**Structured Datatypes**

Structured datatypes are data types that are used to group together multiple values of different data types into a single entity.

It refers to data that is organized and formatted in a specific way to make it easily readable and understandable by both humans and machines.

**ACID Transactions**

These are properties that guarantee reliable processing of database transaction.

A–**Atomicity**–For each statement in a transaction whether to read, write, update or delete data, it  is treated as a single unit.

C–**Consistency –**

I–**Isolation**– when multiple users are reading and writing from the same table all at once, isolation of their transactions ensures that the concurrent transactions don't interfere with or affect one another.

D–**Durability**–ensures that changes to your data made by successfully executed transactions will be saved, even in the event of system failure.

ACID Transactions: **Relational database** supports ACID transactions to ensure data consistency and integrity, they guarantee that the database operations are either fully completed or fully rolled back in the event of errors.

A relational database is a type of database management system that stores data in structured tables with rows and columns.

**OLTP—Online Transaction Processing**

It refers to class of systems applications that manage and support real-time transactions workloads.

They handle a large number of short, interactive and data-centric tasks for business operations.

OLTP uses ACID Transactions to ensure that data consistency and integrity

**Cloud-based data warehouse services for ACID Transactions**

AWS—RDS, Aurora

Azure—Azure SQL Database

Google Cloud—Cloud SQL, cloud Spanner

Agnostic—SQL Server, Oracle, DB2,MySQL,PostgreSQL

**Analytics**

Analytics in structured datatypes such as those in relational database involves using data analysis techniques to extract valuable insights and information from structured datasets

Structured datatypes are the cornerstone of analytics because they provide a well organized consistent, and accessible source of data for analysis

Analytics and columnar databases are related in the context of data processing and analysis particularly when it comes to efficiently handling and querying large datasets

**Types of analytics**

1. **Descriptive analytics-**Involves summarizing and describing data to gain an understanding of its characteristics
2. **Diagnostic analytics—**Aims to understand the causes of specific events or trends ,it involves investigating data anomalies, identifying correlations and performing root cause analysis.
3. **Predictive analytics—**It leverages historical data to build models that can make predictions about future events or trends
4. **Prescriptive analytics—**It suggests actions to optimize outcomes based on predictive models and business constraints

**OLAP—Online Analytical Processing**

This refers to a category of computer systems and tools that are designed for complex and multidimensional data analysis.

OLAP systems are used for querying and reporting large datasets .

OLAP is related to analytics datatypes and plays a crucial role in data analytics eg

* Data storage and Organisation
* Data exploration and Analysis
* Aggregation and Calculation
* Query Perfomance among others

**Cloud-based data warehouse services for Analytics**

AWS—Redshift

Azure—Azure Synapse

Google Cloud—BigQuery

Agnostic—Snowflake, ClickHouse, Druid, Pinot, Dtabricks

**Dictionary Data**

In the context of databases and semi-structured data, a "**dictionary**" typically refers to a data structure or a mechanism for storing and retrieving key-value pairs efficiently. This is often used to manage semi-structured data, such as JSON or XML data, which doesn't fit neatly into traditional relational database tables.

**Key-Value:**  In semi-structured data, information is often organized as key-value pairs, where each key corresponds to a specific attribute or field, and the associated value contains the actual data.

**Key-Value database**  
AWS – DynamoDB  
Azure - Cosmos Db  
Google Cloud - BigTable  
Cloud Agnostic – Redis,ScyllaDB,Ignite

**In-Memory:**  In-memory databases store and manage data primarily in the system's main memory (RAM), which is much faster to access than traditional disk-based storage. Dictionaries are frequently employed in these databases to index and retrieve data elements swiftly.

**Managed in-memory datastores**  
AWS – ElasticCache  
Azure – Azure cache for Redis  
Google Cloud – Memory-Store  
Cloud Agnostic – Redis, Memcached,Hazelcast,Ignite

**Dfn: of 2-d key-vlaue**

A key-value data model, also known as a key-value store, is a type of non-relational database. In this model, data is organized as a collection of key-value pairs.

Each key is associated with a single value, forming an associative array. Keys serve as unique identifiers for values, and these values can represent various types of entities. Key-value databases do not have a predefined schema or structure, and they are well-suited for storing unstructured or semi-structured data. Each key-value pair is typically stored as a separate record, allowing for flexible and efficient data storage and retrieval.

**When to use a key-value db**

when user-session attributes are real-time and random access like online gaming and financial

when the application is developed on queries based on keys

When caching mechanism is for repeatedly accessing data

**Merits.**

Easy to use

Built in redundancy (repetitiveness) makes this db more reliable

Fast response due to simplicity

Scalability of db both vertically and horizontally

**Demerits**

cannot transport queries from one db to another

Databases are not refined meaning you cannot query db without a key

**features**

Built-in redundancy makes db more reliable

Un-complex interms of NoSQL data models

Querying language not present

For CRUD processes simple functions are used

**Examples**

Berkeley DB

Aerospike

Amazon DynamoDB

Couchbase

Riak

In semi-structured data, **the entities belonging to the same class may have different attributes even though they are grouped together**, and the attributes’ order is not important. This is different from structured data where entities can be grouped together to form relations.

**An Entity Relationship Diagram**

**An Entity Relationship Diagram** (ERD) is a type of diagram that lets you see how different entities (e.g. people, customers, or other objects) relate to each other in an application or a database

**Entity**: An entity is a thing that can have data stored about it.

**Attribute**: An attribute is a property of an entity or something that can be used to describe an entity

**Relationship**: A relationship in an ERD defines how two entities are related to each other

**Types of Relationships**

* One to one
* One to many
* Many to many

**graph database**  
AWS – Neptune  
Azure - Cosmos Db  
Google Cloud - JanusGraph + Big Table  
Cloud Agnostic – OrientDB, Neo4j, Giraph

**Nested objects**

**Nested objects** are a common form of semi-structured data. They are often used in JSON data and other similar data formats.

**Nested objects database**  
AWS – DocumentDB  
Azure - Cosmos Db  
Google Cloud - Firestone  
Cloud Agnostic – Mongo DB,Couchbase,Soir

**Nested text is further broken down into useCases;**

1. **Full-text search** is a technique used in databases to find relevant results from a search query

**Full-text search services**  
AWS – Open-search, cloud search  
Azure – Cognittive Search  
Google Cloud – search APIs on DataStores   
Cloud Agnostic – Elastic Search,Soir,Elassandra

1. **Rich text** refers to text that includes formatting information such as fonts, colors, sizes, styles (like bold or italic), and more.

**NB: Rich text and Unstructured Data do not have traditional database. Instead, they use object storage service**

**Rich text and Unstructured Data storage services**  
AWS – S3 (Simple Storage Service)  
Azure – Blob Storage  
Google Cloud – Cloud Storage   
Cloud Agnostic – HDFs,MiniO

**Unstructured Data** Unstructured data refers to information that does not have a predefined model or organization, making it difficult to store, process, and analyze using traditional relational databases or spreadsheets

Examples of unstructured data include emails, word processing documents, PDFs, social media posts, webpages, videos, audio files, presentations, web content, and many other kinds of business documents

The term “BLOB” stands for “Binary Large Object” and is used to describe a collection of binary data stored as a single entity .In the context of cloud storage services like Amazon S3, a blob can refer to any piece of data that is stored without being broken down into files or folders.