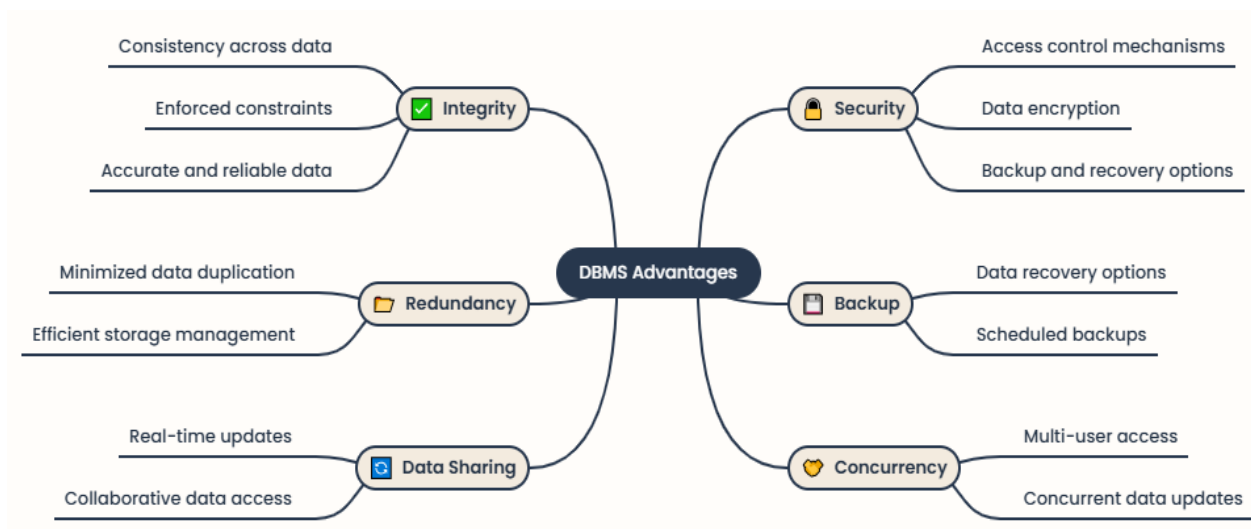**Database Course Documentation**

1. **Comparison Assignment: Flat File Systems vs. Relational Databases**

Point	Flat File Systems	Relational Databases
Structure	Data is stored on a single table (like a spreadsheet with rows and columns).	Data is stored in multiple tables linked by keys (e.g., customer ID, product ID).
Data Redundancy	High redundancy: the same data may appear in multiple places.	Low redundancy: shared data is stored once and referenced across tables.
Relationships	No relationships between records; each entry is independent.	Strong support for relationships via primary and foreign keys.
Example Usage	Storing contact lists, small inventories, or basic customer records.	Managing complex systems like orders, inventory, and customer data across multiple branches.
Drawbacks	- Prone to data duplication and inconsistency. - Becomes hard to manage as data grows.	- Requires more setup and technical knowledge. - More complex queries and maintenance.

Link of the source: https://www.ecisolutions.com/blog/building-supply/the-difference-between-flat-file-and-relational-data/

2. **DBMS Advantages – Mind Map Create a Mind Map that illustrates the advantages of using a Database Management System (DBMS).**

3. Roles in a Database System

Role	Summary of Responsibilities
System Analyst	Acts as the bridge between business needs and technical solutions. Gathers user requirements, analyzes workflows, and defines what the database system must support.
Database Designer	Designs the structure of the database (schema), including tables, fields, keys, and relationships, based on requirements provided by the system analyst.
Database Developer	Implements the database design by writing SQL scripts, stored procedures, functions, and triggers. Ensures efficient data access and manipulation.
Database Administrator (DBA)	Manages and maintains the database system. Responsibilities include installation, configuration, backup and recovery, security management, performance tuning, data integrity, and troubleshooting. DBAs also collaborate with developers and IT teams.
Application Developer	Builds applications (web, desktop, or mobile) that interact with the database. Ensures users can input and retrieve data through user-friendly interfaces.
BI (Business Intelligence) Developer	Extracts and transforms data from databases to build reports and dashboards. Helps businesses make data-driven decisions using tools like Power BI, Tableau, or SQL-based queries.

link of the source: https://www.baker.edu/about/get-to-know-us/blog/role-of-a-database-administrator/

4. **Types of Databases**

Aspect	Relational Databases (SQL)	Non-Relational Databases (NoSQL)
Structure	Data is stored in structured tables (rows & columns).	Data stored in flexible formats (documents, key-value, graphs, wide-columns).
Relationships	Uses primary and foreign keys to link data across tables.	Generally, no joins; data is often nested in one document.
Examples	MySQL, PostgreSQL, Oracle.	MongoDB (document), Cassandra (wide-column), Redis (key-value).
Advantages	ACID compliance, strong consistency, data accuracy, and normalization to avoid redundancy.	High scalability, flexible schema, better for unstructured/big data, and high performance.
Disadvantages	Rigid schema, complex changes, limited horizontal scaling.	Weaker consistency, not ideal for complex relationships, and no standard query language.
Use Cases	Banking systems, ERP, e-commerce (with complex transactions).	AI apps, IoT, big data analytics, and real-time applications.
When to use	are best when data is predictable, highly structured, and relationships between entities are crucial.	are ideal when data is unstructured, requires flexibility, or needs horizontal scaling in cloud environments.

Type	Description	Advantages	Disadvantages	Examples
Centralized	All data is stored and maintained at a single physical location.	- Easy to manage - Low cost - Data consistency	- Single point of failure - High data traffic - Slower multi-user access	Desktop/server CPU, mainframes
Distributed	Data is spread across multiple physical locations but works as one logical database.	- High availability - Faster access - Fault-tolerant - Scalable	- Complex maintenance - Costly - Harder to ensure data consistency	Apache Cassandra, Amazon SimpleDB
Cloud	Databases hosted on cloud platforms and accessed via internet.	- Scalable - Accessible anywhere - Managed services	- Dependency on internet - Ongoing costs - Potential vendor lock-in	Azure SQL, Amazon RDS, Google Spanner

Use case Examples

Use Case	Recommended Type
E-commerce platform with complex orders	Relational or Cloud Relational DB
Real-time IoT data collection	Non-Relational (Document/Key-Value)
A multi-branch business with fast access needs	Distributed Relational or Non-Relational
Business Intelligence dashboards	Cloud Database + BI tools (e.g., Power BI + Azure SQL)
AI-powered recommendation system	Document-based NoSQL (e.g., MongoDB)

Link of the source: 1. https://www.geeksforgeeks.org/dbms/difference-between-centralized-database-and-distributed-database/

2. https://www.mongodb.com/resources/compare/relational-vs-non-relational-databases

5.Cloud Storage and Databases

What is Cloud Storage and How Does It Support Databases?

Cloud storage is a model of data storage where digital data is stored on remote servers accessed via the internet. These servers are maintained by cloud service providers (e.g., AWS, Azure, Google Cloud), who handle the infrastructure, security, and availability.

Cloud storage **supports databases** by offering scalable, elastic, and on-demand storage for structured and unstructured data. It enables database services like Amazon RDS or Azure SQL to operate without requiring on-premise infrastructure, providing:

1. High availability and redundancy
2. Global accessibility
3. Easy integration with backup, disaster recovery, and analytics tools
4. Support for different storage types (e.g., block storage for transactional databases)

Advantages of Cloud-Based Databases

Advantage	Explanation
Scalability	Scale storage and compute resources up or down based on demand.
Cost-Efficiency	Pay-as-you-go pricing reduces upfront infrastructure costs.
High Availability	Data is stored across multiple zones for reliability and fault tolerance.
Fast Deployment	Provision databases in minutes without needing physical setup.
Global Access	Access databases from anywhere with an internet connection.
Security & Compliance	Providers offer encryption, access control, and compliance support (e.g., HIPAA, GDPR).
Integrated Ecosystem	Works well with data lakes, machine learning, analytics, and app development pipelines.

Examples of cloud-based databases:

- Amazon RDS – Fully managed relational database service
- Azure SQL – Cloud version of Microsoft SQL Server
- Google Cloud Spanner – Horizontally scalable relational database with global consistency

Disadvantages / Challenges of Cloud-Based Databases

Challenge	Details
Internet Dependency	Requires reliable network connectivity for access.
Ongoing Costs	Long-term usage may lead to high operational costs if not optimized.
Security Concerns	Sensitive data may require extra layers of encryption and access control.
Latency Issues	Geographic distance from data centers may affect performance.
Data Migration Complexity	Moving large datasets to the cloud can be time-consuming and expensive.
Vendor Lock-in	Moving from one cloud provider to another may be technically and financially difficult.6xtv

Link of the source: https://aws.amazon.com/what-is/cloud-storage/