**b) Natural JOIN:** This is a type of Inner JOIN that returns results from both the tables having the same data values in the columns of both the tables to be joined.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee NATURAL JOIN employee\_info; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Natural-JOIN.jpg)

**c) Cross JOIN:** Cross JOIN returns the result as all the records where each row from the first table is combined with each row of the second table.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee CROSS JOIN employee\_info; |

**Result:**

**Let us do some modification in the above tables to understand Right JOIN, Left JOIN, and Full JOIN.**

**employee table:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employee-table-new.jpg)

**employee\_info table:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employee_info-table-new.jpg)

**a) Right JOIN:** Right JOIN is also known as Right Outer JOIN. This returns all the rows as a result from the right table even if the JOIN condition does not match any records in the left table.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee RIGHT OUTER JOIN employee\_info **on** (employee.EmpID = employee\_info.EmpID); |

**Result:**

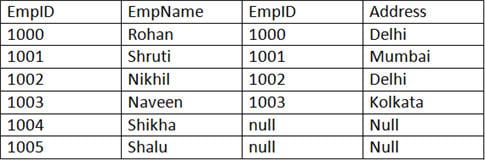
[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2018/02/Right-Join-Example.jpg)

**b) Left JOIN:** Left JOIN is also known as Left Outer JOIN. This returns all the rows as a result of the left table even if the JOIN condition does not match any records in the right table. This is exactly the opposite of Right JOIN.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee LEFT OUTER JOIN employee\_info **on** (employee.EmpID = employee\_info.EmpID); |

**Result:**

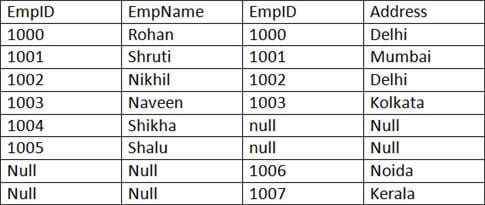
[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Left-JOIN.jpg)

**c) Outer/Full JOIN:** Full JOIN return results in combining the result of both the Left JOIN and Right JOIN.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** employee **FULL** OUTER JOIN employee\_info **on** (employee.EmpID = employee\_info.EmpID); |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Outer-Full-JOIN.jpg)

**What do you understand by ‘Atomicity’ and ‘Aggregation’?**

**Answer: Atomicity** is the condition where either all the actions of the transaction are performed or none. This means, when there is an incomplete transaction, the database management system itself will undo the effects done by the incomplete transaction.

**Aggregation** is the concept of expressing the relationship with the collection of entities and their relationships.

**Define Phantom deadlock.**

**Answer:**Phantom deadlock detection is the condition where the deadlock does not actually exist but due to a delay in propagating local information, deadlock detection algorithms identify the deadlocks.

**Define checkpoint.**

**Answer:**Checkpoint declares a point before which all the logs are stored permanently in the storage disk and is the inconsistent state. In the case of crashes, the amount of work and time is saved as the system can restart from the checkpoint.

**What is Database partitioning?**

**Answer:**Database partitioning is the process of partitioning tables, indexes into smaller pieces in order to manage and access the data at a finer level.

This process of partitioning reduces the cost of storing a large amount of data as well as enhances the performance and manageability.

**Explain the importance of Database partitioning.**

**Answer: The importance of Database partitioning are:**

* Improves query performance and manageability.
* Simplifies common administration tasks.
* Acts as a key tool for building systems with extremely high availability requirements.
* Allows accessing a large part of a single partition.

**Explain the Data Dictionary.**

**Answer:**Data dictionary is a set of information describing the content and structure of the tables and database objects. The job of the information stored in the data dictionary is to control, manipulate and access the relationship between database elements.

**Explain the Primary Key and Composite Key.**

**Answer: Primary Key** is that column of the table whose every row data is uniquely identified. Every row in the table must have a primary key and no two rows can have the same primary key. Primary key value can never be null nor can it be modified or updated.

**Composite Key**is a form of the candidate key where a set of columns will uniquely identify every row in the table.

**What do you understand by the Unique key?**

**Answer:**A Unique key is the same as the primary key whose every row data is uniquely identified with a difference of null value i.e. Unique key allows one value as a NULL value.

**What do you understand by Database Triggers?**

**Answer:**A set of commands that automatically get executed when an event like Before Insert, After Insert, On Update, On Delete of row occurs in a table is called as Database trigger.

**Define Stored procedures.**

**Answer:**A Stored procedure is a collection of pre-compiled SQL Queries, which when executed denotes a program taking input, process and gives the output.

**What do you understand by B-Trees?**

**Answer:**B-Tree represents the data structure in the form of a tree for external memory that reads and writes large blocks of data. It is commonly used in databases and file systems where all the insertions, deletions, sorting, etc., are done in logarithmic time.

**Name the different data models that are available for database systems.**

**Answer: Different data models are:**

* Relational model
* Network model
* Hierarchical model

**Differentiate between ‘DELETE’, ‘TRUNCATE’ and ‘DROP’ commands.**

**Answer:**After the execution of **‘DELETE’** operation, COMMIT and ROLLBACK statements can be performed to retrieve the lost data.

After the execution of**‘TRUNCATE’** operation, COMMIT, and ROLLBACK statements cannot be performed to retrieve the lost data.

**‘DROP’** command is used to drop the table or key like the primary key/foreign key.

**Q #51) Based on the given table, solve the following queries.**

**Employee table**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Employee-table-1.jpg)

**a)** Write the SELECT command to display the details of the employee with empid as 1004.

**SQL Query:**

|  |
| --- |
| **SELECT** empId, empName, Age, Address **from** Employee **WHERE** empId = 1004; |

**Result:**

[SELECT command](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/SELECT-command.jpg)

**b)** Write the SELECT command to display all the records of table Employees.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** Employee; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/display-all-records.jpg)

**c)** Write the SELECT command to display all the records of the employee whose name starts with the character ‘R’.

**SQL Query:**

|  |
| --- |
| **SELECT** \* **from** Employee **WHERE** empName LIKE ‘R%’; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/name-starts-with-character-R.jpg)

**d)** Write a SELECT command to display id, age and name of the employees with their age in both ascending and descending order.

**SQL Query:**

|  |
| --- |
| **SELECT** empId, empName, Age **from** Employee  **ORDER** **BY** Age; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employees-with-their-age-in-ascending.jpg)

|  |
| --- |
| **SELECT** empId, empName, Age **from** Employee  **ORDER** **BY** Age **Desc**; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/employees-with-their-age-in-descending.jpg)

**e)** Write the SELECT command to calculate the total amount of salary on each employee from the below Emp table.

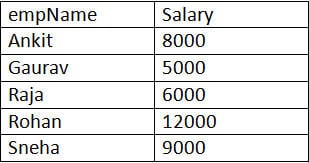
**Emp table:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Emp-table-1-e1492776699935.jpg)

**SQL Query:**

|  |
| --- |
| **SELECT** empName, SUM(Salary) **from** Emp **GROUP** **BY** empName; |

**Result:**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2017/04/Result.jpg)