Assignment - I

Name : OP. Pavani

VTU : 30283

Course code : 10211CA207

Course title : Data Base management systems.

5

3. third Normal form (3NF):

Rule:

* Table must be in 2NF
* NO transitive dependency (non-key attribute should depend on another non-key attribute)

Example (Before 3NF):

Student ID	Student Name	Dept ID	Dept Name
1	Ravi	D1	Maths

Here, Dept Name depends on Dept ID not directly on Student ID

After 3NF

Should Student table.

Student ID	Student Name	Dept ID
1	Ravi	D1

Department table

Dept ID	Dept Name
D1	maths.

4) Boyce-Codd Normal form (BCNF):

Rule:

* Table must be in 3NF
* for every functional dependency (X→Y), X should be a superkey

Example:

Course	Instructor	Room
DBMS	Ravi	R1
DBMS	Ravi	R1

Explain about deadlock and its handling in OSNS

A deadlock is a condition in a multiprogramming System where two or more processes are waiting indefinitely for resources held by each other, causing all of them to remain blocked -forever

Ex:

- Process P_1 holds Resources R_1 and waits for R_2
- process P_2 holds Resources R_2 and wait for R_1

→ Both Process wait forever deadlock occurs

Necessary conditions for deadlock

A deadlock can occur only if all there four conditions hold Simultaneously

1. mutual Exclusion: only one process can use a resource at a time.
2. Hold and wait: A process holding a resource waiting for others.
3. NO preemption: Resources cannot be forcibly taken from a process
4. Circular wait: A Circular Chain process exits, each waiting for a resource held by the next process

Deadlock Handling methods:

1 Deadlock Prevention: Design the system in such a way that at least one of the four necessary conditions never occurs.

Ex: Don't allow hold-and-wait.

1) Normalization and its various types of Normalization

Normalization is a proces of organizing the data in database to avoid data redundancy and improve data integrity.

It divides large tables into smaller related tables and links them using foreign keys.

Objectives:

1. To eliminate data redundency.
2. To avoid Update, insert, and delete anomalies
3. To ensure data consistency and integrity.
4. To make database structure simple and efficient

Types of Normalization:

Rule:

* Each column should contain atomic (indivisible) values
* NO Repeating groups or arrays are allowed

Ex: (Before 1NF)

Student ID	Name	Subjects
1	Ravi	Math, Science

(After 1NF):

Student ID	Name	Subject
1	Ravi	Math
1	Ravi	Science

Explanation:

* Inner query finds DeptID of CSE → 101
* Outer query selects all students with Dept ID 101

Result: Ravi, Arjun.

Example: Nested query with IN

Find students who belong to CSE or ECE

```
SELECT Name
FROM STUDENTS
WHERE DeptID IN (
    SELECT DeptID
    FROM Departments
    WHERE DeptName IN ('CSE', 'ECE')
);
```

2. Joins:

A Join is used to combine data from two or more tables based on a related column

Types of Joins:

a) INNER Join

* Returns only the matching rows from both tables

```
SELECT Students.Name,
Department.DeptName
FROM Students
```

RAID storage and its types

RAID Stands for Redundant Array of Independent Disks. It is a data Storage technology that Combines multiple physical hard drives into a Single logical unit to improve Performance, fault tolerance, and data reliability.

RAID is mainly used in servers and high performance systems where data availabilty and speed are important.

Objectives of RAID

* To increase Storage capacity
* To improve speed
* To Provide data redundancy
* To enhance System reliability.

Working Principle:

RAID uses a technique called data Striping (dividing data into blocks and Storing them across multiple disks) and parity (extra information used for recovery if one disk fails).

Types of RAID levels:-

RAID 0 (Striping): Data is divided into blocks and Stored across multiple disks.

RAID 1 (Mirroring): Same data is copied on two or more disks.

RAID 2 (Bit-level Striping): Data is Striped at bit level with error Correction code (ECC).

naive USER
(TELLERS AGENTS
WEB-USERS)

USE

Application
Interfaces

Application
Programmers

WRITE

Application
programs

Sophisticated
USERS (Analysts

USE

Query
TOOLS

Data BASE
Administrator

USE

Administration
TOOLS

Application
programing
Code

Compiler
Linker

DML
Query

DDL
interpreter

Query
Processor

Query evaluation
Engine

DML Compiler
and organizer

Buffer
Manager

File
manager

Authorized &
integrity manager

Transaction
manager

INDEXES

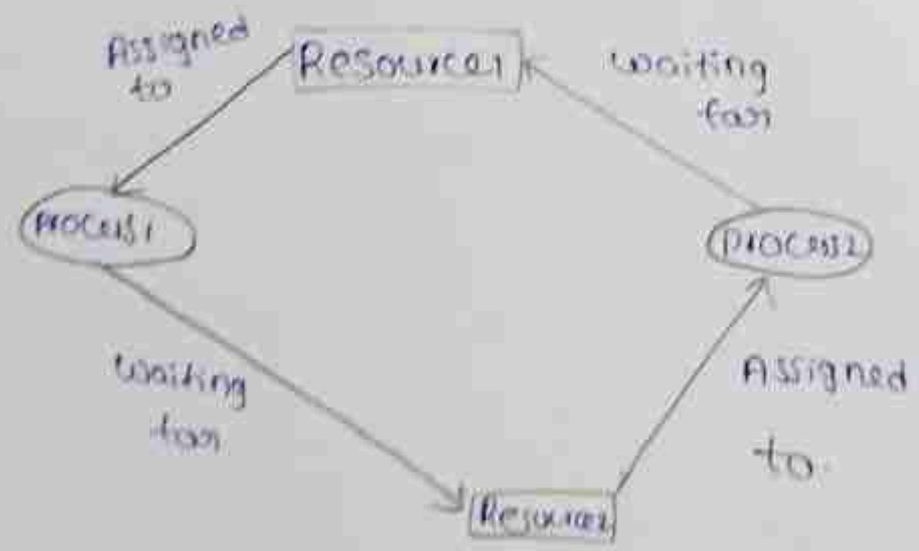
DATA

Statistical
DATA

Query
Storage

2 Deadlock Avoidance: Dynamically check resource requests to ensure the system never enters an unsafe state.
Ex: Banker's Algorithm.
3 Deadlock Detection and Recovery: Allow deadlock to occur, detect it using an algorithm and then recover.
4 Deadlock Ignorance: The system assumes that deadlock never occur used in most OS because deadlocks are rare.

Deadlock handling is essential in operating systems to ensure smooth execution of process and efficient resource utilization. Different techniques are used depending on system requirements and complexity.



Explain in detail about Data base architecture with neat diagram.

Database Architecture describes how a database management system is designed, how its components are arranged, and how users interact with the database.

It defines:

* physical structure
* logical structure
* External Interaction.

Key Components of Database Architecture

1. Users : End users or applications that request data.
2. DBMS software : manages data storage, retrival, security, & ccurency.
3. Database (storage) - stores actual data in files tables, and indexes.
4. Application layer: Handles business logic and rules.

Explain in detail about nested queries & solve with suitable Examples

* A nested query (or subquery) is a query inside another query.
* The inner query runs first and its result is used by the outer query.
* They are usually written in the where, having or from clause.

Example :

Student ID	Name	Age	Dept ID	Dept Name	Phone no
1	Ravi	20	101	CSE	98~~7~~[illegible]
2	Sita	21	102	ECE	9876543[illegible]
3	Arjun	22	103	Mechanical	9~~876~~291
4	meena	23	101	civil	9182763[illegible]

Example:1 : simple Nested Query -

Find the students who belong to CSE department

```
SELECT Name
FROM students
WHERE DeptID = (
    SELECT DeptID
    FROM Departments
    WHERE DepartmentName = 'CSE'
);
```

RAID 3 (Byte level Stripping with parity): Data is Striped
a byte level and a seperate disk stores Parity Info.
RAID 4 (Block-level Stripping with Parity): Parity Information
is stored on a dedicated disk.
RAID 5 (Block level Stripping with Distributed Parity):
Parity is distributed across all disks
RAID 6 (Double Distributed Parity): Similar to RAID 5
but uses two parity blocks for extra protection.
RAID 10 (combination of RAID 1 + RAID 0): Combined
mirroring and striping.
RAID Storage technology provides a balance between
performance and reliability.
The selection of RAID level depends on system
needs, whether speed / cost and data safety is the
priority.

RAID 1

(a) presentation layer (client layer).

* User interface: web browser, mobile app, desktop app.
* Collects user input and shows results

(b) Application Layer (Business logic layer)

* middle tier that processes requests
* performs. Authentication. validation (calculation and send)

(c) Database layer (DBMS layer).

* Stores actual data.
* DBMS like mysql, oracle, postgre sql manages data, concurrency, backup, and security.

Advantages

* High security (users never directly access database).
* Scalabiity (supports thousands of users).
* Better performance with caching & load balancing

Assignment

Name:- OP Pavani

VTU :- 30283

Sub :- DBMS

5

on students.DeptID = Departments.DeptID;

ID didn't exist, Dept Name would be Null

() Right Join

* opposite of LEFT Join
* Returns all rows from departments,
and matched students.

full outer Join

* Returns all rows from both tables.
(match on no match.)

* In my SQL we simulate it with Union

SELECT Students. Name,
Departments. Dept Name
From Students
LEFT Join Departments
On Students. Dept ID = Departments

on Students. Dept ID = Departments. Dept ID;

Result:

Ravi CSE

Sita EEE

Arjun CSE

Meena Mechanical

(eft Join (outer Join).

Returns all rows from the left table (students)

and matched rows from departments.

if no match → NULL

SELECT Students. Name,

Departments. Dept Name

FROM Students.

Left Join Departments.

5) Fifth Normal form (5NF):

Rule

* Table must be in 4NF
* Should not have join dependency.

Ex:-

Relation R = {Subject, Lecture, Semester}

Subject	Lecturer	Semester
Computer	Anshika	Semester I
Computer	John	Semester I
Math	John	Semester II
Math	John	Semester II
Chemistry	Praveen	Semester I

R_1 = {Semester, Subject}

R_2 = {Subject, Lecturer}

R_3 = {Semester, Lecturer}

All three relations are now in 5NF

Assignment

Name :- Op. Pavani

MTU :- 30283

Sub:- BBMS

5 Qtr

Fourth Normal form (4NF):

Rule:

* Table must be in BCNF
* There should be no multi valued dependencies

Ex:

Student	Hobby	Language
Ravi	Cricket	English
Ravi	music	Hindi

Here, "Hobby" & "language" are independent multi-valued facts.

After 4NF:

Student Hobby:

Student	Hobby
Ravi	cricket
Ravi	MUSIC

Student language:

Student	language
Ravi	English
Ravi	Hindi

2) Second Normal form (2NF):

Rule:

* Table must be in 1NF
* No partial dependency - i.e, a non-key attribute should not depend on part of a composite key

Ex: (Before 2NF):

Student ID	Course ID	Student Name	Course Name
1	C1	Ravi	DBMS

Here:

* Student ID + Course ID = Primary key.
* Student Name depends only on Student ID (Partial dependence)

After (2NF): Student table:

Student ID	Student Name
1	Ravi

Course table:

Course ID	Course Name
C1	DBMS

Enrollment table:

Student ID	Course ID
1	C1