

INTRODUCTION TO DATABASE SYSTEMS

Week 1 – Lesson 2

LEARNING OUTCOME

- Understand Flat files and their limitations,
- Discuss Database Models

FLAT FILES

A flat file contains records which have no structured interrelationship. They are typically called text files, because data is stored in plain text.

In the Flat file, each line of the plain text file holds only one record.

These records are separated using delimiters, such as tabs and commas.

LIMITATIONS OF FLAT FILES

- The File based system is limited to a smaller size and cannot store large amounts of data.
- Data in flat files are stored in an unstructured way because of which retrieval and update becomes difficult.

LIMITATIONS OF FLAT FILES CONTD'

- Data security is poor in flat files.
- Flat based systems cannot support complicated queries, data recovery etc.
- It increased Redundancy and inconsistency.
- It is difficult to share the data stored in flat files with multiple users.

NEED FOR A DBMS

To address the challenges of flat files a DBMS is necessary because of the following:-

- All related data is *grouped* together in an organized form.
- It is easy to *retrieve* data in a DBMS because it is *structured* and organized in the form of rows and columns.
- Data *security* is good because *access controls* can be put in place to protect the data from **unauthorized access**
- DBMS provide *mechanisms* of sharing data with *multiple users* and applications

DATABASE MODELS

A **database model** determines the **logical structure** of a database and determines in which manner data can be stored, organized and manipulated on a fundamental basis.

TYPES OF DATA MODELS

1. Relational Database Model

- A relational model is a system where data is organized in **two-dimensional tables** using **rows** and **columns**.
- Every table in a database has a **key field** which uniquely identifies each record. This type of model is the most widely used DBMS.

EXAMPLE OF Relational Database Model

The diagram illustrates the components of a relational database table:

- attributes**: Points to the column headers (SID, SName, SAge, SClass, SSection).
- column**: Points to the vertical axis of the table.
- tuple**: Points to a specific row in the table.
- table (relation)**: Points to the entire horizontal structure of the table.

SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

Main highlights of this model

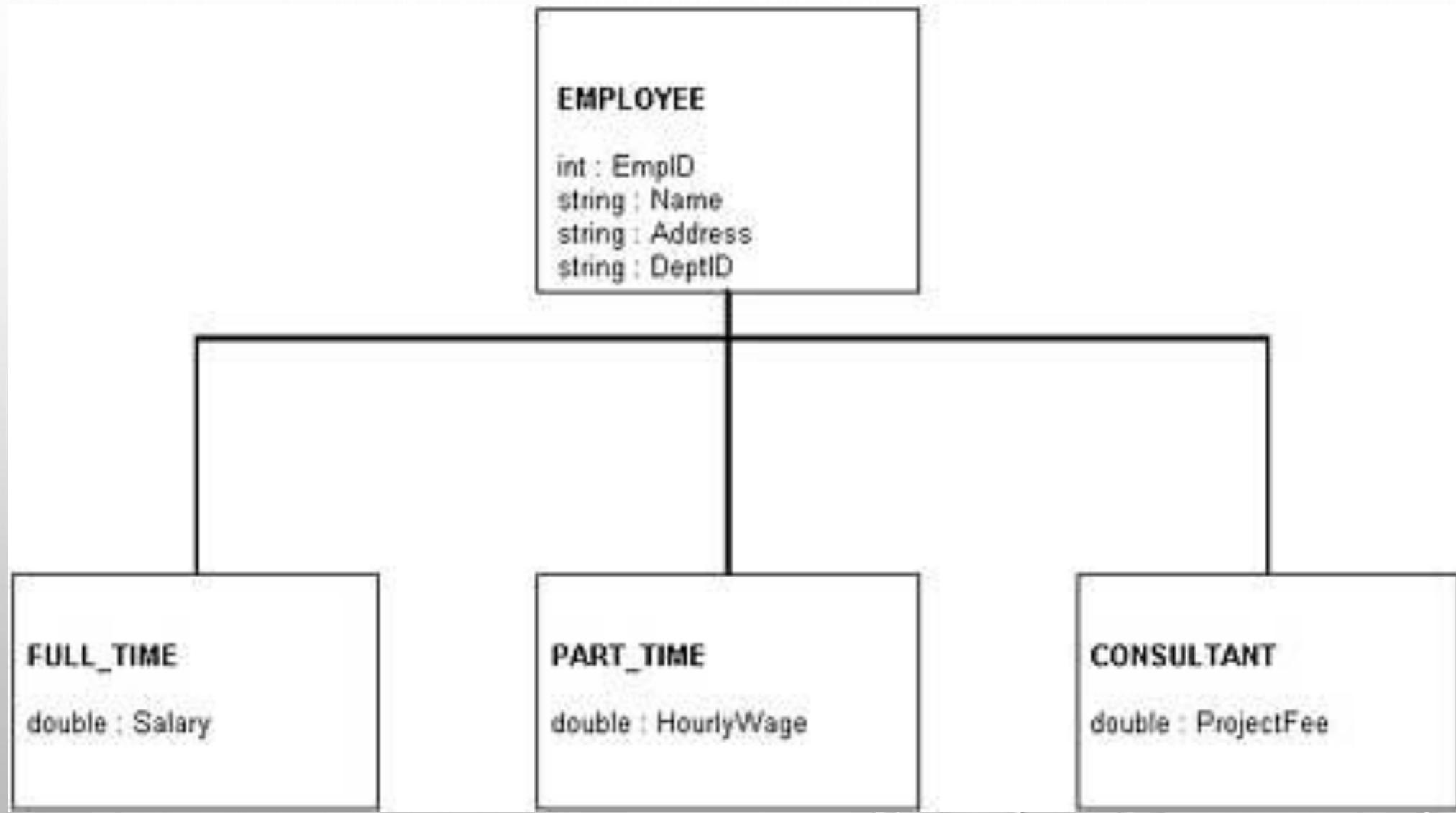
- Data is stored in **tables** which are also called **relations**.
- Relations can be **normalized**.
- In normalized relations, values saved in fields as **atomic values**.
- Each row in a relation represents a **single record**
- Each column in a relation represents the **fields** in of tables

2. Object Oriented Database

It is a system where information or data is represented in the form of **objects** which are used in object-oriented programming.

- It is a combination of **relational database concepts** and **object-oriented principles**
- OOPs principles are data **encapsulation**, **inheritance**, and **polymorphism**.

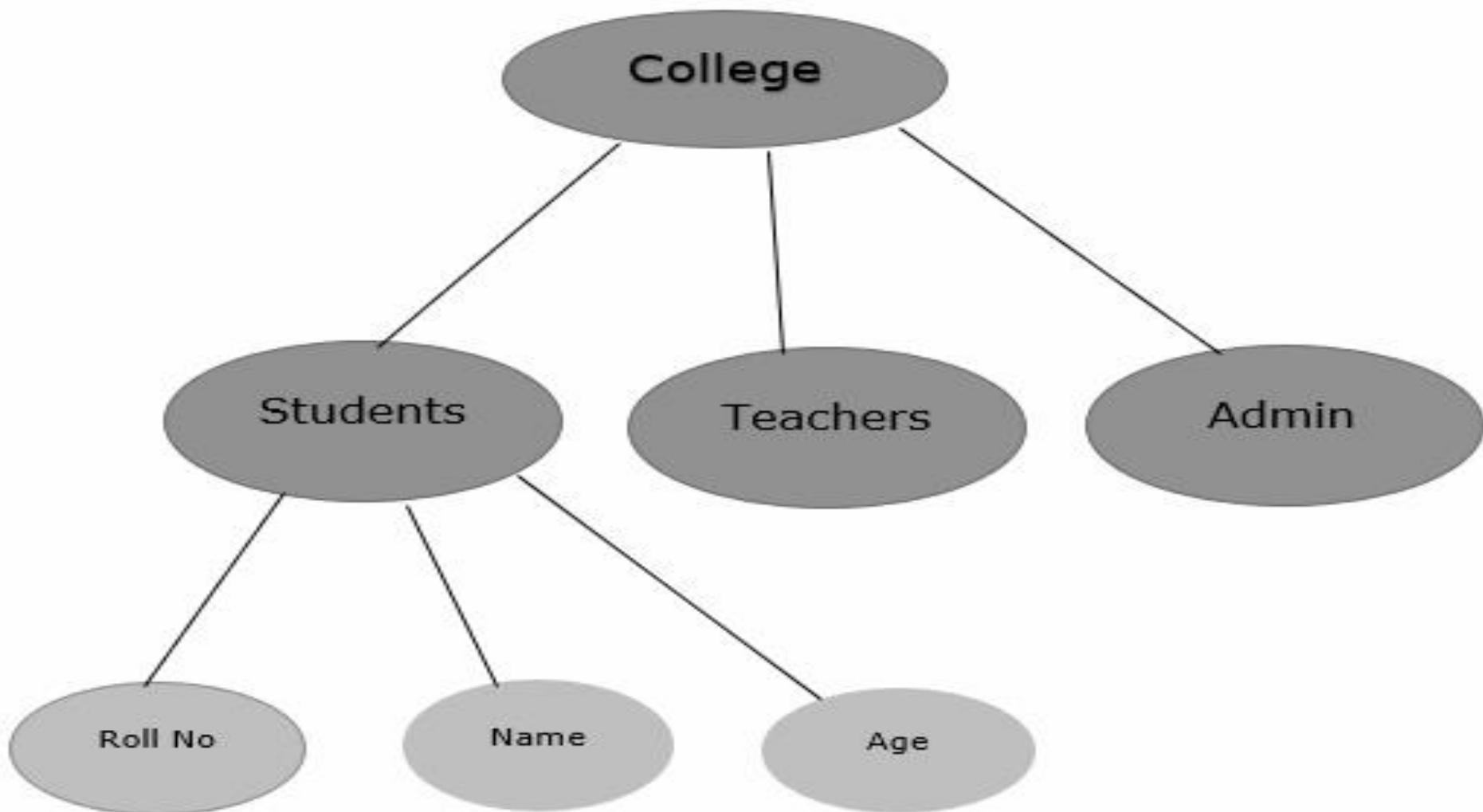
EXAMPLE OF Object Oriented Database Model



3. Hierarchical Database Model

- It is a system where the data elements have a **one to many relationship** (1: N). Here data is organized like a **tree** which is similar to a folder structure in your computer system.
- The hierarchy starts from the **root node**, connecting all the **child nodes** to the **parent node**.

EXAMPLE OF Hierarchical Database Model



4. Network Database Model

- A **Network database model** is a system where the data elements maintain one to **one relationship (1: 1)** or **many to many relationship (N: N)**.
- In this type of model, a child can be linked to **multiple parents**, a feature that was not supported by the hierarchical data model. The **parent nodes** are known as **owners** and the **child nodes** are called **members**

EXAMPLE OF Network Database Model

