**ASSIGNMENT**

**Submitted By,**

**Sajo Sam**

**S2 Regular MCA B**

**Roll No. 32**

* **What is 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)](https://www.oracle.com/in/database/what-is-database/#WhatIsDBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

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.

* **What is Structured Query Language (SQL)?**

SQL is a programming language used by nearly all [relational databases](https://www.oracle.com/in/database/what-is-database/#relational) to query, manipulate, and define data, and to provide access control. SQL was first developed at IBM in the 1970s with Oracle as a major contributor, which led to implementation of the SQL ANSI standard, SQL has spurred many extensions from companies such as IBM, Oracle, and Microsoft. Although SQL is still widely used today, new programming languages are beginning to appear.

* **Types of Database Objects.**

There are four different types of database objects that help users to compile, enter, store, and analyze data in various formats:

* Tables.
* Queries.
* Forms.
* Reports.
* **Types of Database Users.**
* Database Administrator (DBA).
* End-User.
* System Analyst.
* Application Programmer.
* Database Designer.
* **Types of Database.**

There are many different types of databases. The best database for a specific organization depends on how the organization intends to use the data.

1. **Relational Database.**
2. **What is relational database?**

The relational database refers to the collection of the relation between the values in the table. And it is the type of database that stores its data used in the form of a table where every row is used to contain data of every single user. The RDBMS is one of the widely used and accepted models that most developers prefer. Once the data is stored, it can be received whenever the developer needs to use the data.

1. **Advantages.**

* Secure Option.
* Simple to Use.
* Fast Performance.
* Present Data Accurately.
* Accessible Anytime.
* It Can Have Multi-User

1. **Disadvantages.**

* Increase Expenditure.
* Some Information Can Be Lost.
* Required To Use Of Physical Storage Devices.
* Performance Issue.
* It Can Be Complex To Understand.

1. **Examples.**

* MySQL.
* PostgreSQL.
* Oracle Database.
* MariaDB.

1. **NoSQL Database**
2. **What is NoSQL?**

These are used for large sets of distributed data. There are some big data performance issues which are effectively handled by relational databases, such kind of issues are easily managed by NoSQL databases. There are very efficient in analyzing large size unstructured data that may be stored at multiple virtual servers of the cloud.

1. **Advantages.**

* Handle Large Volumes of Data at High Speed with a Scale-Out Architecture.
* Store Unstructured, Semi Structured, or Structured Data.
* Enable Easy Updates to Schema and Fields.
* Developer-Friendly.
* Take Full Advantage of the Cloud to Deliver Zero Downtime

1. **Disadvantages.**

* Lack of Standardization.
* Backup of Database.
* Consistency.

1. **Examples.**

* MongoDB.
* Apache Cassandra.
* Apache HBase.
* Neo4j.

1. **Centralized Database.**
2. **What is Centralized Database?**

The information(data) is stored at a centralized location and the users from different locations can access this data. This type of database contains application procedures that help the users to access the data even from a remote location.

Various kinds of authentication procedures are applied for the verification and validation of end users, likewise, a registration number is provided by the application procedures which keeps a track and record of data usage. The local area office handles this thing.

1. **Advantages.**

* It allows for Working on Cross-Functional Projects.
* It is Easier to Share Ideas Across Analysts.
* Analysts can be Assigned to Specific Problems or Projects Centrally.
* Higher Levels of Security can be Obtained.
* Higher Levels of Dependability are Present Within the System.
* It Reduce Conflict.
* Organizations can Act with Greater Speed.
* It Helps an Organization Stay Close to a Focused Vision.

1. **Disadvantages.**

* It can Become Unresponsive to the Needs of the Business.
* There are Lower Levels of Location-Based Adaptability.
* It can Have a Negative Impact on Local Morale.
* Succession Planning can be Limited with a Centralized Database.
* It Reduces the Amount of Legitimate Feedback Received.
* It may Increase Costs**.**
* There is a Risk of Loss.

1. **Distributed Database.**
2. **What is Distributed Database?**

Just opposite of the centralized database concept, the distributed database has contributions from the common database as well as the information captured by local computers also. The data is not at one place and is distributed at various sites of an organization. These sites are connected to each other with the help of communication links which helps them to access the distributed data easily.

There are two kinds of distributed database, viz. homogenous and heterogeneous. The databases which have same underlying hardware and run over same operating systems and application procedures are known as homogeneous DDB, for eg. All physical locations in a DDB. Whereas, the operating systems, underlying hardware as well as application procedures can be different at various sites of a DDB which is known as heterogeneous DDB.

1. **Advantages.**

* Modular Development**.**
* More Reliable.
* Better Response.
* Lower Communication Cost.
* Improved Performance.

1. **Disadvantages.**

* Complexity.
* Cost.
* Security.
* Integrity Control More Difficult.
* Lack of Standards.
* Lack of Experience.
* Database Design More Complex.

1. **Cloud Database.**
2. **What is Cloud Database?**

Now a day, data are actually stored in a public cloud, a hybrid cloud or a private cloud, also known as a virtual environment. A cloud database is an automated or built-in database for such a virtualized environment. A cloud service offers various advantages, including the ability to pay per user storage capacity and bandwidth and provides scalability on request, as well as high availability. In addition, a cloud platform allows companies to support enterprise applications in the delivery of [software as a service](https://www.educba.com/what-is-software-as-a-service-saas/).

1. **Advantages.**

* Scalability.
* Reduced Administrative Burden.
* Improved Security.
* Cost Savings.
* Ease of Use.

1. **Disadvantages.**

* Inflexibility.
* Risk of Data Confidentiality.
* Depends on Internet Connection.
* Vulnerable in the Event of an Attack.
* Data Mobility.

1. **Examples.**

* Aws => Amazon RDS, Amazon SimpleDB, Amazon DynamoDB.
* Oracle database.
* Microsoft Azure.
* Google Cloud Platform.
* IBM DB2.

1. **Object Oriented Database.**
2. **What is Object Oriented Database?**

An object-driven database is an object-driven and relational database collection. There are different items, such as java, C++, that can be saved in a relational database using object-oriented programming languages, but object-oriented databases are suitable for these components. An object-oriented database will be organized instead of actions around objects and data instead of logic. In contrast to an alphanumeric value, for example, a multimedia record in a relational database can be a definable data object.

1. **Advantages.**

* Complex data sets can be saved and retrieved quickly and easily.
* Object IDs are assigned automatically.
* Works well with object oriented programming languages.

1. **Disadvantages.**

* Object databases are not widely adopted.
* In some situations, the high complexity can cause performance problems.

1. **Examples.**

* GemStone/S.
* ObjectDB.
* ObjectDatabase++.
* Objectivity/DB.
* ObjectStore.
* Versant.

1. **Graph Database.**
2. **What is Graph Database?**

Graph databases focus on the relationships between data along with the data itself. The database uses structures of connected graphs of nodes and edges for users to run queries. Companies may choose graph databases if they have datasets with complex relationships.

1. **Advantage.**

* Object-Oriented Thinking.
* Performance.
* Better Problem-Solving.
* Flexible Online Schema Environment.
* Make Powerful Recursive Path Query Easily Accessible.

1. **Disadvantages.**

* Graph are inappropriate for transactional-based systems.
* The user-base is small, making it hard to find support when running into a problem.
* There is no standardized query language.

1. **Examples.**

* JanusGraph.
* Neo4j.
* DGraph.
* DataStrx Enterprise Graph.

1. **Time Series Database.**
2. **What is Time Series Database?**

A time series database is a database optimized for time-stamped, or time series, data. Examples of this type of data include network data, sensor data, and application performance monitoring data. All of those Internet of Things sensors that are getting attached to everything put out a constant stream of time series data.

1. **Advantages.**

* More Accurate and Meaningful Time Series Measurement.
* Resource-Efficient Data Storage.
* Lightning-Fast data Queries.

1. **Disadvantages.**

* Problems with Generalization from a Single Study.
* Difficulty in Obtaining Appropriate Measures.

1. **Examples.**

* InfluxDB.
* Prometheus.
* RRDtool.
* TimeScale.
* Timestream.

1. **Network Database.**
2. **What is Network Database?**

It is the database that typically follows the network data model. Here, the representation of data is in the form of nodes connected via links between them. Unlike the hierarchical database, it allows each record to have multiple children and parent nodes to form a generalized graph structure.

1. **Advantages.**

* Conceptual Simplicity.
* Ease of Data Access.
* Data Independence.
* Database Standards.

1. **Disadvantages.**

* System Complexity.
* Operational Anomalies.
* Absence of Structural Independence.

1. **Examples.**

* TurboIMAGE.
* Inivac DMS-1100.
* IDMS.

1. **End-User Database.**
2. **What is End-User Database?**

The end user is usually not concerned about the transaction or operations done at various levels and is only aware of the product which may be a software or an application. Therefore, this is a shared database which is specifically designed for the end user, just like different levels’ managers. Summary of whole information is collected in this database.

1. **Advantages.**

* May Help to Reduce the Hidden Backlog.
* Faster Design.
* More Acceptable to Users.
* Reduces Communications Problems Between Users and IS.

1. **Disadvantages.**

* Duplication or Efforts and Waste Resources.
* Greatly Increased Costs.
* Loss of Control Over Data.