E.G.S. Pillay Engineering College

An Autonomous Institution Affiliated to Anna University, Chennai | Approved by AICTE, New Delhi
Accredited by NAAC with A++ Grade | Accredited by NBA T1 (B.E. – CSE, B.E. – ECE & B.Tech – IT)
Old Nagore Road, Thethi, Nagore Village, Nagapattinam – 611002, Tamil Nadu, India

Academic Year: 2025-26 | Odd Semester | 3rd Semester B.E / B.Tech Programs | Regulation:2023

Department of Artificial Intelligence & Data Science (AI&DS)

2302AS302 – DATABASE SYSTEMS

Practice Test -1

Part A **3*2=6**

1. Discuss the term Entity set and Relationship set.
2. How to specify the total participation in ER Diagram?
3. Identify the key applications of Database Management Systems.

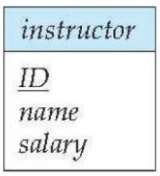
Part B **2*12=24**

1. Illustrate the components and structure of a database system architecture through a well organized diagram.

2. Distinguish the characteristics and purpose of a DBMS by comparing it with a traditional file system in a real-world scenario. Explain how a DBMS helps resolve issues like data redundancy, inconsistency, and limited access control.

Answer

1. Discuss the term Entity set and Relationship set.



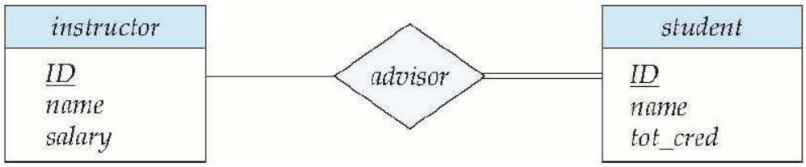
An **entity** is an object that exists and is distinguishable from other objects.
• Example: specific person, company, event, plant
An **entity set** is a set of entities of the same type that share the same properties.
• Example: set of all persons, companies, trees, holidays
An entity is represented by a set of attributes; i.e., descriptive properties possessed by all members of an entity set.
Eg: instructor = (ID, name, salary)

A r**elationship** is an association among several entities.
Example:
44553 (Peltier) advisor 22222 (Einstein)
student entity relationship set instructor entity
A **relationship set** is a collection of similar relationships between two or more entity sets.
The **degree** means the number of entity sets participating:

- **Binary Relationship Set** – between two entity sets (most common).
- **Ternary Relationship Set** – between three entity sets.

2. How to specify the total participation in ER Diagram?

Total participation (indicated by double line): every entity in the entity set participates in at least one relationship in the relationship set.

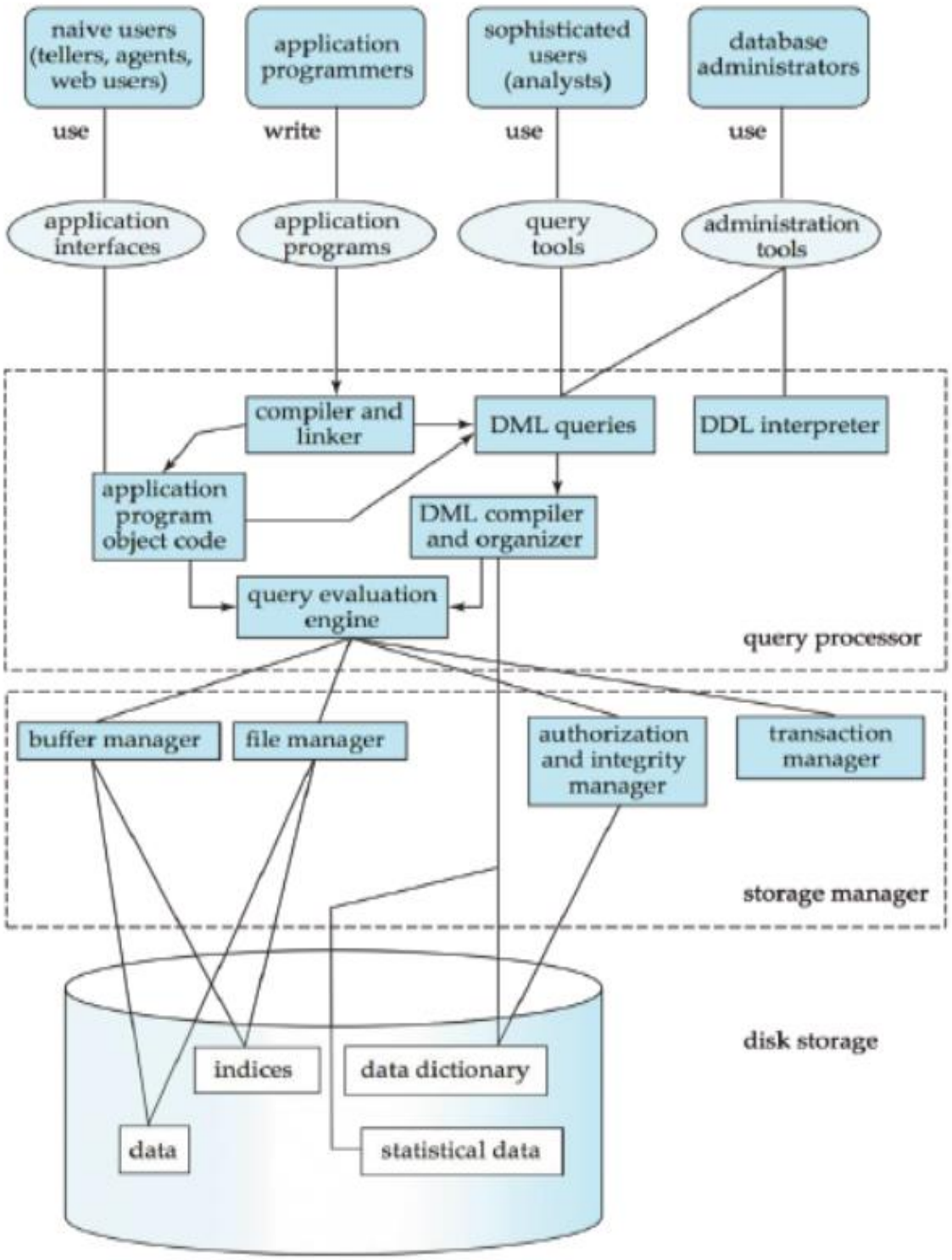


participation of student in advisor relation is total - every student must have an associated instructor

3. Identify the key applications of Database Management Systems.
   DATABASE APPLICATIONS
   - Banking: all transactions
   - Airlines: reservations, schedules
   - Universities: registration, grades
   - Sales: customers, products, purchases
   - Online retailers: order tracking, customized recommendations
   - Manufacturing: production, inventory, orders, supply chain
   - Human resources: employee records, salaries, tax deductions

Part B **2*12=24**

1. Illustrate the components and structure of a database system architecture through a well organized diagram.



Database Users:

Users are differentiated by the way they expect to interact with the system:

Application programmers:

o Application programmers are computer professionals who write application programs. Application programmers can choose from many tools to develop user interfaces.

o Rapid application development (RAD) tools are tools that enable an application programmer to construct forms and reports without writing a program.

Sophisticated users:

o Sophisticated users interact with the system without writing programs. Instead, they form their requests in a database query language.

o They submit each such query to a query processor, whose function is to break down DML statements into instructions that the storage manager understands.

Specialized users :

o Specialized users are sophisticated users who write specialized database applications that do not fit into the traditional data-processing framework.

o Among these applications are computer-aided design systems, knowledge base and expert systems, systems that store data with complex data types
(for example, graphics data and audio data), and environment-modeling systems.

Naïve users :

o Naive users are unsophisticated users who interact with the system by invoking one of the application programs that have been written previously.

o For example, a bank teller who needs to transfer $50 from account A to account B invokes a program called transfer. This program asks the teller for the amount of money to be transferred, the account from which the money is to be transferred, and the account to which the money is to be transferred

.

Database Administrator:

Coordinates all the activities of the database system. The database administrator has a good understanding of the enterprise’s information resources and needs.

Database administrator's duties include:

o **Schema definition:** The DBA creates the original database schema by executing a set of data definition statements in the DDL.

o **Storage structure and access method definition.**

o **Schema and physical organization modification:** The DBA carries out changes to the schema and physical organization to reflect the changing needs of the organization, or to alter the physical organization to improve performance.

o **Granting user authority to access the database:** By granting different types of authorization, the database administrator can regulate which parts of the database various users can access.

Specifying integrity constraints.

Monitoring performance and responding to changes in requirements.

Query Processor:

The query processor will accept query from user and solves it by accessing the database.

Parts of Query processor:

DDL interpreter

This will interprets DDL statements and fetch the definitions in the data dictionary.

DML compiler

a. This will translates DML statements in a query language into low level instructions that the query evaluation engine understands.

b. A query can usually be translated into any of a number of alternative evaluation plans for same query result DML compiler will select best plan for query optimization.

Query evaluation engine

This engine will execute low-level instructions generated by the DML compiler on DBMS.

Storage Manager/Storage Management:

A storage manager is a program module which acts like interface between the data stored in a database and the application programs and queries submitted to the system.

Thus, the storage manager is responsible for storing, retrieving and updating data in the database.

The storage manager components include:

o **Authorization and integrity manager:** Checks for integrity constraints

and authority of users to access data.

o **Transaction manager:** Ensures that the database remains in a consistent state although there are system failures.

o **File manager:** Manages the allocation of space on disk storage and the data structures used to represent information stored on disk.

o **Buffer manager:** It is responsible for retrieving data from disk storage into main memory. It enables the database to handle data sizes that are much larger than the size of main memory.

o **Data structures implemented by storage manager.**

o **Data files:** Stored in the database itself.

o **Data dictionary:** Stores metadata about the structure of the database.

o **Indices:** Provide fast access to data items.

Evaluation Criteria:

1. **Question – 12 Marks**

    Diagram – 4 Marks

    Database User – 3 Marks

    Storage Manager – 3 Marks

    Query Processor & Disk Storage – 2 Marks

2. Distinguish the characteristics and purpose of a DBMS by comparing it with a traditional file system in a real-world scenario. Explain how a DBMS helps resolve issues like data redundancy, inconsistency, and limited access control.

A **Database Management System (DBMS)** is a software system that enables efficient storage, retrieval, and management of data in a structured manner. In contrast, a **traditional file system** manages data through individual files stored in directories, without built-in mechanisms for data integrity, security, or consistency.

To understand the difference, consider a **real-world example** of a **college managing student records** such as names, roll numbers, courses, grades, and attendance.

Real-World Scenario: College Student Records

- In a **file system**, each department (e.g., academics, administration, hostel) maintains its own student data in separate files.
- In a **DBMS**, all departments access a **centralized database**, ensuring shared, consistent, and secure access to data.

Aspect	Traditional File System	Database Management System (DBMS)
Data Redundancy	High – same data stored in multiple files	Low – data is centralized and normalized
Data Inconsistency	Common due to duplicate data	Reduced via constraints and controlled access
Data Access	Manual, program-specific	Query-based (SQL), flexible and efficient
Security / Access Control	Minimal – OS-level only	Role-based access, fine-grained control
Data Integrity	Hard to enforce	Enforced through constraints (PK, FK, CHECK)
Concurrent Access	Risky – may lead to corruption	Supports ACID transactions for consistency
Backup and Recovery	Manual and error-prone	Automated and robust mechanisms
Scalability	Limited	Highly scalable with indexing and partitioning
Data Relationships	Handled through custom code	Built-in relational model supports joins, keys
Example	File folders, spreadsheets	MySQL, PostgreSQL, Oracle, SQLite

Issues in File Systems and DBMS Solutions

1. Data Redundancy

- **Problem:** Multiple files contain the same data (e.g., student name repeated in grade file, fee file, attendance file).
- **DBMS Solution:** Through normalization, data is stored once and referenced via keys. This saves storage and improves consistency.

2. Data Inconsistency

- **Problem:** A change in one file (e.g., address update) may not reflect in others, leading to confusion and errors.
- **DBMS Solution:** Centralized control ensures that updates are made once and reflected system-wide, avoiding conflicting data.

3. Limited Access Control

- **Problem:** Anyone with access to a file can read or modify it. There's no user-specific restriction or auditing.
- **DBMS Solution:** DBMS allows administrators to define roles and permissions. For example, students may only view their own data, while teachers can update marks.

4. Poor Data Integrity

- **Problem:** There's no restriction on data types, formats, or duplication in files.
- **DBMS Solution:** Integrity constraints (e.g., NOT NULL, UNIQUE) ensure valid and accurate data entry.

5. Lack of Concurrency Handling

- **Problem:** Two users updating the same file can overwrite each other's data.
- **DBMS Solution:** Transactions and locking mechanisms in DBMS ensure safe, concurrent data access.

Evaluation Criteria:

2. Question – 12 Marks

Differences (10 Points) – 8 Marks

Real world Scenario - 4 Marks