LET'S START WITH DBMS :)

Query Optimization

Whenever we want to improve the efficiency of a query, make it execute faster and consume fewer resources we use query optimization techniques.

Optimizing SQL queries is essential for improving the performance of database applications

1.Use Indexes Efficiently: Index the columns frequently used in WHERE, JOIN, ORDER BY, and GROUP BY clauses to speed up data retrieval.

2.Select Only Necessary Columns : Avoid SELECT *, specify only the columns you need in your query. This reduces the amount of data transferred and processed.

LET'S START WITH DBMS :)

Query Optimization

3. Optimize JOIN Operations : Choose the Right JOIN Type and use indexed columns in Join conditions.

4. Partition Large Tables: For very large tables, consider partitioning them to improve query performance. Partitioning allows the DBMS to scan only relevant partitions, reducing I/O and improving response times.

5. Cache results : Cache the results of frequently executed queries to avoid redundant calculations.

LET'S START WITH DBMS :)

Physical Storage And File Organization

Physical Storage

Storage device : For storing the data, there are different types of storage options available.

- Primary Storage -> Main memory and cache (fastest and expensive and as soon as the system leads to a power cut or a crash, the data also get lost.)

- Secondary Storage -> Flash Memory and Magnetic Disk Storage (non-volatile)save and store data permanently.

- Tertiary Storage-> Optical disk and Magnetic tape (used for data backup and storing large data offline)

LET'S START WITH DBMS :)

Physical Storage And File Organization

Databases store data in files on disk. Each table or index may correspond to one or more files. Each file has sequence of records.
Accessing data from RAM is much faster than accessing data from a hard disk or SSD. This is because RAM is designed for high-speed read and write operations. Data retrieval in RAM typically involves fetching data in nanoseconds.

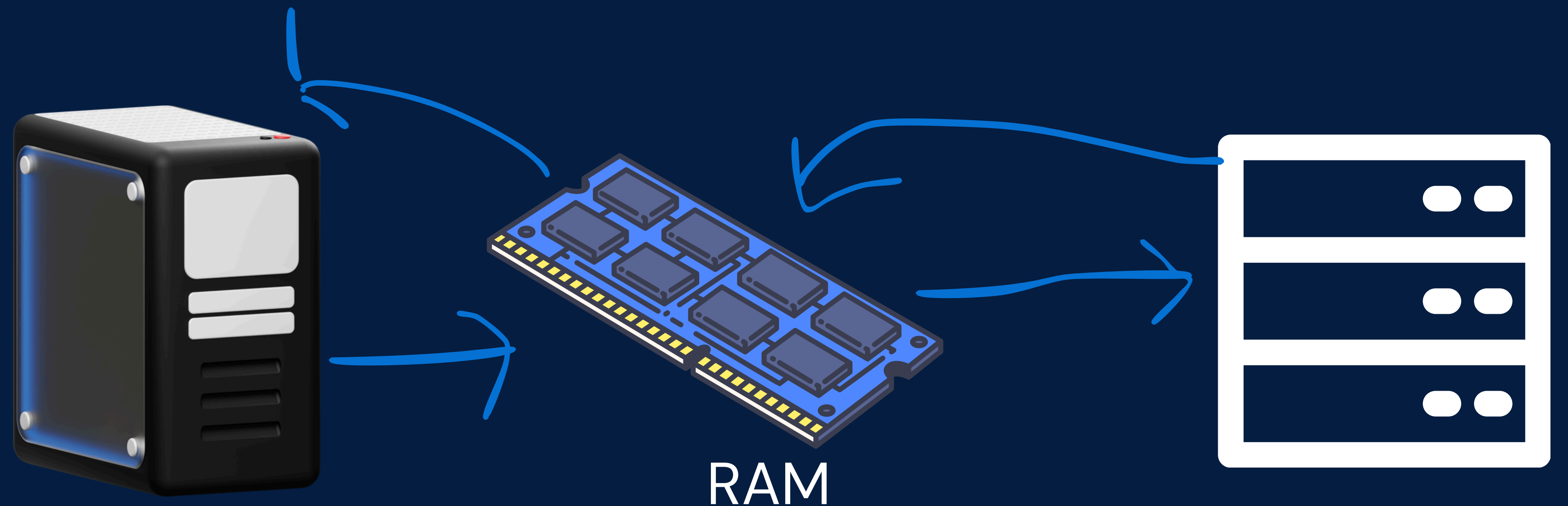
- **RAM** : Provides fast, volatile memory for quick data retrieval. Data is accessed via addressable locations and cache memory enhances speed.

- **Hard Disk:** Provides slower, non-volatile storage with mechanical components that impact access speed. Data retrieval involves moving the read/write head and waiting for disk rotation.

LET'S START WITH DBMS :)

Physical Storage And File Organization

When a program or application needs to access data, it first checks if the data is available in RAM. If the data is not in RAM, it must be loaded from a slower storage device like a hard disk or SSD. Index files in a Database Management System (DBMS) are stored on disk but are also managed in RAM to optimize performance



Query

CPU

RAM
(Volatile)

Hardisk
(non-volatile)

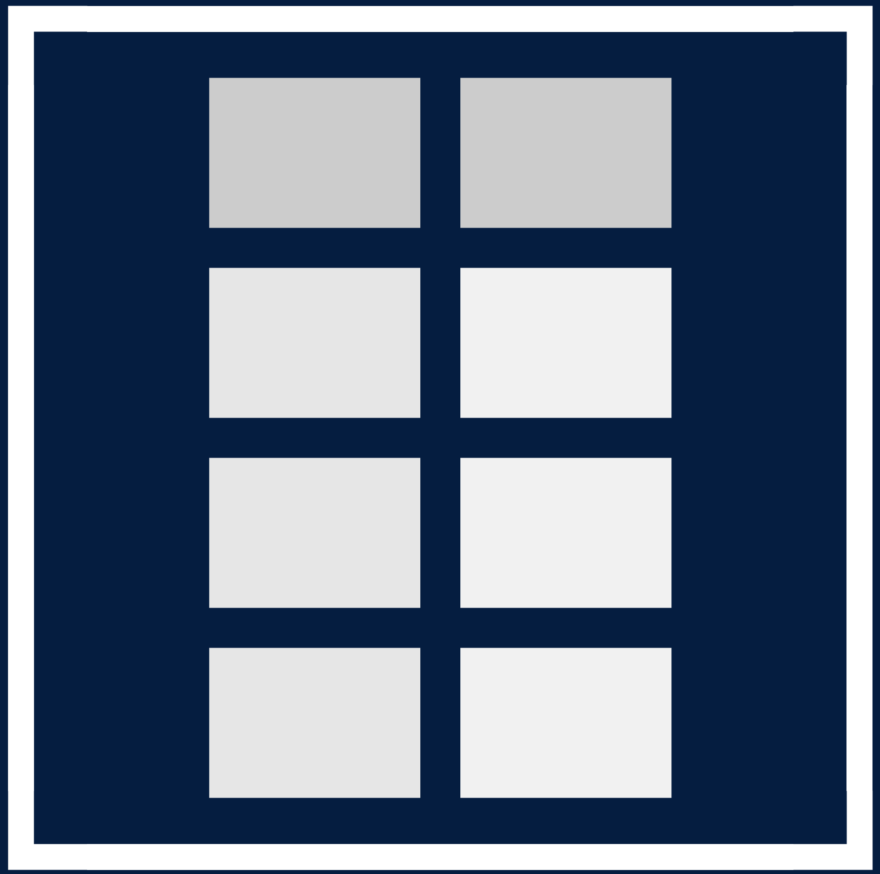
LET'S START WITH DBMS :)

Physical Storage And File Organization

How file is organized?

Rollno	Sname	SAge

DB



HARDISK

Files are allocated on the hard disk in contiguous blocks to reduce fragmentation.

No of records stored in each block= Size of block/size of record

Records can be stored in 2 ways-> sorted(seraching is fast) or unsorted(searching is slow, insertion is fast)