SNS COLE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

COURSE NAME : 19IT408 - DATABASE MANAGEMENT SYSTEMS

II YEAR / III SEMESTER

Unit 1- Introduction

Topic 1 : Purpose of Database System

PURPOSE OF DATABASE MANAGEMENT SYSTEM

The purpose of the data management system is to follow

- it is used to optimize data management
- it transforms data into information
- Data management systems were developed to handle the following difficulties of typical file-processing systems supported by conventional operating systems.
- Data redundancy
- Data inconsistency
- Difficulty in accessing data
- Integrity problems

- Atomicity problems
- Concurrent access anomalies
- Security problems
- Data isolation

DATA REDUNDANCY

Data redundancy means duplication of data at several places. Data repetition is possible that the same information may be duplicated in different files this leads to data redundancy and results in memory wastage.

Example:

Student ID	Student Name	Teacher id	Teacher Name
1	Irum	1	Shehla
2	Sajid	2	Kiran
3	Javed	1	Shehla

The student 1 and 3 are learning from teacher Shehla, and student 2 is learning from teacher Kiran. Here, the teacher ID and Teacher name are repeated two times.

Solution of Data Redundancy

Teacher ID and teacher name can be stored in a separate table. The new student table and the teacher table are as follows.

Student ID	Student Name	Teacher id
1	Irum	1
2	Sajid	2
3	Javed	1

Teacher_id	Teacher_Name
1	Shehla
2	Kiran

DATA INCONSISTENCY

Data inconsistency occurs when various copies of the same data may no longer get matched. Because of data redundancy, it is possible that data may not be in a consistent state.

Student ID	Student Name	Teacher id	Teacher Name
1	Irum	1	Shehla
3	Javed	1	Shehla

Student ID	Student Name	Teacher id	Teacher Name
1	Irum	1	Shehla Khan
3	Javed	1	Shehla

Solution of Data Inconsistency

Teacher id and teacher name can be stored into a separate table. New student table and the teacher table are as follows.

Student ID	Student Name	Teacher id
1	Irum	1
2	Sajid	2
3	Javed	1

Teacher id	Teacher Name
1	Shehla
2	Kiran

DIFFICULTY IN ACCESSING DATA

The conventional file system does not allow to retrieval of the desired data in an efficient and convenient manner. Accessing data is not convenient and efficient in the file processing system.
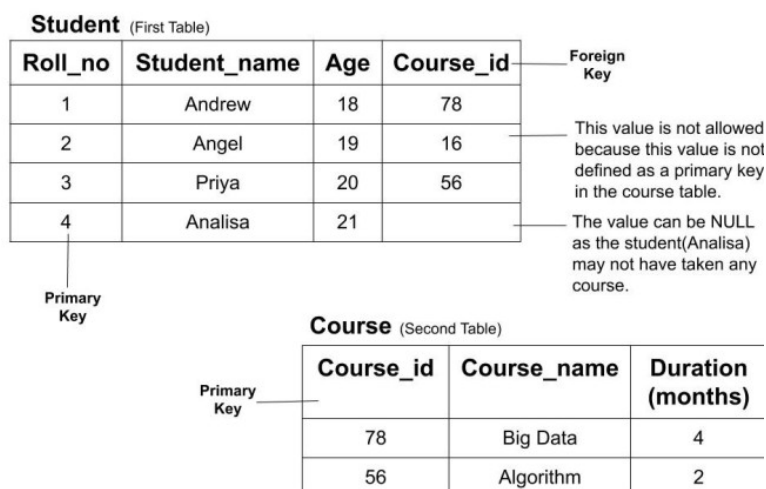
Example:

One of the bank officers needs to find out the names of all customers who live within a particular postal code area. If there is no application program for this men, the officer has 2 alternatives:

1.preparing the list manually from the list of all customers

2.ask system programmer to write the necessary application programs

INTEGRITY PROBLEMS

Data integrity means that the data contained in the database is both correct and consistent, for this purpose, the data stored in the database must satisfy the correct constraints.

Example: Let us suppose we have two tables of the student (student_id, name, age, course id) and course (course id, course name, duration). Now, if any course id is present in the student table which is not there in the course table then this is not allowed. The course id in the student table should either be null or if any course id is present in the student table then it should also be present in the course table. This is how referential integrity is maintained.

Student (First Table)

Roll_no	Student_name	Age	Course_id
1	Andrew	18	78
2	Angel	19	16
3	Priya	20	56
4	Analisa	21	

Course_id — Foreign Key

This value is not allowed because this value is not defined as a primary key in the course table.

The value can be NULL as the student(Analisa) may not have taken any course.

Primary Key

Course (Second Table)

Primary Key

| Course_id	Course_name	Duration (months)
78	Big Data	4
56	Algorithm	2

REFERENTIAL INTEGRITY

ATOMICITY PROBLEMS

Any operation on a database must be atomic (i.e) it must happen in its entirety or not at all.

Example:

If Remo has account A having $30 in his account from which he wishes to send $10 to Sheero's account, which is B. In account B, a sum of $ 100 is already present. When $10 will be transferred to account B, the sum will become $110. Now, there will be two operations that will take place. One is the amount of $10 that Remo wants to transfer will be debited from his account A, and the same amount will get credited to account B, i.e., into Sheero's account. Now, what happens - the first operation of debit executes successfully, but the credit operation, however, fails. Thus, in Remo's account A, the value becomes $20, and to that of Sheero's account, it remains $100 as it was previously present.
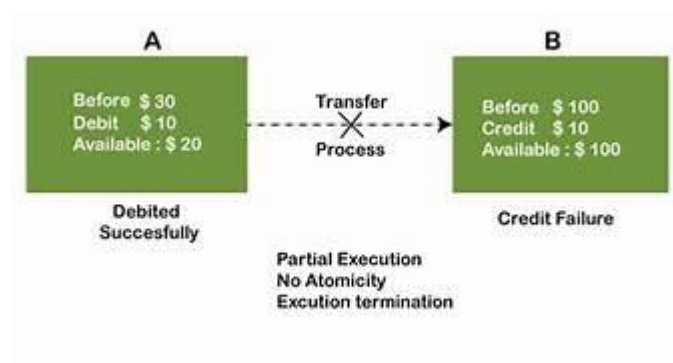
the above diagram, it can be seen that after crediting $10, the amount is still $100 in account B. So, it is not an atomic transaction.
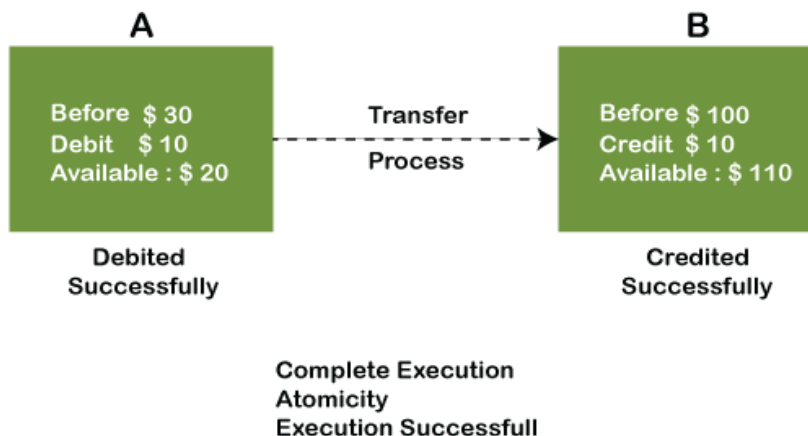
The below image shows that both debit and credit operations are done successfully. Thus the transaction is atomic.

CONCURRENT ACCESS ANOMALIES

Multiple users are allowed to access data simultaneously, this is for the sake of better performance and faster response.

Example:

Consider an operation to debit (withdrawal) an account. The program reads the old balance, calculates the new balance, and writes new balance back to database. Suppose an account has a balance of Rs.5000. now, a concurrent withdrawal of Rs.1000 and Rs.2000 may leave the balance Rs.4000 or Rs.3000 depending upon their completion time rather than the correct value of Rs.20000

SECURITY PROBLEMS

Data security means prevention of data access by unauthorized users. Database should be accessible to users in limited way. Each user should be allowed to access data concerning his requirements only.

Example:

If a student can access his data in the college library then he can easily change books issued date also he can change his fine details to zero

To overcome these difficulties, the database management system (DBMS) was developed

DATA ISOLATION

Data isolation is physical, network, and operational separation of data to keep it safe from external cyberattacks and internal threats.

Example:

Applications of DBMS

- Railway Reservation System
- Library Management System
- Banking
- Education Sector
- Credit card exchanges
- Social Media Sites
- Broadcast communications
- Online Shopping
- Airline Reservation System
- Healthcare

Advantages of Database Management System

- Better data transferring
- Better data security
- Better data integration
- Better decision making
- Increased end-user productivity:
- Simple:

- Data abstraction:

- Reduction in data Redundancy:

- Application development:

- Scalability:

- Data sharing

- Data organization

- The atomicity of data can be maintained

- Data consistency and accuracy

- Improved data security

- Efficient data access and retrieval

- Scalability and flexibility

- Improved productivity