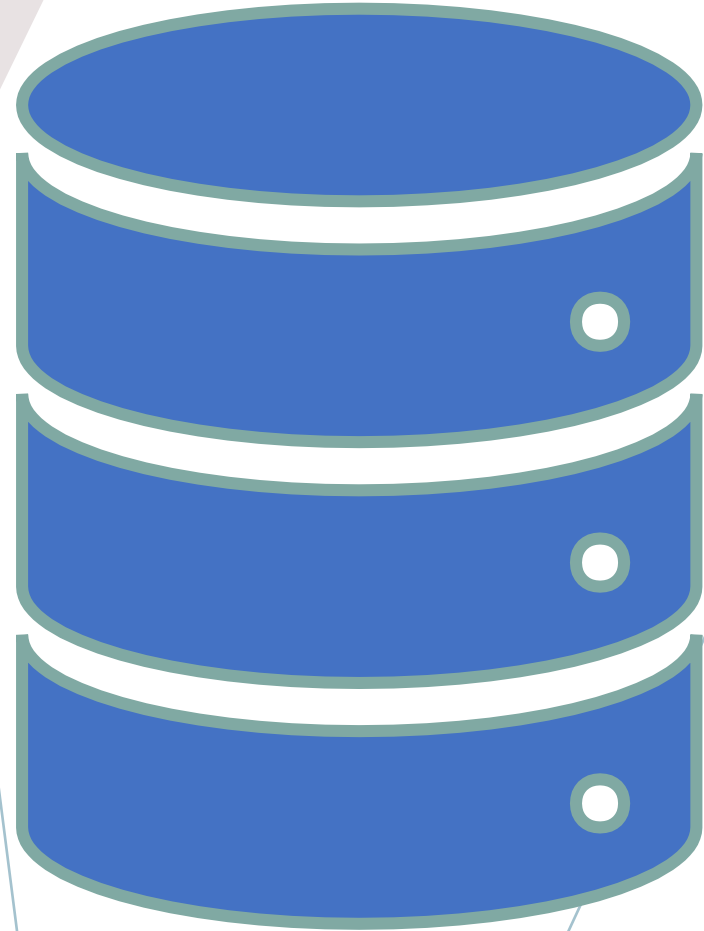


DATA BASE MANAGEMENT SYSTEMS

NOSQL DATABASE

CASE STUDY

GROUP-10



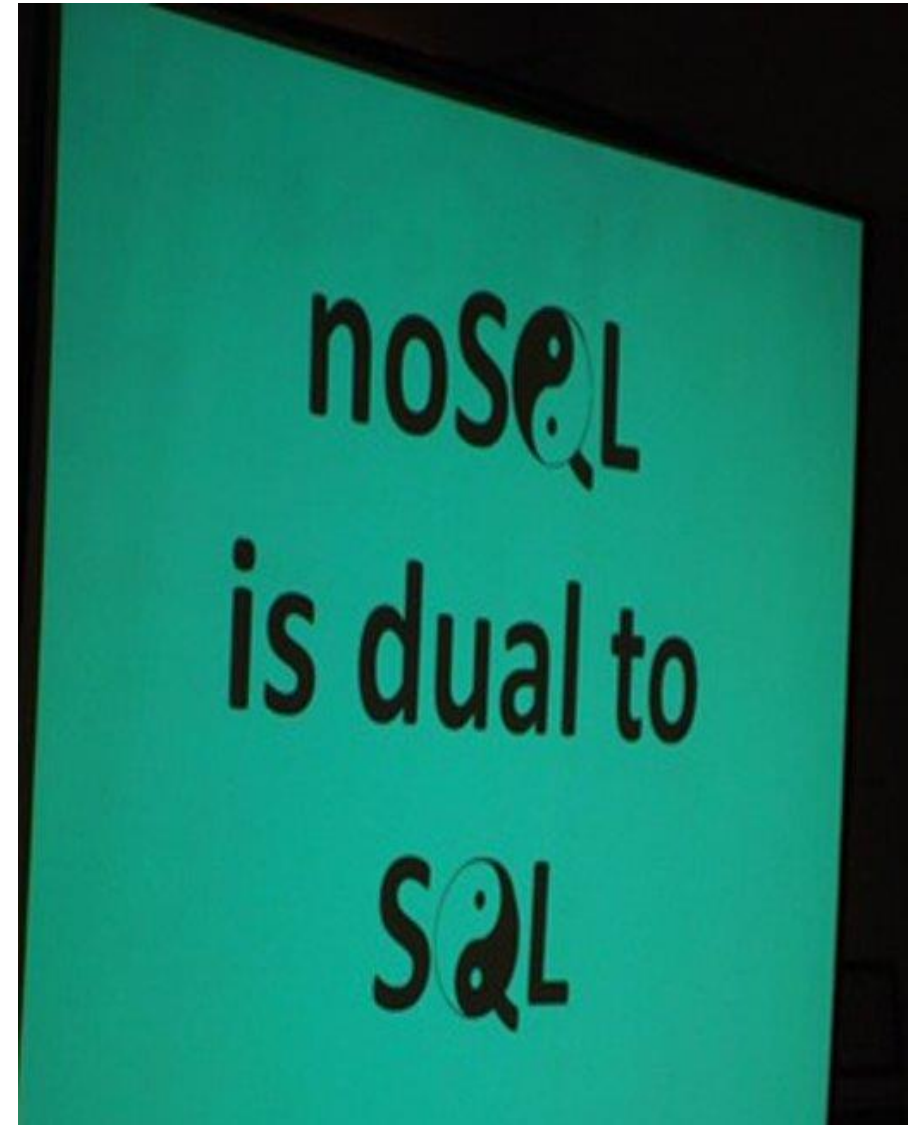
Team Members

1. CB.EN.U4CSE19453- ABHINAV
2. CB.EN.U4CSE19459 - SHANTHAN
3. CB.EN.U4CSE19405 - HEMANTH
4. CB.EN.U4CSE19449 - KOUSHIK



INTRODUCTION:

- It is a non-relational database
- NOSQL means Non-SQL
- Also called as "Not only SQL" as it supports SQL- like query languages.
- In NOSQL you can insert, update data on the fly.
- It deals with large amount of data and these are much faster and easy to scale in doing so.
- Used by Google, Facebook, Amazon...



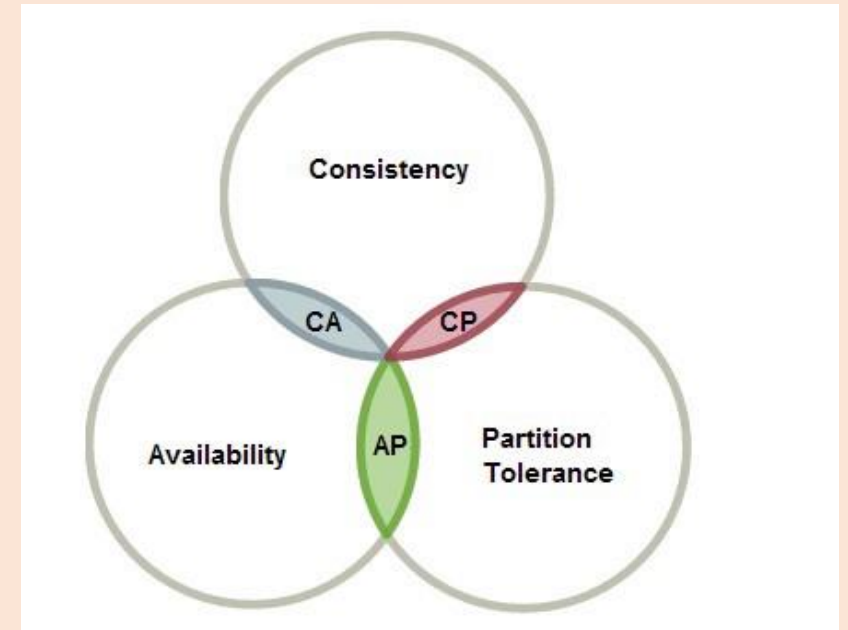
HISTORY

- At the time of 1970's the data storage was not organised and was also not safe, then came relational database that arranged data into rows and columns with specific key to each row.
- Relational databases use SQL query language.
- But in Mid-1990's when internet gained more popularity relational database could not keep up with the flow , Non-relational databases were used.
- in 1998 by Carlo Strozzi named his open source relational database as NOSQL as it did not use SQL.

CAP THEOREM

- The **CAP** Theorem is an important part of non-relational databases.
- It states that a distributed data store “cannot” simultaneously offer more than “two of three” guarantees.

- **C**onsistency
- **A**vailability
- **P**artition Tolerance



Relational Databases:

- A relational database typically stores information in tables containing specific pieces and types of data
- Relational databases use Structured Query Language (SQL).
- Mostly used in large enterprise scenarios.
- Relational databases work best when the data they contain doesn't change very often
- Examples: Microsoft SQL, PostgreSQL, MySQL



Non-relational Databases (NoSQL databases)

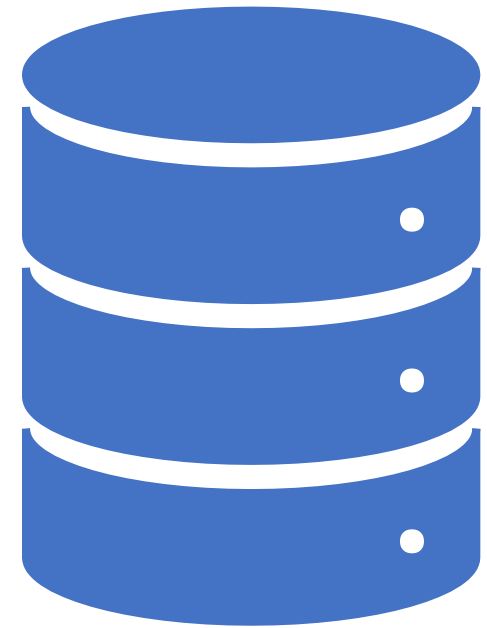
- A **non-relational database** is any database that does not use the tabular schema of rows and columns like in relational databases
- Non-relational databases are often used when large quantities of complex and diverse data need to be organized.
- Non-relational databases often perform faster
- Types: Document- oriented databases, Key-Value Stores ,Wide-Column Stores ,Graph Stores
- Example: MongoDB, Redis



Relational Databases Vs Non-Relational Databases

- The main difference between these is **how they store their information**.
- A **non-relational database** stores data in a non-tabular form, and tends to be more flexible than the traditional, SQL-based, relational database structures.

It does not follow the relational model provided by traditional relational database management systems.



Benefits of Introducing NoSQL Databases



Store heap of data that might not have a structure



Speed up the development process



Reap the benefits of cloud computing and storage



Elevate horizontal scalability

ACID (RDBMS) Vs BASE (NoSQL)

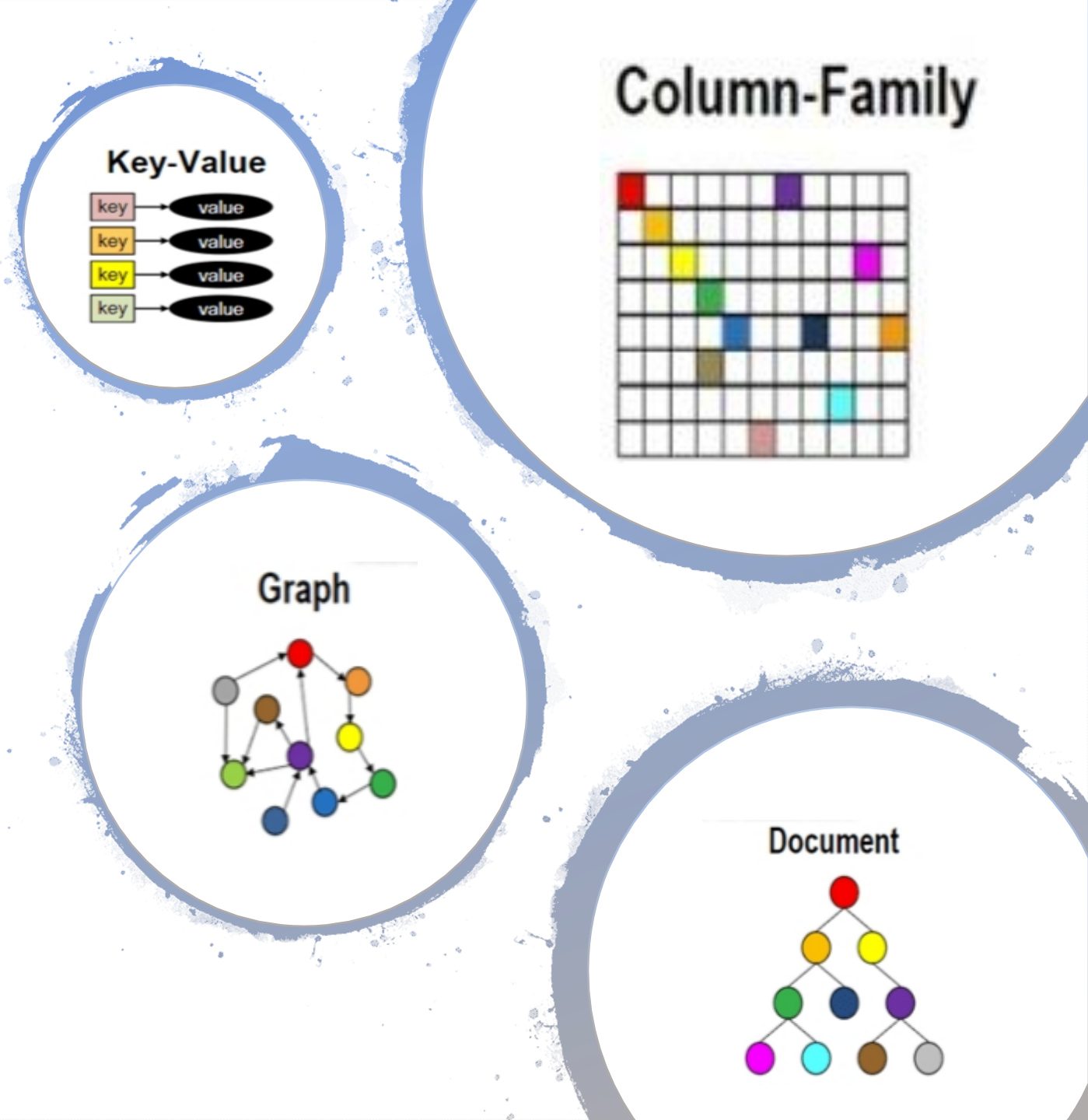
- **A**tomicity: All data and commands in a transaction succeed or all fail and roll back
- **C**onsistency: All committed data must be consistent with all data rules including constraints, triggers, cascades, atomicity, isolation and durability
- **I**solation: Other operations cannot access data that has been modified during a transaction that has not been completed
- **D**urability: Once a transaction is committed, data will survive system failures and can be recovered

- **B**asically **A**vailable: Guaranteed Availability
- **S**oft-state: State of the system may change, even without a query (because of a node updates)
- **E**ventually consistent: System will become consistent over time

ACID	BASE
Strong consistency	Weak consistency
Isolation	Last write wins
Transaction	Programmer Managed
Available / Consistent	Available / Partition Tolerant
Robust Database / Simpler Code	Simpler Database / Harder Code

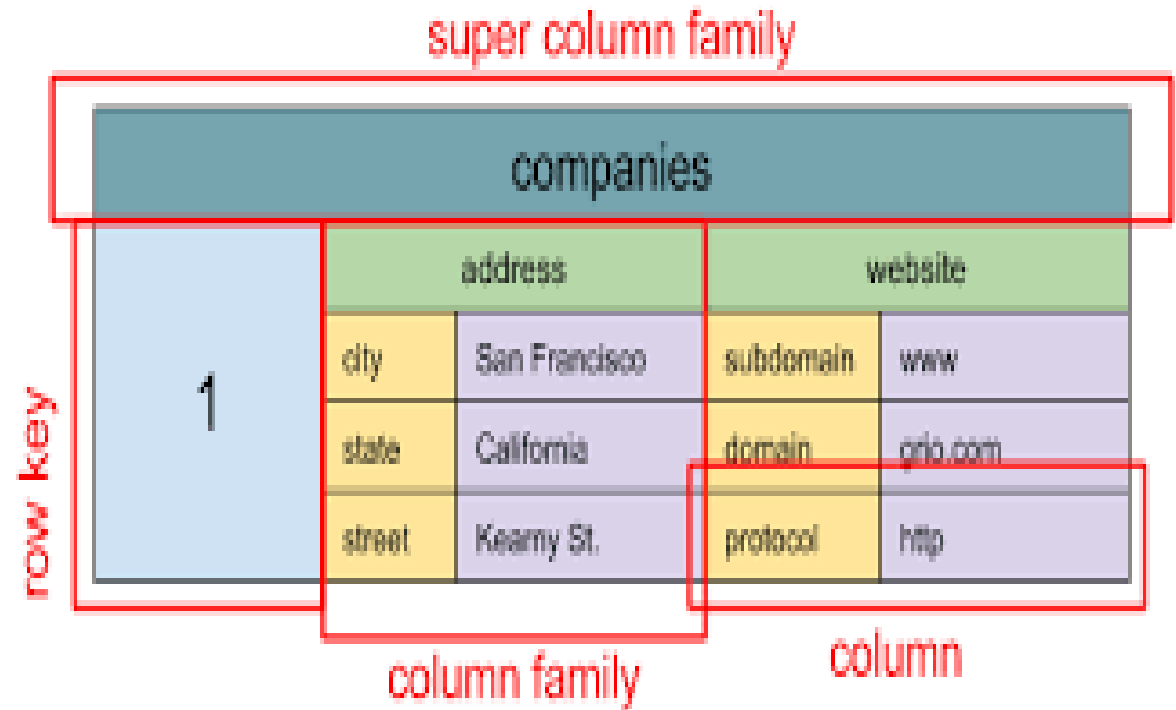
Types of NoSQL Databases:

- Key-value Pair Based
- Column-oriented Graph
- Graphs based
- Document-oriented



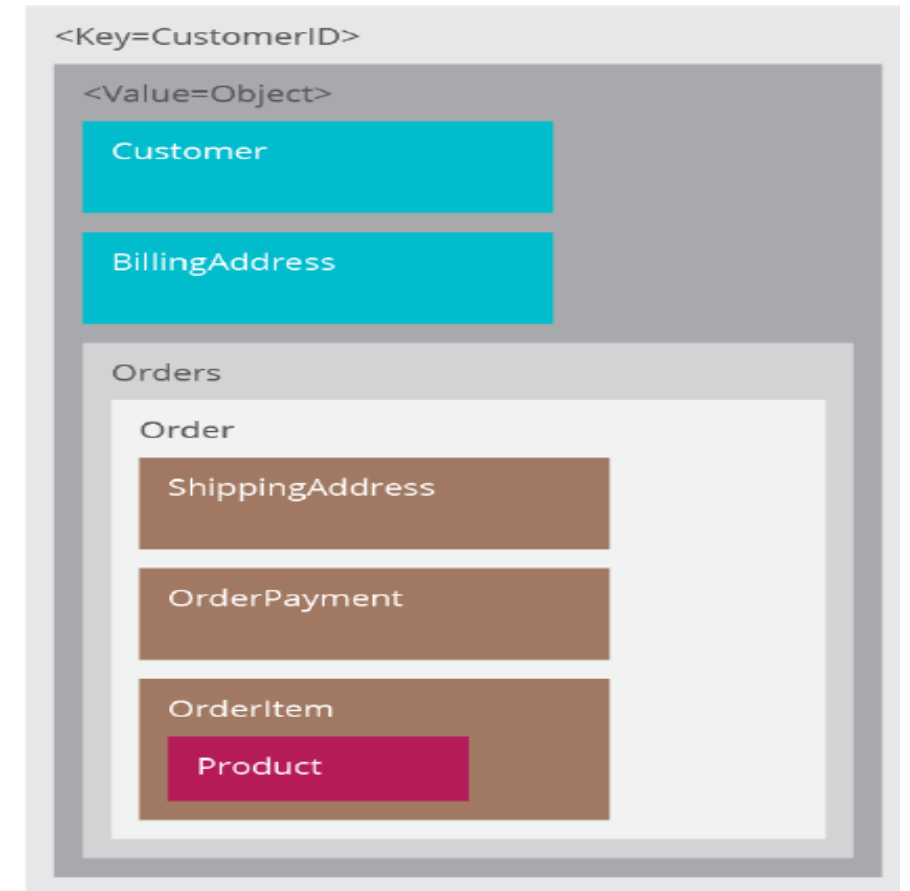
Column Based

- It store data as Column families contain in grows that have many columns associated with a row key. Each row can have different columns.
- Column families are groups of related data that is accessed together.
- Example: Cassandra, HBase, Hypertable, and Amazon DynamoDB.
- It is a tuple that contains a name, value,timestamp.
- We use it for content management systems, blogging platform.
- We would avoid it for systems that are linearly development, changing query patterns.



Key Value Pair Based

- Simplest Nosql databases
- The main idea is the use of a hash table
- Access data by strings called keys
- Data has no required format data may have any format.
- Data model: (key,value)
- Basic operations:
 - Insert(key,value)
 - Fetch(key)
 - Update(key)
 - Delete(key)



Graph Based



- Store entities and relationships between these entities as nodes and edges of a graph respectively. Entities have properties.
- Traversing the relationships is very fast as relationship between nodes is not calculated at query time but is actually tells as a relationship.
- Example: Neo4J, InfiniteGraph, OrientDB, FlockDB.
- It is well suited for connected data, such as social networks, spatial data, routing information for goods and supply.

Document Based

- The database stores and retrieves documents. It Stores documents in the value part of the key-value store.
- Self-describing, hierarchical tree data structures consisting of maps, collections, and scalar values.
- Example: LotusNotes, MongoDB, CouchDB, OrientDB, RavenDB.
- Documents can contain many different key- value pair or key- array-pair.
- We use it for content management systems, blogging platforms, web analytics, real-time analytics, e-commerce applications.
- We would avoid it for systems that need complex transactions spanning multiple operations or queries against varying aggregate structures.
- Indexes will be done via B- Trees.

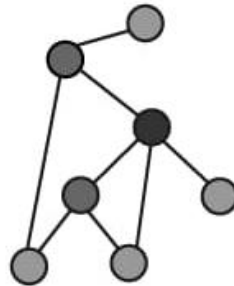


NoSQL DATABASE TYPES

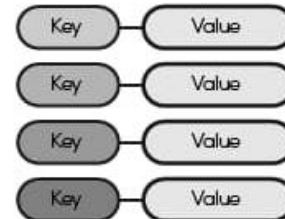
Document



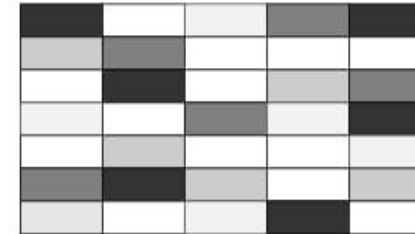
Graph



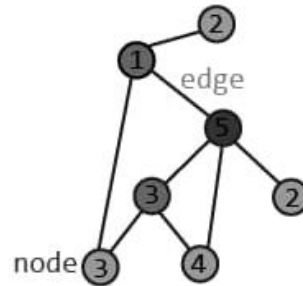
Key-Value



Wide-Column



```
{
  "user": {
    "id": "143",
    "name": "improgrammer",
    "city": "New York"
  }
}
```



1	Fruit	A Foo	B Baz	
2	City	E DC	D PLA	G FLD
3	State	A NZ	C CL	



Advantages of NoSQL



- Can be used as Primary or Analytic Data Source
- High scalability
- High availability
- No Single Point of Failure
- NoSQL databases don't need a dedicated high-performance server
- Support Key Developer Languages and Platforms
- Simple to implement than using RDBMS
- It can serve as the primary data source for online applications.
- It provides fast performance and horizontal scalability.
- Can handle structured, semi-structured, and unstructured data with equal effect

Disadvantages of NoSQL

- No standardization rules
- Limited query capabilities
- It does not offer any traditional database capabilities, like consistency when multiple transactions are performed simultaneously.
- Doesn't work as well with relational data
- Open source options are not so popular for enterprises.



THANK YOU

