

ASSIGNMENT : 1&2

Data Base

Management

System

Name: K. Soomarath Reddy

VTU : 27832

Dept Name: CSE (AI & ML)

⑤

(at)

1) Explain the details about database system architecture in a neat diagram?

A) Database System Architecture

The database system architecture can be divided into three main parts: users, query processor, and storage manager, with the disk storage at the bottom.

1. Users

Different types of users interact with the database:

- Naive users (tellers, agents, web users) → use application interfaces.
- Application Programmers → write application programs.
- Sophisticated users (analysts) → use query tools.
- Database Administrators (DBAs) → use administration tools.

2. Query Processors

This is responsible for interpreting and executing queries.

- Application Program Object code: Generated by compiler and linker.
- DML Queries: Data Manipulation Language (e.g., SELECT, INSERT, UPDATE, DELETE).
- DDL Interpreter: Interprets schema definitions.
- DML Compiler and Optimizer: Checks Syntax and optimizes queries.

→ Query Evaluation Engine: Executes optimized queries.

→ Together, these form the Query Processor.

### 3. Storage Manager

→ The storage manager controls how data is stored and retrieved.

→ Buffer manager: minimizes disk I/O by storing frequently accessed data in memory.

→ File manager: manages allocation of space and file structures.

→ Authorization and integrity manager: Enforces data security and integrity constraints.

→ Transaction manager: Enforces consistency, concurrency control, and recovery.

### 4. Disk Storage

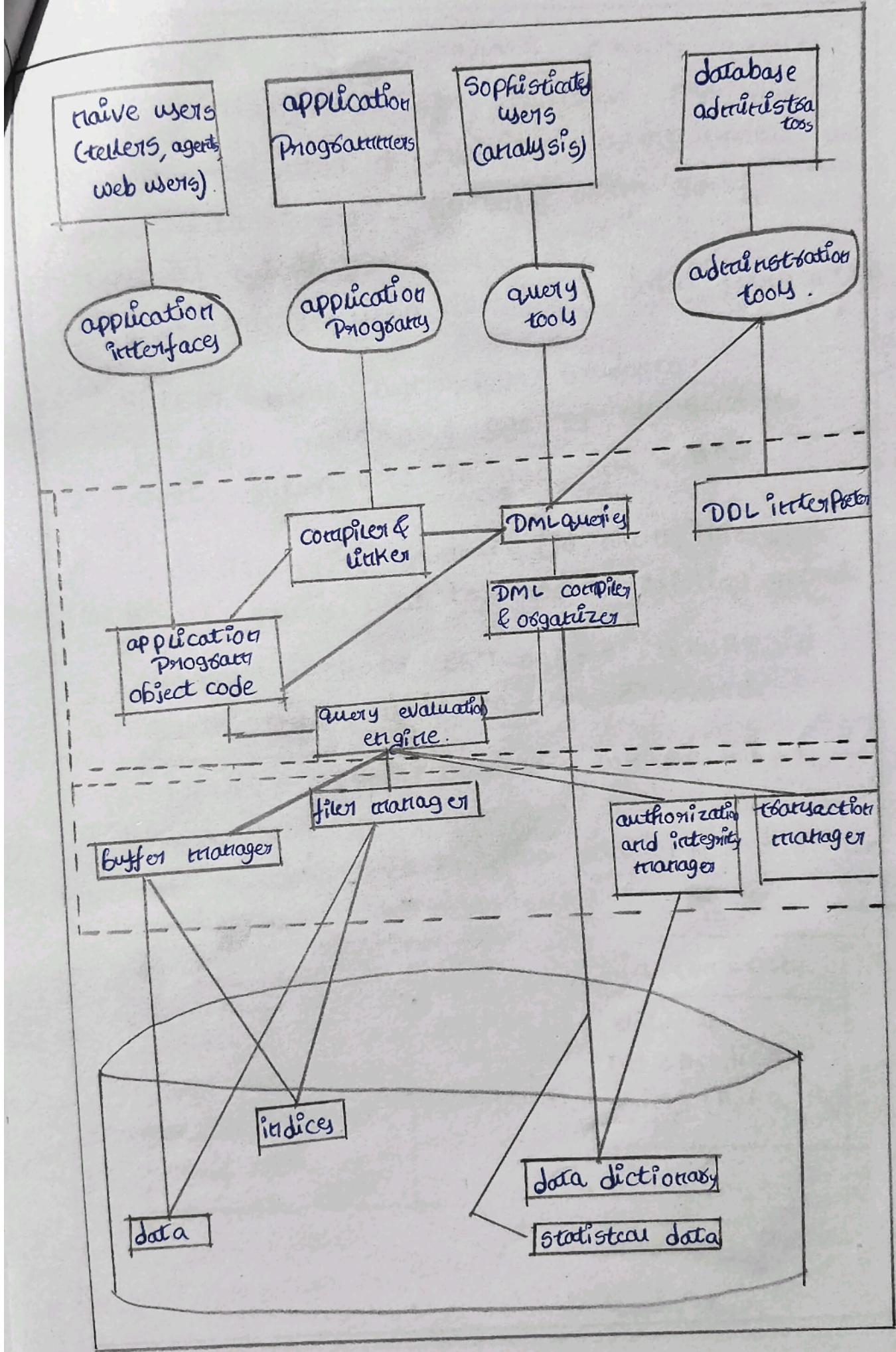
→ This is the Physical Layer where actual data resides.

→ Data: Tables and records.

→ Indices: used for fast searching.

→ Data Dictionary: stores metadata.

→ Statistical Data: used for query optimization.



a) Explain in details about various queries and joins with the suitable examples?

Queries are used to retrieve, insert, update, and delete data from a database using SQL.

Types of queries:

1. Select query - used to fetch data from a table

SQL

SELECT name, age FROM students;

FETCHES name and age of all students;

2. Insert query - used to add new records.

SQL

INSERT INTO students (id, name, age)

3. Update query - used to modify existing records

SQL

UPDATE students SET age = 21 WHERE id

4. Delete query - used to remove records.

SQL

DELETE FROM students WHERE id = 1.

Joined relations

Join operations take two relations and return as a result another relation

Join Types
inner Join
left outer Join
right outer Join
full outer Join

Join conditions
natural on predicate using $(A_1, A_2, \dots, A_n)$

## Database Example - join

instructors

ID	name	dept_name
10101	Sririvasan	COMP-SCI
12121	wu	Finance
15151	Mozart	MUSIC.

Teaches

ID	course_id
10101	CS-101
12121	FIN-201
76766	BIO-101

## INNER JOIN

Returns only the matching rows from both tables based on a given condition

SELECT instructors.ID, name,

course\_id

FROM instructors

INNER JOIN teaches

ON instructors.ID = teaches.ID;

ID	name	course_id
10101	Sririvasan	CS-101
12121	wu	FIN-201

### Full Outer JOIN

Returns all rows from both tables, matching rows where available, and filling NULLS where there's no match.

```
SELECT instructors.ID, name, course_id
FROM instructors
RIGHT JOIN teachers ON instructors.ID=teachers.ID;
```

ID	name	course_id
10101	Srinivasan	CS-101
12121	Wu	FIN-201
15151	Mozart	NULL
76766	NULL	BIO-101

### EQUI JOIN

A type of INNER JOIN that uses an equality (=) operator to match rows

```
SELECT instructors.ID, name, course_id
```

```
FROM instructors, teachers
```

```
WHERE instructors.ID=teachers.ID;
```

ID	name	course_id
10101	Srinivasan	CS-101
12121	Wu	FIN-201

Cross  
Return  
row

### Left Outer Join

Selects all rows from the left table and the matching rows from the right table. If no match, NULLS are shown for right table columns.

SELECT instructors.ID, name, course\_id  
FROM instructors

LEFT JOIN teaches

ON instructors.ID = teaches.ID;

ID	name	course-id
10101	Sriravasan	CS-101
12121	WU	FIN-201
15151	Mozart	NULL

### Right Outer Join

Selects all rows from the right table and the matching rows from the left table. If no match, NULLS are shown for left table columns.

SELECT instructors.ID, name, course\_id  
FROM instructors

RIGHT JOIN teaches

ON instructors.ID = teaches.ID;

ID	name	course-id
10101	Sriravasan	CS-101
12121	WU	FIN-201
76766	NULL	BIO-101

ble and the CROSS JOIN  
 Return the cartesian Product of two tables. every  
 row from the first table joined with every  
 row from the second table  
 SELECT name, course\_id  
 FROM instructors  
 CROSS JOIN teachers

Name	course_id
Sriravasari	CS-101
Sriravasari	FIN-201
Sriravasari	BIO-101
WU	CS-101
WU	FIN-201
WU	BIO-101
Mozart	CS-101
Mozart	FIN-201
Mozart	BIO-101

### Natural Join

A Type of join that automatically joins table using columns with the same name and compatible data types

SELECT \*  
 FROM instructors  
 NATURAL JOIN teachers;

ID	Name	dept-Name	course_id
10101	Sriravasari	comp sci	CS-101
12121	WU	Finance	FIN-201