Laksh Lukhi

Page No
Date

Q. 1

1. Explain Timeline of Fuzzy database to till today database with their names along with one line of description.

→ 1960's → Flat File systems → data stored in Plain Files. no relations.

1970 → Hierarchical DB (IMS by IBM) → Data in tree structure.

1970 → Network DB (COBASYL) → Data in grown structure.

1970's → Relations DB (RDBMS, E.F, codd) → Tables with Rows & columns.

1980's → object - Oriented DB → Stores object with attributes & methods.

1990's → Data wareh house → for Business analysis & decision making.

2000's → NOSQL DB → key-value, Document, column graph DB for big Data.

2010's → NoSQL DB → combines SQL Power with NOSQL scalability.

Today. cloud & Distributed DB → Examples. mono DB Atlas. Google Bigtable, AWS Dynamo DB.

2. Explain Advantages & Disadvantages of File data management System

* Advantages :-

1. Simple to Design & implement.
2. Fast for Small Data sets.

* Disadvantages :-

1. Data redundancy & intensistem.
2. Difficult to query and merge large.

3. Explain store Data persistently maintian consistency. Ensure Availabilty.

⇒ Store Dated Persisistenty → Data must be correct, valid and folial rules across DB.

Maintain Data consistency → Data must be correct even after system shutdown.

Ensure Data availbility → Data should be accessible whenever user / app need it.

4. Explain key-value pair databases features and list at least two Advantage and DisAdvantage for it.

☆ Features :-

- Data stored as key (unique identifier) → value (data)
- very fast lookup using key.
- Example : Redis, DynamoDB.

☆ Advantages :

- High Performance for simple queries.
- easy scalability for large Data.

Lukhi Laksh

Page No
Date

★ Disadvantages :
- NO computer querying of evolutionghiR.
- limited support for structured data.

5. Explain Document and collection in mongodb with an example.

⇒ * Document → A single JSON - like Record (key - value - pairs)
Ex :- { "name" : "khushi", "age" : 21, "{name : ADT}

* collection → group of Related documents (like, a table in RDBMS).
Ex :- student, collection containing many student documents.

6. Explain : Cluster, keyspale and Data-center in cassandra with an Example.

⇒ cluster → A group of nodes (computer) that store distributed data.
keyspale → The hight level container for tables. (like database in RDBMS)
Data-center → logical groping of felated notes inside a cluter.

Example :-
Clusker → "universtiy DB".
keyspale → "student Data"
Data-center → "Delhi_DC", "Mumbai_DC" (two locations storing replicas.")

Q. 2.

1. Explain how Homogeneous and Hetrogenous datahouse management works with example.

⇒ Homogemeous DBMS :-

1. all sites use the same DBMS software and data model.
2. Easier to manage, comsistent query proceliny.

Ex :- multiple oracle databuses connected in Distributed system.

Hetrogeneous DBMS :-

Defination :- Different sites may use different DBMS software & Data models.

working : middleware or wappers translate queries so systems can communicate.

Advantages : complexity, slower qury Processing.

Ex :- A company uses oracle (HR) Musql (sales), MongoDB (Inventory).

⇒ sites may use Different or sume In Both of them.

2. Replication and Fragmenation In Distributed Data base system.

⇒ * Replication Defination :- copying and maintaing data at multiple sites.

* types :-

1. Full Replication → whole DB copied every where. (high reliability But costly.)
2. Partical Replication → only selected tables Exagement copied (Balance Between cost and availability.)
3. No Replication → Each site store unique date only.

* Advantages :- High Availability fault tolexame fusted local queries.

* Disaduantage :- High storge cost uPxate conFlicts.

* Fragementation Definition :-

" Splitting a Database into smaller Pieces Stored at Different sites."

* Types :-

1. Hoxizontal Fragmenction → Rows divided.
   Ex :- (customer_USA . customer_India).

2. Vertical Fragmenatation → Columns divided,
   Ex :- (customer_Name,
   Customer_Add in one site,
   Customer_Balance,
   customer_credit in anoler.)

3. Hybrid Fragmentation → combination of Both.

* Advantage : Improves Performance By localizing queries.
* DisAdvantage :- complex query Processing across Fragments.

3. Explain graph Database with example.

⟹ * Definition :- " A Type of NoSQl database that store data as Nodes (entities) and edges (Relationships)

* working :- Relationships are stored discty, making traversal very fast.

* Disadvantages :- Not idel For tabular data. May not bandle very large-scale data like Qumm stores.

* ea :- N004j , Amazon Netune orient DB.

* use Cases :-

- Social network (Friends , Followers).
- Fraud detection (linking supicius allounts.)

- Recommendation engines (users, Products,).

Ex :- Note : { name : "Luksh" }
Note : { name : "Vyom" }
Edge : (Khushi) - [FRIEND] → (Vyom)

4. Explain could Database and API Database with example.

⇒ Definition :- "Databases hosted on clould Intrustructure, accessible old the Intrnet"

* working :- managed by cloud Provider, offering scalability, security, and Backups.

* Aduanteases :- on demond scalbity.
— Automatic Backup & recovery.
— Pay Pur we modal.

* DisAduantayes :- Internet dependeny
— vendor louk-in (diffult to swith Provider).

* Example :- Aws RDS
Google BigQury
mongoDB Atlas
Azure SQL Database.

* use cases :- E-commarce apps storing Produte & customer duta.
— Duta analytis For Big Duta.
— Mobile apps with Cloud Backed.

* API DataBase Definition :-
"A database accessible through APIs. Instead of direct SQL queries."

* working :- Applications Interact with the DB via RESI / graph QL APIS → CRUD, operations Done through API calls.

* Advantage :- • Easy Integration Between multiple apps.
• Real time access to data.

* Disdvantages :- • limited query Flexibility Compared to SQL.
• security Risk If APIS are not Properly managed.

* Examples : Fire Base Realtime Database, Airtable, API, Supabase.

* use cases :- ~~Mobile~~
• mobile apps syncing chat messages.
• IoT devices sending sensor data to DB via API.
• web apps Fetching live data.

5. Methods of MongoDB to create After and drop collection with example:

⇒ 1. Create collection :-

db.createCollection("Student")

↳ This create an empty collection named students.

- you can also Insert Diretoy.

db.student.InsertOne{name, "Khyshi"}

↳ mongodb auto crea collection.

2. After collation: Mangodb not support Direct After. But you can update collection.

Ex :- db.Student.update one ({ name: "luksh"} {age :22})

3. Drop collection :

ex : dp.student.drop ()

↳ This Permanenty deives the coliuction.

6. Different between Dowment oriented and columm orined Dytubuse.

⇒

Document oriented	Columm oriented
• Store deta as JsoN. like Jouments.	• Store data in columms group Inis Familes
• schema-less support nested stotude.	• seni - stoulture
• Best for compely Heriearchical datee query	• Best For Anullcut queries.

• Real time APPS. content management • Ex :- {name : "Laksh, age 22."}	• Big data Analytis. • columms :- 'name'

7. Data consisteny levels with. Read / write Examples

⇒ 1. strony consistenty
- Read always gets the letest write.
- Example -

write : Insert age = 21
Read Immediately → returns 21.

2. Eventual consistency
- Data becomes comsistnt after some time.
- Ex :- write age = 21 on one replica.

3. causal comstency
- keeps resext the cause. effxt ofer of operation.
- Ex :- user write a comment → mult see their own commt Before seeing reeplies.

4. Read Your own write consisency.
- A user always see their lettest writes.

Ex :- After Uploading Profile Picture → your own next read shows Uploted pic.

5. Explain CAP Theorm.

⇒ Definition :- CAP Theorem states that in a distributed Database system. it is Impossible to guranted all three Properies consisiency, Availbility and Parlion Tolerance.

→ three Properies of CAP :-

1. consistency :
→ All notes in the system see the sume data at the sume time.
→ If a user wries data to one Node. all other must return the update value Immdiatey
→ Ex :- In Banking If ₹ 300 is with drawn all servers should show the reduced Balance Instanty.

2. Availavility (A)
→ Every reuest reciver a response evenif some note are down.
→ The system should always be Up and seung clients
→ Ex :- Amazon com must show Produas even if one sever is offline.

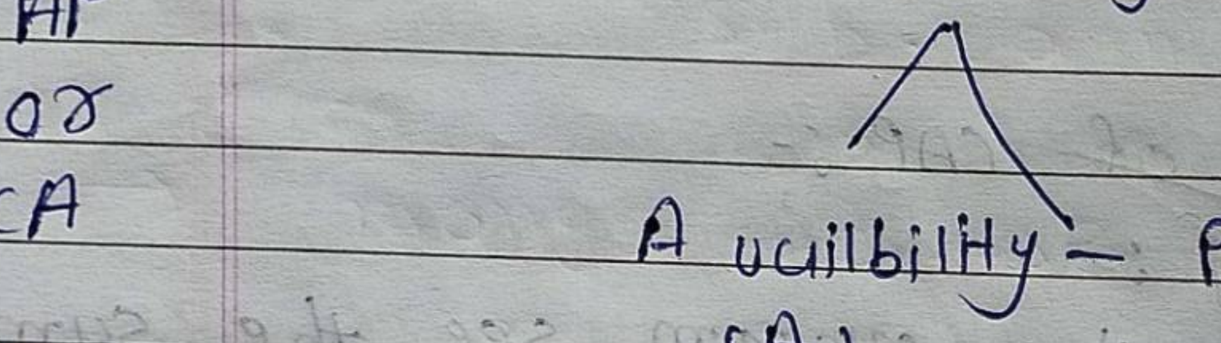
3. Partition Tolerance (P) :-

→ The system continue to function even if communi with Between some note fails.

→ Essential for Distribute systems where network failurs are (commun.)

→ ex :- Im a multi - region cloud DB. if a link Break Between data centers the system should still Run.

Diagram :- (for revision.) :-



9. Column family Data model of Cassandra.

→ Cassanr follows a column family Data model inspirs By google Big table.

→ It's is similar to a table in RDBMS But more flexible, It contain Row and Each Row have unique Key.

→ Each Row is a collection of column (key - value) group Into (Column)

→ Ex :- Student Record.

Row key : 101
Name → "Laksh"
subject → "RDBMS"
Row key : 102
Name → "RAM"
subject → "AI"

→ cassant alloss super coumm. sub collumm
→ Aduantage : High scalubility
→ cassandra use d Colymm - family model.
→ where datd store in Rows identify by d Row key and each Row contain multiple colymms.
→ colymms are grouPd in colymm Family.
→ unlike relation tables, not all Rows Need the sume colymm.
→ cassata have Nusted colymm → super colmm
→ This model enablen Fast writes, scalability and efficient Big detd handin:

10. Document data model of mongodb with embetted datd

→ mongodB is document oriented database that store data in JSON like BSON document
→ Document are ground in collection.
→ embeRad Data more :- Intecd of story releaed delta in multiple collion mongoDB allows embcoing document inere other document.

→ Ex :-

```
{
  "order-id": 1001
  "customer": "Lukh"
  "items": { "Product: laptop. }
  "Address": {"city": "A" Bad"
      "Pincode": 380001}}
```

* Benifit :-

Faster read Performance (no join)
Logical grouping of feuted data
Stitable for small related
sub-document

→ mongo DB used document Bused model whare date is stored in Bson document.

II. Process of Sharding in monoDB.

→ Sharding → Process of horizotaly.

→ Used in monoge DB for humaing very large trouble

* components :-

I. Shard → each Shard is a mongo DB instance holding a subsent of teta.

2. config servers →
Store metadata about data distribution.

3. query Routers (mongos) → Routers query to correct shard.

* Process —
→ Choose a shard key.
→ mongo DB divided documents into chunks Based on shard key.
→ Data is distributed across multiple shard for load Balanceing.

* Advantes :

→ scalability
→ High availability
→ Parallel query execution.
→ It's ensure scability and efficient handbling of large database.
→ query Rysult difult queries to the figh - Shard while config servers maintain metadetel.

12. Column and super - column in Data model.

→ Ex :- Smallest unit in cassantra, key - value Pair with times lamp.

Ex 1-

Super column : Address
City → "Surat"
Pincode → 380000

→ cassandra data model

- keyspace → container for data.
- column family → group of Rows
- Row → Identify Row key.
- column → Date storage unit.

- data store in Key space →
  column family →
  Rows → column.

- A column is a Key - value Pair

13. Cassandra's Architecture :

→ Cassandra is a Peer to Peer distributed database.

→ Main components :

1. Node :- Basic storage unit.
2. Cluster : Collection of Node.
3. Keyspace : Top level name space

* Feature of Architecture :-

Gossip Protocol → Note share info about each other.

Partitioners → Desire which note store data.

Snitch → Decide data Placement Based on topology.

* Advantage :-

- No singe point of failure
- liner scalability
- High Availability.

→ Node format in cluter
→ It use gossip Project

14. Read one write operation in cassandra.

→ write operation :-

1. Data is written to a commit log for darability.
2. written into mentable.
3. If not Found full → Flushed to sstable on disk.

→ Advantage :-

High write speed due to sequential write.

15. features of Neo4j + Difference Between RDBMS graph DB.

⇒ feature of neo4j :-

1. graph - Bused storage

2. Supports Cypher Query language
→ Easy graph Query

3. Highy suitble for social networks, frow detection.

* Differme :-

RDBMS → Tables
Rows
Joins

GraphDB → nodes
edeges
Direct felution ships
(No Joins)

→ RDBMS suits structured data, graph DB suits connected data.

→ features :- graph - Based storge
cypher Query Languge.
good for connocty data.

16. creating, Alltaing and Deleting Data base in Neo4j.

→ creak data Base

CREATE DATABASE Stymt

→ After database

ALTER DATABASE Stymt DB SET ACCESS Read only.

→ Delete Data Base.

DROP DataBase stymtDB;

→ only allows in Neo4j, enterprice enjoy not
→ not community.
→ model Building
→ Prediction
→ output (Analyse) correct be nor verify & Testing a or model