Research Assignment 1

Section A : Database Fundamentals

1. @NoSQL Databases
   (b) Relational Databases
   (c) Cloud Databases
   (d) Vector Database
   (e) Time-series databases
   (f) Object-oriented databases
   (g) Graph databases
   (h) Hierarchical databases
   (i) Network databases (Ali, 2024)

2. **Relational Database Management System (RDBMS)** - databases that store data in tables structured into rows and columns. Each row represents a unique record, and each column represents a specific attribute of that record. (Ali, 2024).

3.) **Primary key** is a unique identifier within its table. It enforces uniqueness within their table, ensuring each record is identifiable.

- **Foreign key** is a reference in one table to a primary key in another. It is used to establish and navigate relationships between tables. (TiDB, 2024).

4. Database normalization is a technique used to structure a relational database in a way that minimizes data redundancy and ensures data dependencies are logical (Connolly & Begg, 2015).

5. Database Schema - the logical structure or blueprint of how data is organized in a database. It defines how tables are arranged, what columns they contain, how those tables relate to each other, and what constraints or rules apply to the data. (Connolly & Begg, 205).

6. Structured data is data whose elements are addressable for effective analysis. It has been organized into a formatted repository that is typically a database. Whereas semi-structured data is information that does not reside in a relational database but that has some organizational properties that make it easier to analyze. Unstructured data is a data which is not organized in a predefined manner or does not have a predefined data model, thus it is not a good fit for a mainstream relational database. (GeeksforGeeks, 2025)

7. Fact tables are the core of a data warehouse, storing quantitative data for analysis. Their strength lies in their ability to hold numerical and additive informative that can be aggregated to answer key business questions. Whereas dimension tables provide the raw data for analysis, dimension tables offer the necessary context. These tables contain descriptive attributes that you can use to filter, group, and label the data stored in fact tables. (MacDonald, 2024).

8. A data model is an integrated collection of concepts for describing and manipulating data, relationships between data, and constraints on the data in an organization, (Connolly & Begg, 2015).
- Data Modeling is important because it clarifies business requirement, improves database design, reduces redundancy and inconsistency, enhances data integrity and accuracy, facilitates communication and supports query and system performance (Connolly & Begg, 2015).

9. A database is a collection of real-time information that has been stored in an organized way. WHEREAS a data warehouse is a central repository for storing large amounts of historical data. It uses

Online Analytical Processing (OLAP) to allow pre-aggregation and multidimensional data analysis. WHEREAS data lake is a repository for any kind of raw, unfiltered data. Unlike a data warehouse, which is hierarchical, a data lake tends to have a flat architecture, (Sage, 2025)

10. A data mart is a subset of a data warehouse, focused on a specific business function or department, while a data warehouse is a centralized repository designed to store and integrate data from across the entire organization for analysis and reporting
- A data mart is a smaller, department-specific repository that focuses on a single business function, such as sales or finance, (Lu, 2025).

Section B

11. A query language is a type of computer language used to retrieve, manipulate and manage data stored in databases. It allows users to ask queries about data and get specific results without manually searching through the database.

It is most widely used because of the following reasons:

1. Standardization
2. Ease of use
3. Powerful Data management.
4. Integration and Compatibility
5. Widespread Adoption.

12. Indexes in Databases:

Indexes are data structures that speed up data retrieval. They allow the database to find rows faster without scanning the entire table, improving query performance, though they can slightly slow down insert operations.

13. A transaction is a unit of work that include one or more database operations executed as a single process.

-ACID properties ensure reliability:

* Atomicity - All or nothing.
* Consistency - Data remains valid
* Isolation - Transactions don't interfere.

* Durability - Changes persist after completion.

14. Database engine is the core software component that stores, retrieves, and processes data. It impacts speed, concurrency, and reliability of operations.

15. View is a virtual table based on a SQL query.
- Stored Procedure is a predefined SQL code saved for reuse.
- Trigger is an automatic action executed in response to an event (e.g. insert or update).

16. ETL (Extract, Transform, Load): Data is transformed before loading into the target system.
- ELT (Extract, Load, Transform): Data is loaded first, then transformed inside the target database (common in modern data warehouses).

17. Batch Processing: Processes large volumes of data at scheduled intervals.
Stream Processing: Processes data in real time as it arrives.

18. A join is SQL, combines data from two or more tables using related columns.

- INNER JOIN → matches rows in both tables.

Example:

```
SELECT column(s)
FROM table1 AS A
INNER JOIN table2 AS B
ON A.common_column = B.common_column
```

- LEFT JOIN: All from left + matches from right.

```
SELECT A.teacher_id,
       teacher_name,
       COALESCE(subject_name, 'No Subject Assigned')
           AS Subject_name
FROM teachers AS A
LEFT JOIN subjects AS B
ON A.teacher_id = B.teacher_id
ORDER BY teacher_name ASC;
```

- RIGHT JOIN: All from right + matches from left.

```
SELECT A.user_id,
       name,
       COUNT(login_date) AS login_count
FROM users AS A
RIGHT JOIN logins AS B
ON A.user_id = B.user_id
GROUP BY A.user_id, name;
```

- FULL JOIN : All records from both sides.

```
SELECT A.project,
       name,
       COUNT(task_id) AS task_count
FROM projects AS A

SELECT COALESCE(A.cust_id, B.cust_id)
       AS cust_id,
       order_total,
       return_total,
  CASE WHEN return_total IS NOT NULL
       THEN 'Returned'
       ELSE 'No Return'
  END AS return_status
  FROM orders AS A
  FULL OUTER JOIN returns AS B
  ON A.cust_id = B.cust_id ;
```

~~-co~~

- CROSS JOIN : Cartesian Product (all combinations).

```
SELECT P.product_name,
       C.color
  FROM Products P
  CROSS JOIN Colors C;
```

19. Referential Integrity

- It ensures relationships between tables remain consistent e.g a foreign key in one table must match a primary key in another. It prevents invalid references and maintains data accuracy.

Section C: Data Management and Analytics Concepts.

21. Cloud vs On-Premise Databases:
* Cloud is hosted on remote servers, scalable, accessible anywhere, managed by providers.
* On-Premise is installed locally, full control but higher maintenance and infrastructure cost.

22. Data governance is a framework for managing data availability, usability, integrity, and security. It ensures data is consistent, compliant, and reliable for decision-making.

23. Data Integrity refers to the accuracy and consistency of data over its lifecycle. Maintained through constraints, validation rules, access controls, and regular audits.

24. Data Quality measures how accurate, complete, consistent, and timely data is. High-quality data ensures reliable analytics and better business decisions.

25. A Data Analyst collects, cleans and interprets data to uncover insights. Uses SQL, Excel and visualization tools to support decision-making

and report findings.

26. Responsibilities of a DBA:
- Install, configure, and maintain databases.
- Ensure backup and recovery
- Optimize performance.
- Manage user access and security.
- Monitor availability and capacity.

27. Steps in Designing a Data Pipeline:
(a) Extract data from sources.
(b) Transform (clean, format, validate)
(c) Load into storage or data warehouse.
(d) Monitor performance and errors.

28. Data growth and storage limits.
- Performance tuning.
- Backup and recovery.
- Security and access control.
- Maintaining data consistency across systems.

29. MySQL: Web applications, open-source projects.
PostgreSQL: Complex queries, enterprise apps
Oracle: Large enterprises, mission-critical systems.
SQL Server: Microsoft environments.
Snowflake: Cloud-based analytics and warehousing.

30. Data storage formats in analytics:

- CSV : simple, readable, but large in size.
- JSON : Semi-structured, great for APIs.
- Parquet : Columnar, efficient for big data analytics.
- Avro : compact binary format for streaming data.