Modise LE
Research Assignment : 1

1. Main types of database:

- **Relational Database:**
  - Stores data in tables (rows and columns) best for structured data such as customers info, sales records.)

- **NoSQL Databases:**
  - Stores unstructured or semi-structured data and doesn't use SQL as main query language (e.g Documents, key-value, Column and graphs, also best for dig data, flexible schemas and performance is fast.

- **Object-oriented databases:**
  - Stores data as objects, like in object-oriented programming (OOP) each object include both data and methods and is best for applications built using Java, C++

- **Data Warehouse:**
  - Designed for data analysis and reporting, not daily transactions and it combines data from many sources for business and analytics, large-scale, reporting and decision-making (e.g. Amazon Redshift, Snowflake and google BigQuery.

2) Relational Database Management Systems (RDBMS)

- It stores data in table with rows and columns and can use SQL to manage and query data
e.g. MySQL, Oracle, SQL Server

3) Primary key and foreign key:

- Primary key: is a unique ID for each record in a table.
- Foreign key: is a field that links one table to another using the primary key

4) Database normalization:

- Means organizing data to reduce duplication and improve consistency, helps make the database faster and easier to maintain

5) Database schema:

- It is the structure or layout of the database - shows how fields, tables and relationships are organized

6) Structured, semi-structured and unstructured data:

- Structured data:
Stored in tables (e.g. SQL databases)
- Semi-structured:
Has some structure but flexible (e.g. xml)
- Unstructured
No fixed format (e.g. videos, emails, images)

7) Fact Table vs Dimesion Table:

- Fact Table: store measurable data (e.g. sales amount).
- Dimension Table: store descriptive details (e.g. product_name, customer info).

8) Data model:

- It is a visual Plan that show how data is connected and stored. It helps design databases correctly

9) Database vs Data wavehouse vs Data Lake:

Database: Stores daily operational Data

Data Wavehouse: Stores large amounts of historical data for analysis.

Date Lake: Stores raw data (structured and unstructed) for big data ~~BASE~~.

10) Data mart:

A small part of a data wavehouse focused on one department (e.g. sales or finance), it is easier and fast to use.

Section B: SQL and Data processing.

11) Query language & SQL:

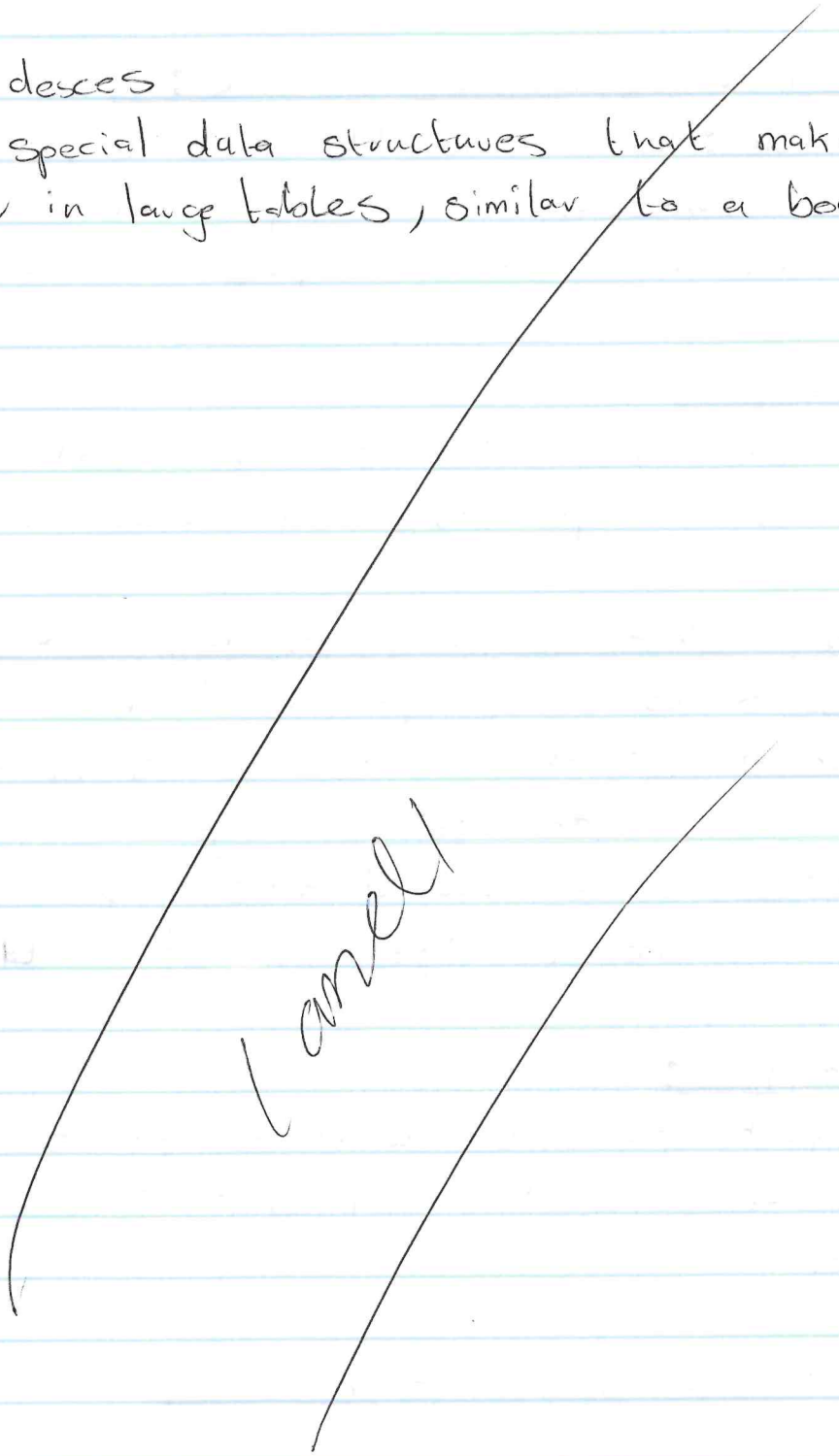
A query language helps you ask the database for information, we use SQL most because its standard, powerful and easy to learn

12) Transactions & ACID properties:

- Transactions is set of database operations done together

13) Indexes
- Are special data structures that make searches faster in large tables, similar to a book's index.

14)



ACID properties: ensure reliability

- Atomic - All or nothing
- Consistent - keeps data valid
- Isolated - Transactions dont interfere.
- Durable - changes stay even after a crash

13) Indexes:

- Are special data structures that make searches faster in large tables, similar to a book's index.

14) Database engine:

- It is core software that reads, writes, and manage data.
- It retrieval and process data and it impacts performance through optimization of queries, caching and storage methods.

15) Views, Stored Procedures, and Triggers:

- Views: A saved SQL query that shows specific data
- Stored Procedures: A saved set of SQL commands.
- Triggers: Runs automatically when something happens (like insert or delete).

16) ETL vs ELT

- ETL - Extract Transform Load: Transform data before storing and it make ~~data~~ data clean after extracting from data sources.

ELT - Extract load Transform: Loads raw data first into the target (e.g. data lake), then transforms it there (store first, then transform)

ETL is traditional systems, ELT is for modern cloud systems.

17) Batch: Processes data in groups at set times and processing handles data in large groups. (e.g. nightly reports)

Stream: Processes data continuously in real time as it arrives.(

18) SQL Joins:
A Join combines data from two or more tables.

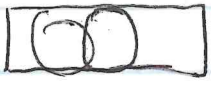
INNER JOIN: Returns (only) matching rows.

~~SELECT * FROM~~
SELECT *
FROM employees
INNER JOIN departments ON employees.dept_id = departments.id);

- left JOIN: All from left table+ matches from right

- Right Join: Opposite of LEFT, all from right table.

19) Full (Outer) Join: All records from both sides



19) Referential integrity:

- It makes sure relationship between tables stay valid - for example, a foreign key must match a valid primary key.

20) Data redundancy:

- It means duplicate data stored in many places, increasing storage costs and risking inconsistencies (e.g. update one copy, others outdated. It affects performance by slowing queries by larger datasets and write multiple updates

SECTION C: DATA MANAGEMENT AND ANALYTICS

21) Cloud vs On-Premise databases:

- Cloud: stored on the internet, managed by providers, scalable (e.g. AWS RDS, Azure SQL)

- On-Premise: stored on company serves, full control but more cost and maintenance (e.g. Self-hosted Oracle)

22) Data Governance:

- It is set of rule and policies to ensure data is accurate, secure and used courectly
- It is a framework of standards for managing data assets and helps to improve compliance.
- It is important in data management to ensure data privacy, reduce risk and foster trust in analytics outcomes.