Course Details

Course Title	:	Database Management System
Semester	:	Third
Course Code	:	22319
TH-ESE	:	70
TH-PA	:	30
PR-ESE	:	25#
PR-PA	:	25

Syllabus Details

Sr.no	Chapter Name	Marks
1	Database System Concepts	12
2	Relational Data Model	18
3	Interactive SQL and Advance SOLSQL Performance Tuning	14
4	PL/SQL Programming	16
5	Database security and Transaction Processing	10
Total		**70**

Course Introduction

- Each and every organization like shopping mall, hospital, banking, institutes, industry needs to share huge amount of data in effective manner.

- This course aims to develop skills in students to create, store, modify, manage and extract information from a database.

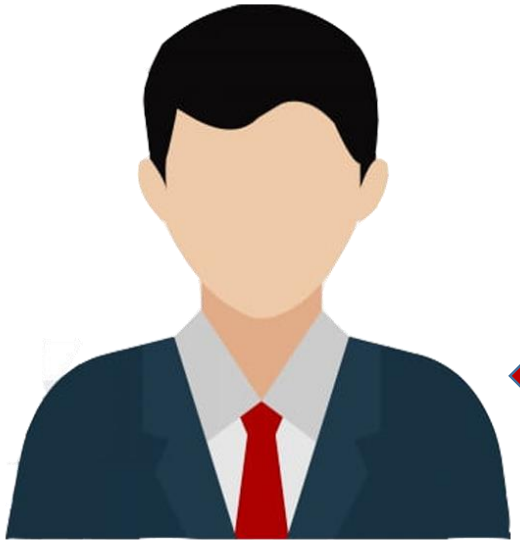
- Database system can be used as a backend for developing database applications.

What is Database Management System (DBMS)

- ✓A Database Management System (DBMS) is software designed to store, retrieve, define, and manage data in a database.

- ✓DBMS software primarily functions as an interface between the end user and the database, simultaneously managing the data, the database engine, and the database schema in order to facilitate the organization and manipulation of data.

How it Works...



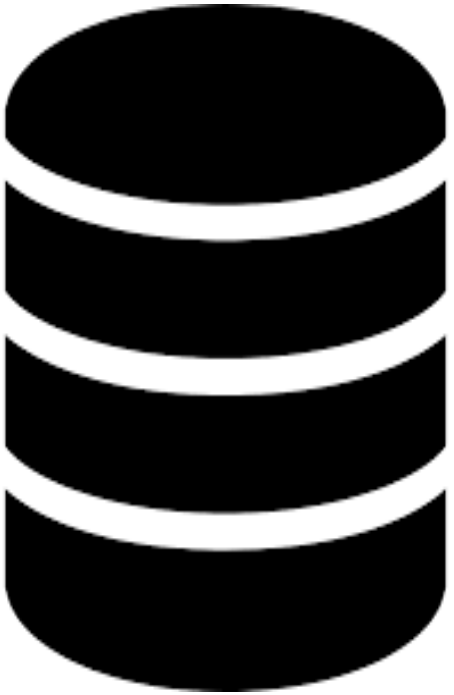
User

User Communicate to DBMS interface for efficient and faster way to fulfilling her requirement's

Software application based on Business Logic

D B M S

Different program to insert , delete , update and query database for information.



Database

UNIT 1

Database System concept

Lecture 01

DEPARTMENT OF COMPUTER TECHNOLOGY

Concept of Data , Database , DBMS

- **Data** is the **information** which has been translated into a form that is more convenient to process or move.

- **Database-** The collection of **related data** is termed as Database which is organized in such a way that it can be **easily retrieved and managed**.

- **A Database Management System (DBMS)** is **System software** which manages the data. It can perform various tasks like **creation , retrieval, insertion, modification and deletion** of data to manage it in a systematic way as per requirement.

Purpose of Database System

- A major **purpose** of a **database system** is to provide users with an abstract view of the data. That is, the **system** hides certain details of how the data are stored and maintained.

File Processing System

- File Processing System is a Computer based system in which all the information is stored in various **computer files.**

- It is useful but as the requirement of **data processing** and the **size of data increases** ,the drawback of systems comes.

Disadvantages of Traditional File Processing System

1.Data Redundancy
2. Data Inconsistency
3. Limited Data Sharing
4. Difficulty in Accessing Data
5. Data Dependence
6. Poor Data Control
7. Problem of Security
8. Concurrency Problems
9. Poor Data Modelling of Real World
10. Data isolation
11. Integrity Problems
12. Atomicity Problem

Advantages of DBMS over File Processing System

1.Controlling Data Redundancy
2. Data Consistency
3. Sharing of Data
4. Data Independence
5. Data Control
6. Security
7. Control over Concurrency
8. Data Modelling of Real World

Advantages of DBMS

- **Controls database redundancy:** It can control data redundancy because it stores all the data in one single database file and that recorded data is placed in the database.
- **Data sharing:** In DBMS, the authorized users of an organization can share the data among multiple users.
- **Easily Maintenance:** It can be easily maintainable due to the centralized nature of the database system.
- **Reduce time:** It reduces development time and maintenance need.
- **Backup:** It provides backup and recovery subsystems which create automatic backup of data from hardware and software failures and restores the data if required.
- **multiple user interface:** It provides different types of user interfaces like graphical user interfaces, application program interfaces

Disadvantages of DBMS

1.Incresed costs
2.Complexity
3.Size
4.Frequent upgrade/replacement cycle
5. Higher impact of a failure
6. Performance

Disadvantages of DBMS

Cost of Hardware and Software: It requires a high speed of data processor and large memory size to run DBMS software.

Size: It occupies a large space of disks and large memory to run them efficiently.

Complexity: Database system creates additional complexity and requirements.

Higher impact of failure: Failure is highly impacted the database because in most of the organization, all the data stored in a single database and if the database is damaged due to electric failure or database corruption then the data may be lost forever.

Application of DBMS

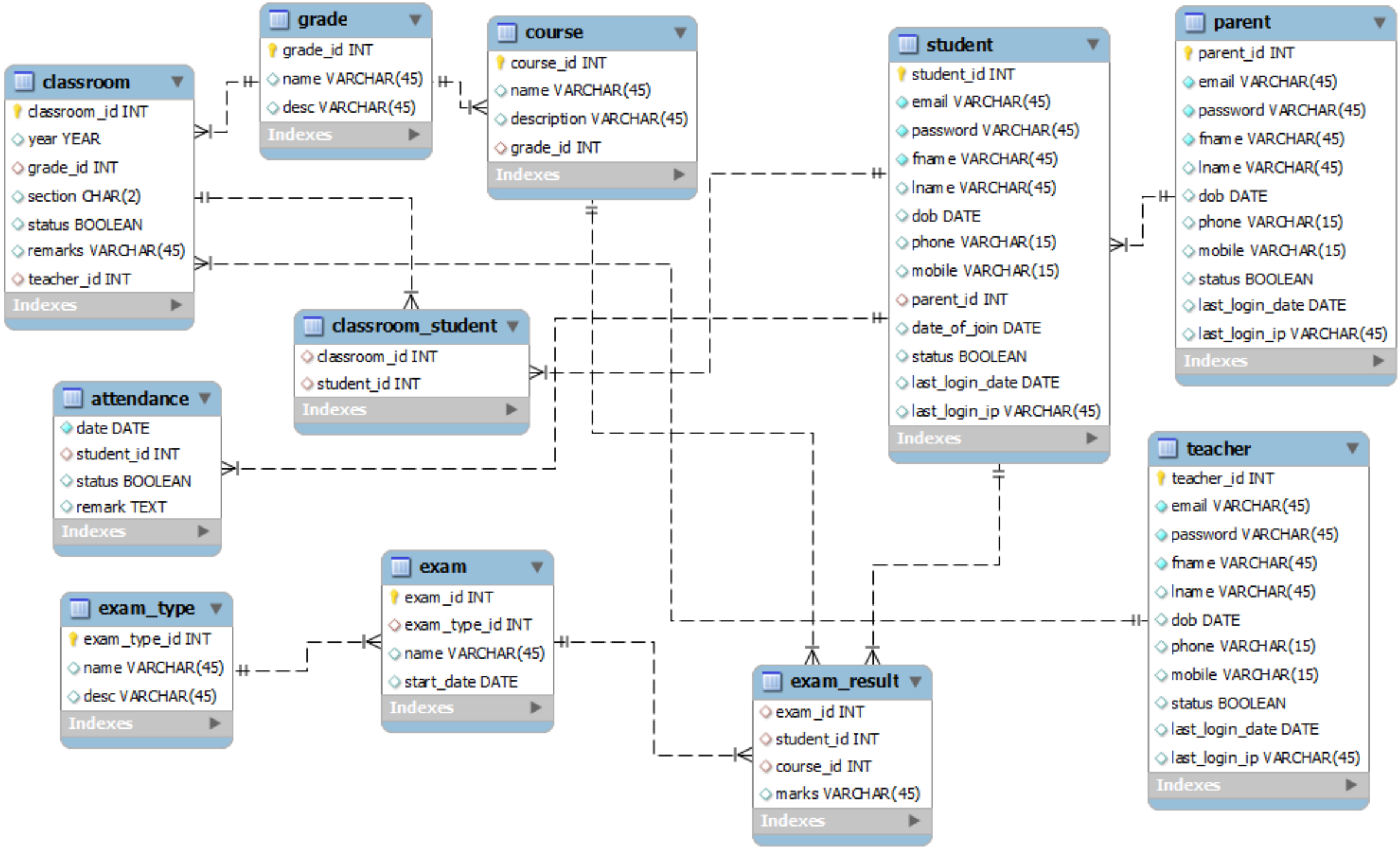
1.Telecom
2. Banking
3. Industry
4. E-Commerce
5. Airlines
6. Education Systems
7. Railway Reservation System
8. Library Management System
9. Social Media Sites

UNIT 1

Database System concept

Lecture 02

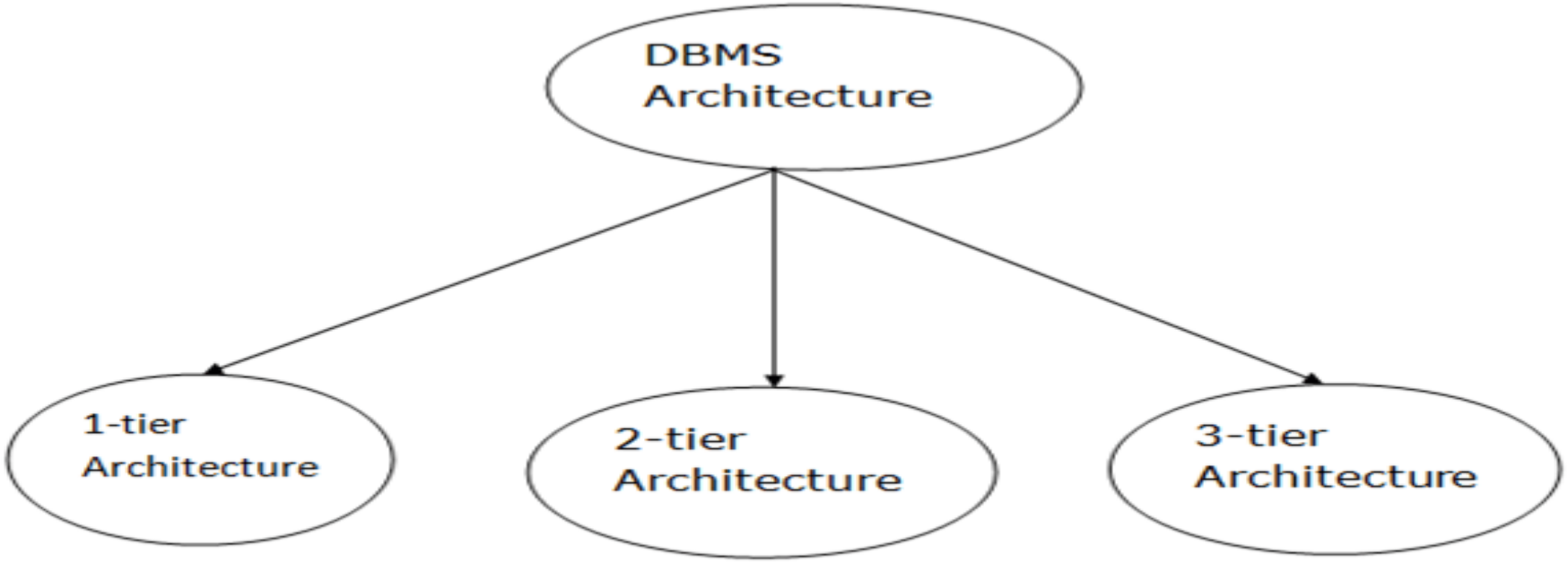
DEPARTMENT OF COMPUTER TECHNOLOGY


grade
grade_id INT
name VARCHAR(45)
desc VARCHAR(45)
Indexes
course
course_id INT
name VARCHAR(45)
description VARCHAR(45)
grade_id INT
Indexes
classroom
classroom_id INT
year YEAR
grade_id INT
section CHAR(2)
status BOOLEAN
remarks VARCHAR(45)
teacher_id INT
Indexes
student
student_id INT
email VARCHAR(45)
password VARCHAR(45)
fname VARCHAR(45)
lname VARCHAR(45)
dob DATE
phone VARCHAR(15)
mobile VARCHAR(15)
parent_id INT
date_of_join DATE
status BOOLEAN
last_login_date DATE
last_login_ip VARCHAR(45)
Indexes
parent
parent_id INT
email VARCHAR(45)
password VARCHAR(45)
fname VARCHAR(45)
lname VARCHAR(45)
dob DATE
phone VARCHAR(15)
mobile VARCHAR(15)
status BOOLEAN
last_login_date DATE
last_login_ip VARCHAR(45)
Indexes
classroom_student
classroom_id INT
student_id INT
Indexes
attendance
date DATE
student_id INT
status BOOLEAN
remark TEXT
Indexes
exam
exam_id INT
exam_type_id INT
name VARCHAR(45)
start_date DATE
Indexes
exam_type
exam_type_id INT
name VARCHAR(45)
desc VARCHAR(45)
Indexes
exam_result
exam_id INT
student_id INT
course_id INT
marks VARCHAR(45)
Indexes
teacher
teacher_id INT
email VARCHAR(45)
password VARCHAR(45)
fname VARCHAR(45)
lname VARCHAR(45)
dob DATE
phone VARCHAR(15)
mobile VARCHAR(15)
status BOOLEAN
last_login_date DATE
last_login_ip VARCHAR(45)
Indexes

Three Level Architecture of Database System

- A data base system is collection of **related data** and **system software** which manages the **data.**
- The data is generally stored in a **detailed and complex manner.**
- It is important to provide an **abstract view** of data to the user.

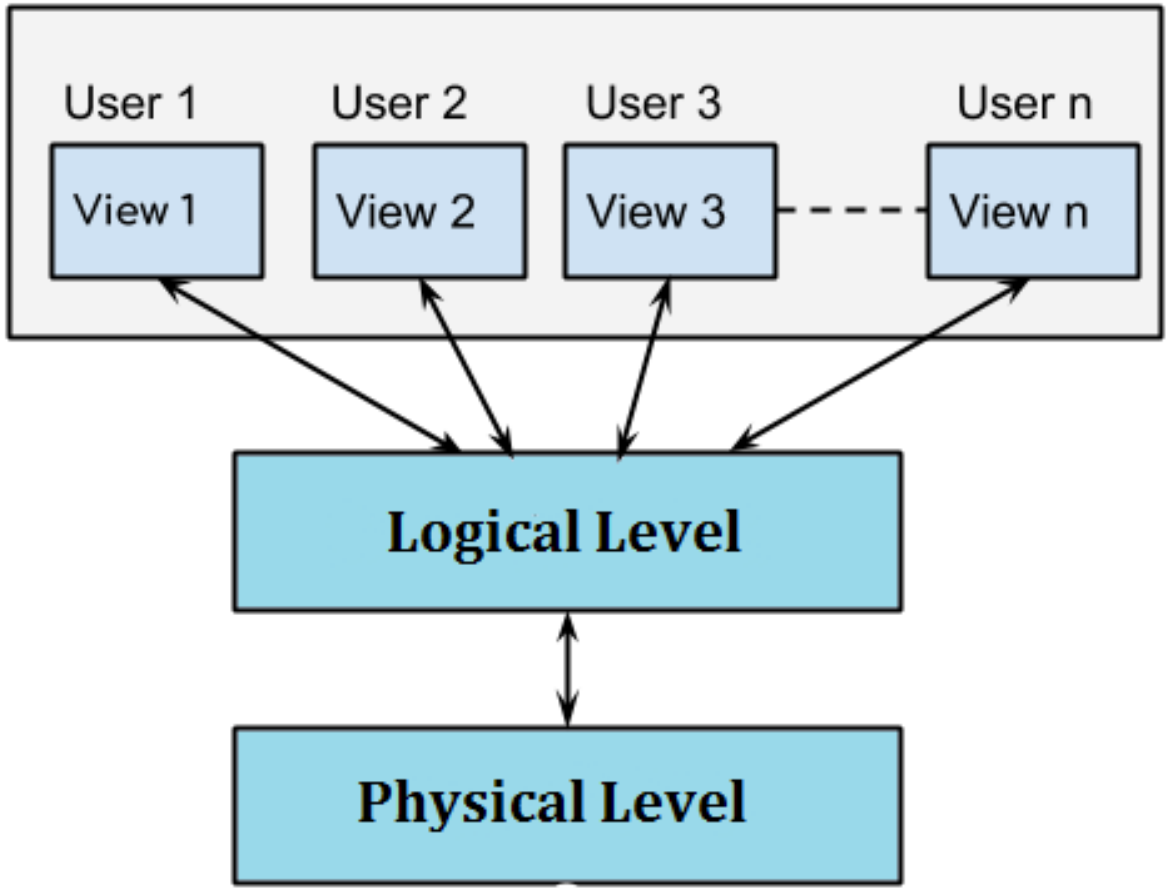
DBMS Architecture



Data Abstraction

- **Extracting the important** data by **ignoring the remaining** irrelevant details is known as **abstraction.**
- This process of **hiding irrelevant** details from user is called **data abstraction.**
- The **complexity of database** can be hiding from user by using different **level of abstraction.**

Different Levels of Data Abstraction



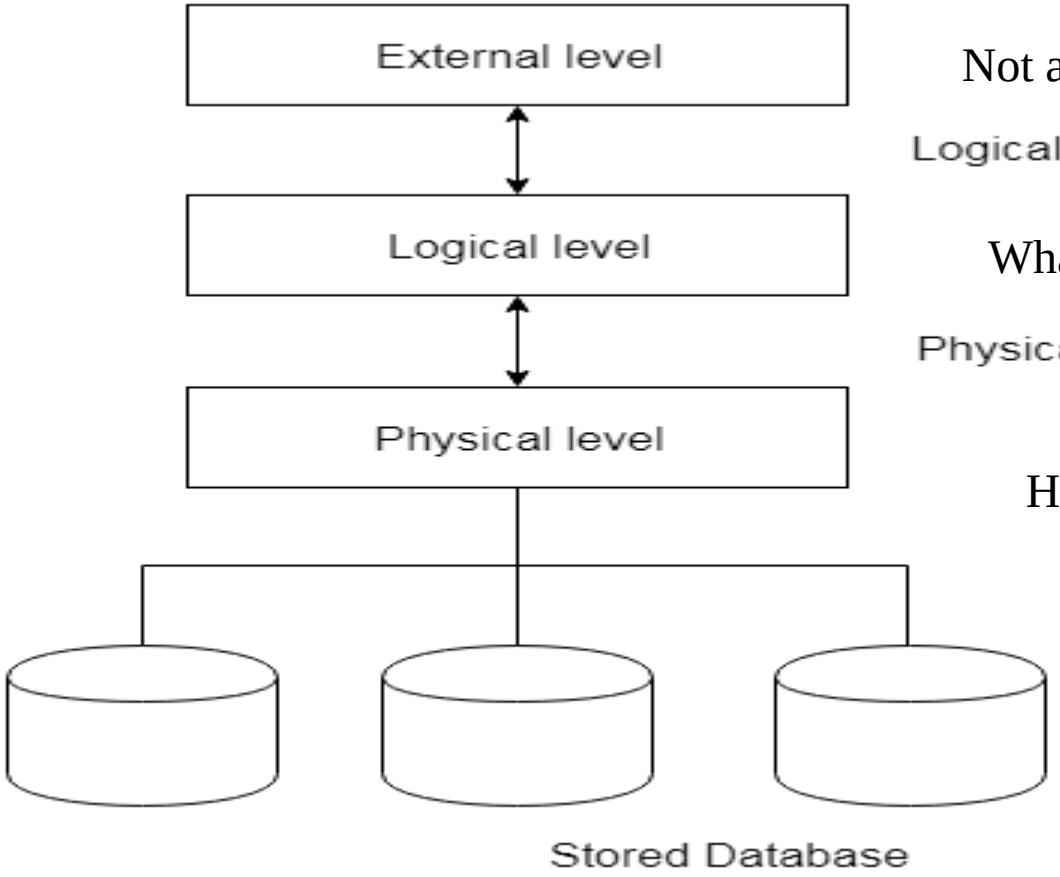
Different Levels of Data Abstraction cont.

1. **Physical Level –**This is the **lowest level** in the three level architecture. The physical level describes **how data is actually stored** in the database. In the lowest level, this data is stored in the **external hard drives** like hard disk, magnetic tapes etc.

2. **Logical Level-** This is the **next higher level** of abstraction which is used to describe **what data the database stores**, and what **relationships exist** in between the data items. Database administrators use the logical level of abstraction to decide **what information to keep** in a database.

3. **View Level-** It is the **highest level** of data abstraction. This level describes the **user interaction with database** system. End user interacts with system with the help of **GUI** and enters the details at the screen at view level. User is not aware of **how the data is stored** and **what data is stored**; such details are hidden from them.

DBMS Abstraction **Example:- IRCTC**



Not aware of storing the user just interact with database.

Logical Data Independence

What data is stored in the database

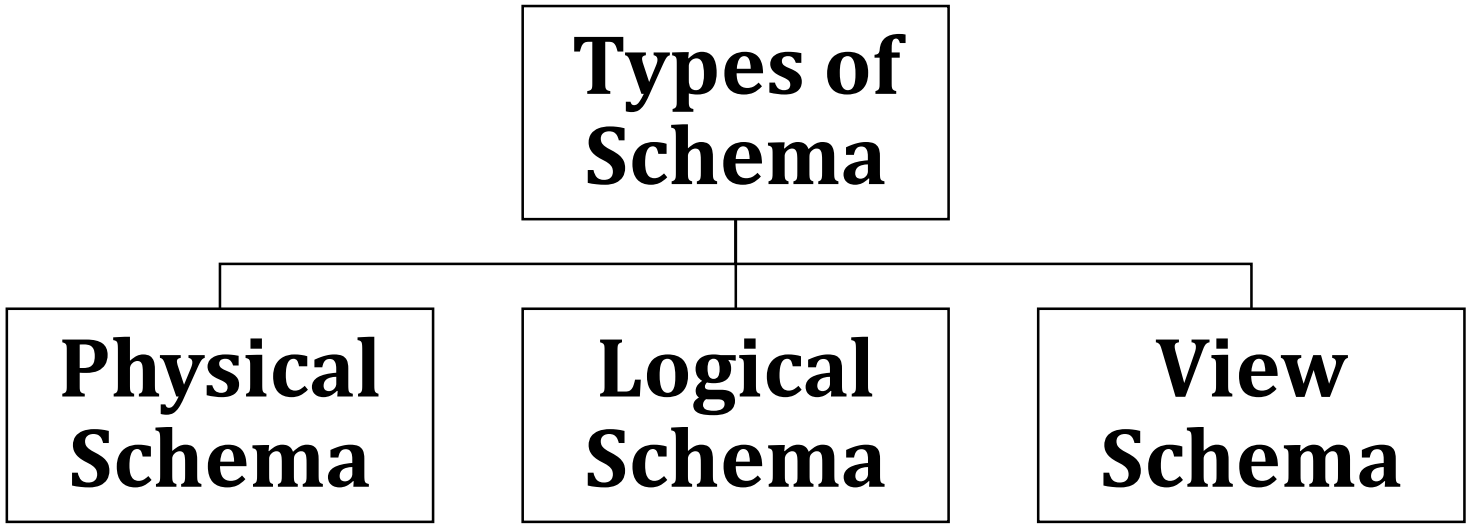
Physical Data Independence

How data is actually stored in the database

Instance and Schema

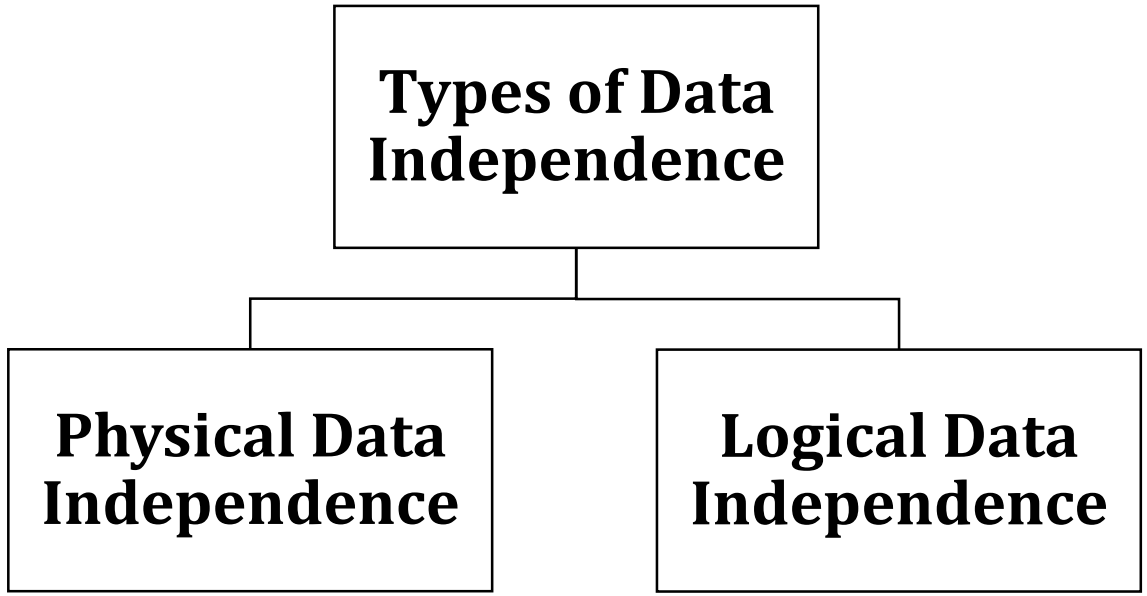
- **Instance-**The data is stored in the database **at particular moment** is called as **instance** of the database.

- **Schema-** The design of a database is called the **schema.**



Data Independence

Data Independence -The ability to modify schema definition in one level without affecting schema definition in the next higher level is called data independence.



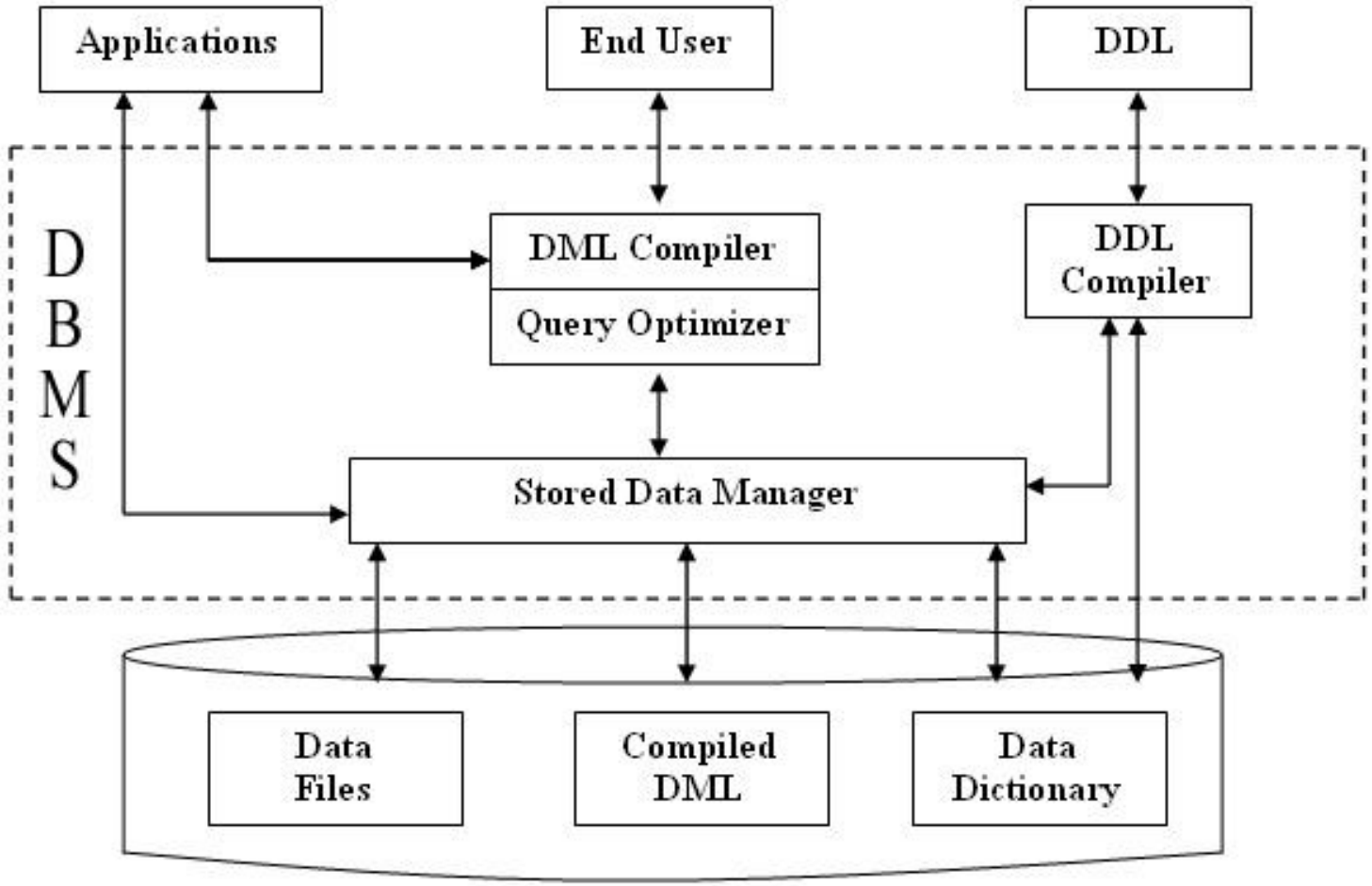
UNIT 1

Database System concept

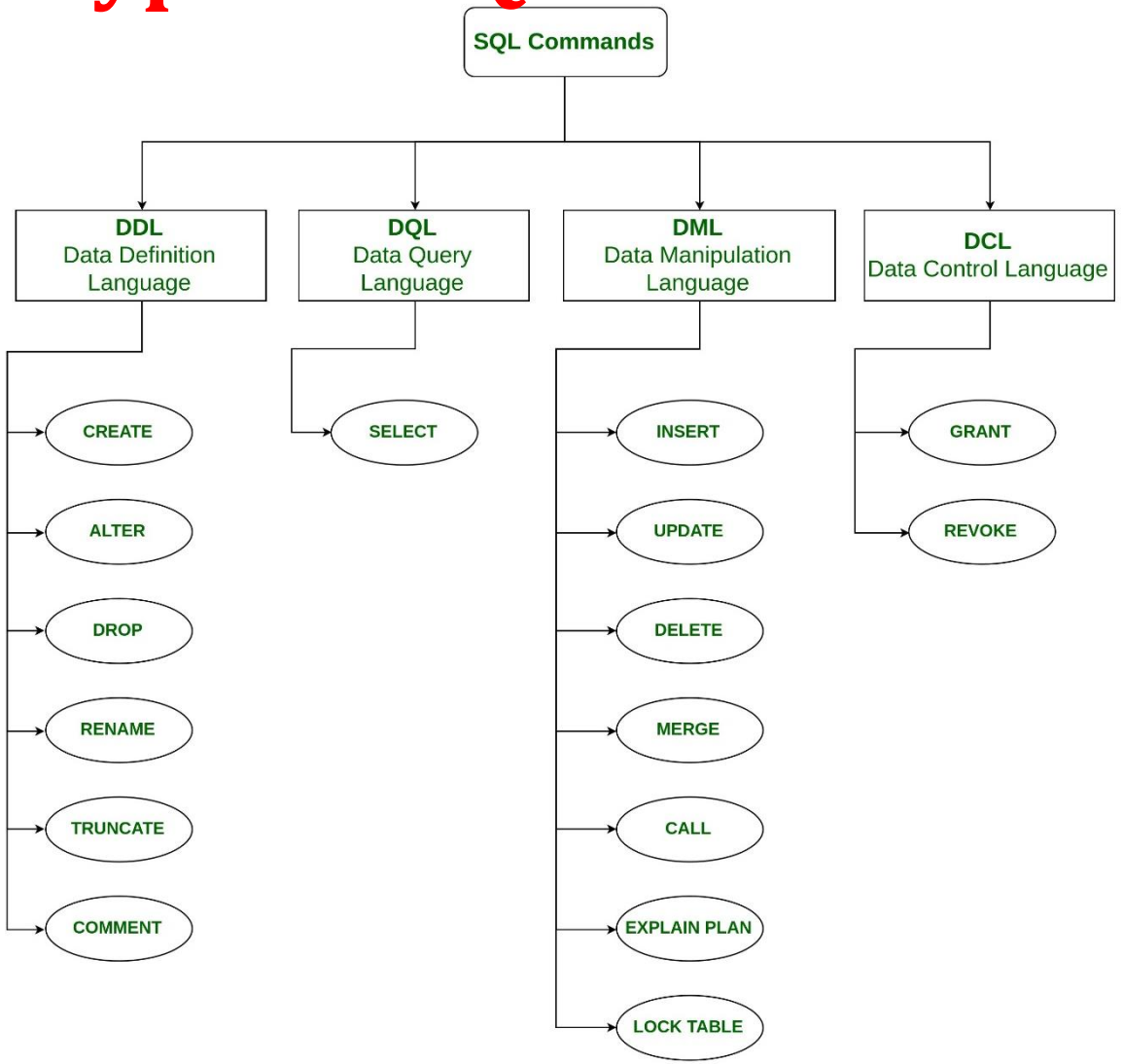
Lecture 03

DEPARTMENT OF COMPUTER TECHNOLOGY

Structure of DBMS



Types of SQL Commands



Structure of DBMS

1. DDL Compiler: It converts the DDL commands into set of table containing **metadata** stored in data dictionary.

2. DML Compiler: It receives the DML commands from application program and converts DML commands into **object code** for understanding of database.

3. Query Optimizer: It optimized the **object code** to execute query in best way and then send to store data manager.

4. Storage Data Manager: It is a program module which is responsible for storing, retrieving and updating data in the database. It receives the request from query optimizer to machine understandable form.it makes the actual request inside the database.

Structure of DBMS

1. **Data Files:** It stores the **database** itself.

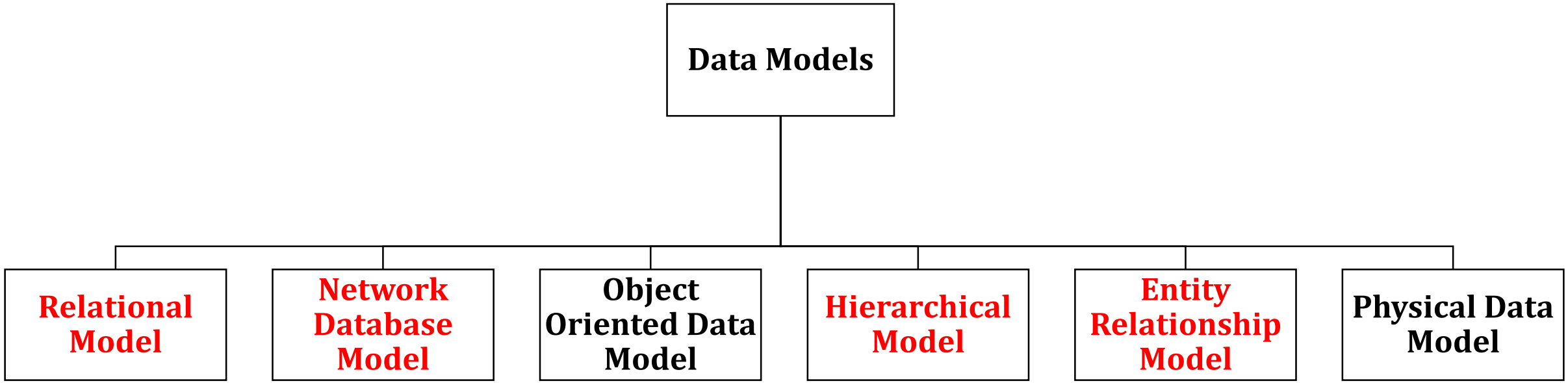
2. Compiled DML: The DML Compiler converts the high level Queries into low level file access commands known as **Compiled DML.**

3. Data Dictionary: It stores **metadata**, i.e. file, storage , attributes , access path , etc.

Data Models

- The process of analysis of data object and their relationships to other data objects is known as **data modeling.**
- It is the **conceptual representation** of data in database.
- It is the first step in **database designing**.
- Data models define **how data is connected to each other** and **how they are processed** and **stored inside the system**.
- A data model provides a way to describe the **design of a database** at the **physical, logical and view levels.**

Types of Data Models



1.Relational Model

1. Record Based Logical Model/Relational Model-

- Developed by **E.F.Codd.**
- Tables are used and also known as **relations.**
- Record are known as **tuples** and fields are known as **attributes.**
- Every record must have a **unique identification or key.**
- **Advantages-**
  1. Support SQL
  2. Flexible

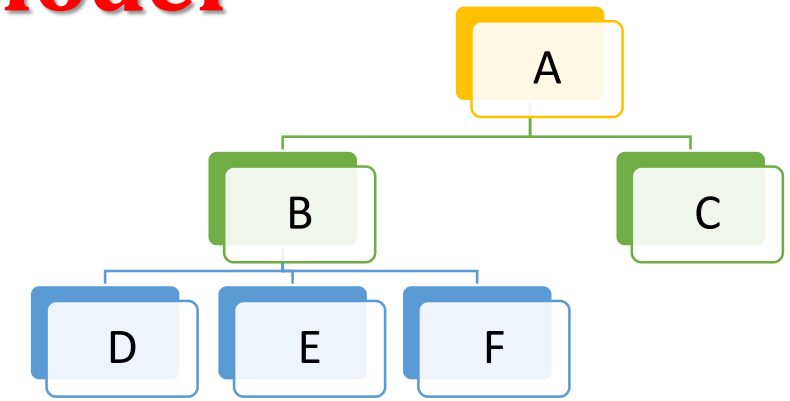
Key

Tuple →

Stud_ID	Stud_Name	DOB
101	Prajakta	03/03/1995
102	Rakesh	13/01/1996
103	Rahul	16/08/1995

2.Hierarchical Model

2. Hierarchical Model



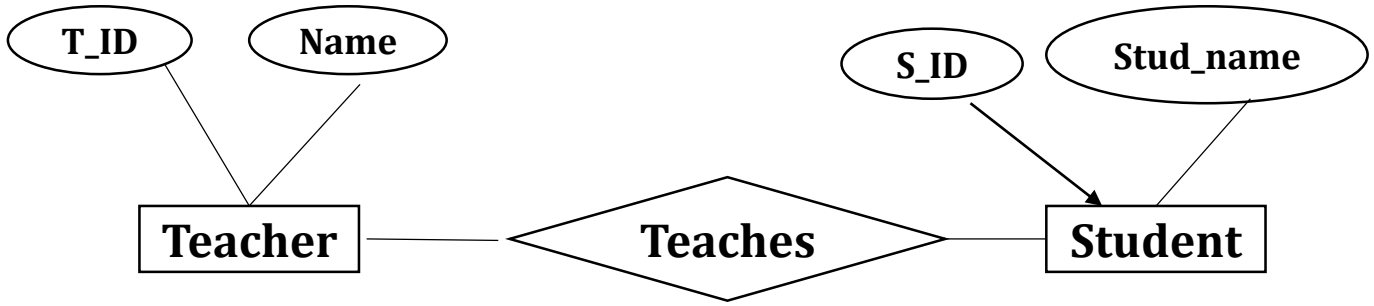
- Organized into a **tree structure.**
- **Parent-child** relationship.
- Data is stored in the form of **Record.**
- Record is collection of **fields** and it contains only **one value.**
- One **parent can have many child nodes** but **one child node can have only one parent node.**
- **Advantages-**
    1. Simple to Understand.
    2. Database Integrity 3.Efficient

3.Network Database Model

3.Network Database Model

- Extended type of Hierarchical data Model but any **child can have multiple parent.**
- **No need of Parent-child** relationship.
- Allows multiple **Records linked in same file.**
- **Advantages-**

1. Design is Simple.
2. Capability to handle various relationship.
3. Easy to access.

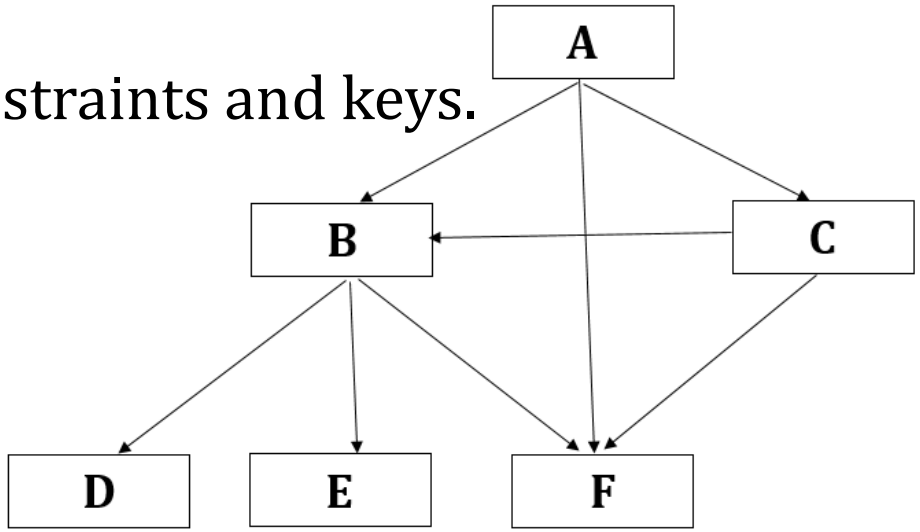


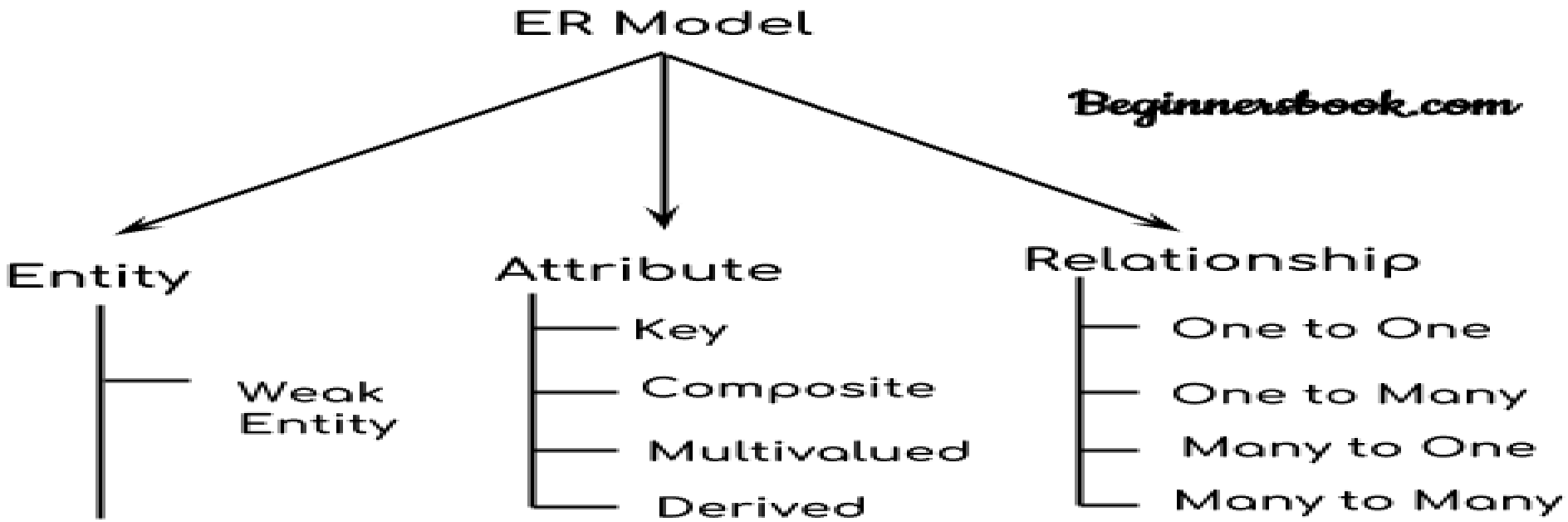
4.Model/E-R Model

4.Entity Relationship Model/E-R Model

- Represents overall structure of database.
- **Design technique of database**.
- Real world object can be easily mapped with entities of E-R model.
- Diagrams are used which known as Entity relationship digram,ER digrams or ERD's.
- Concept of ER model-Entity,attribute,Relationship,constraints and keys.
- **Advantages-**

1. Design is Simple.
2. Effective representation
3. Connected with Relational Model





Components of ER Diagram

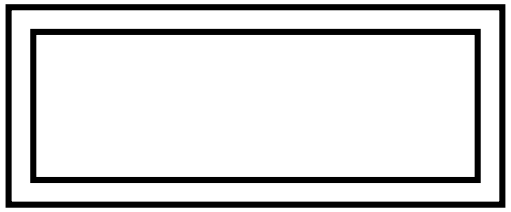
Entity and Entity Set

- An entity is a thing that exists either **physically or logically.**
- An entity is nothing but a thing having **its own properties.**
- These properties helps to **differentiate** the **object from other objects**.
- An **entity set** is a set of entities which share the **same properties.**

Types of Entity

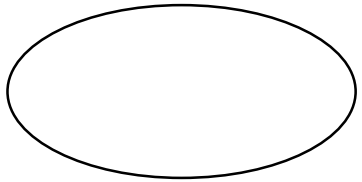
1.Strong Entity or Regular Entity- If an entity having it's **own key attribute** specified then it is a strong entity. Key attribute is used to identify that entity uniquely among set of entities in entity-set. Strong entity is denoted by a single rectangle.

2.Weak Entity- The entity which **does not have any key attribute** is known as weak entity. The weak entity has a partial discriminator key. Weak entity depends on the strong entity for its existence. Weak entity is denoted with the double rectangle.

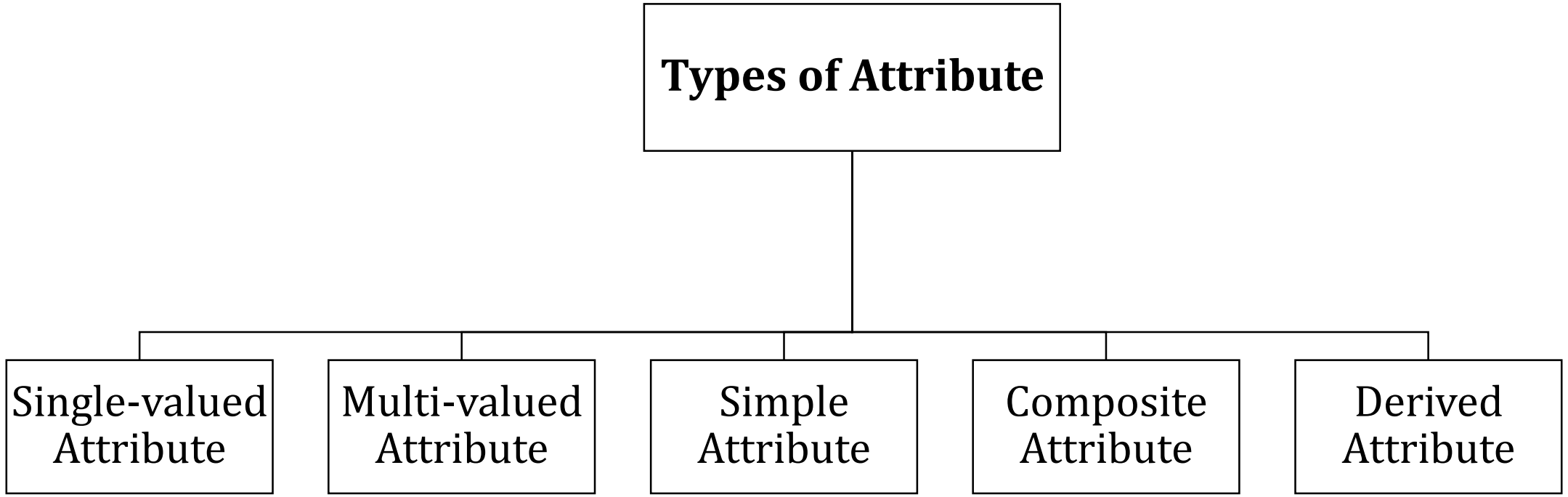


Attribute

- An attribute is a **characteristic** of an entity.
- Entities are represented by means of their attributes.
- All attributes have their **own specific values.**
- Attribute is denoted by a Ellipse.

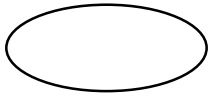


Types of Attribute

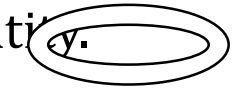


Types of Attribute

1.Single- valued Attribute- A Single-valued attribute is the attribute which **can hold single value** for the single entity.



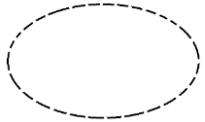
2. Multi-valued Attribute -A multi-valued attribute is the attribute which **can hold multiple values** for the single entity.



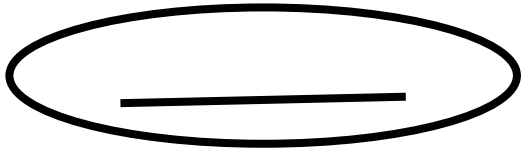
3. Simple Attribute- An attribute whose value **cannot be further divided** is known as simple attribute. That means it is atomic in nature.

4. Composite Attribute- The composite attributes are the attributes which **can be further divided** into sub parts. These sub parts represent the basic entities with their independent meaning.

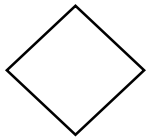
5. Derived Attribute -The attribute which is **not physically exist in database**, but its value can be calculated from the other present attributes is known as derived attribute.

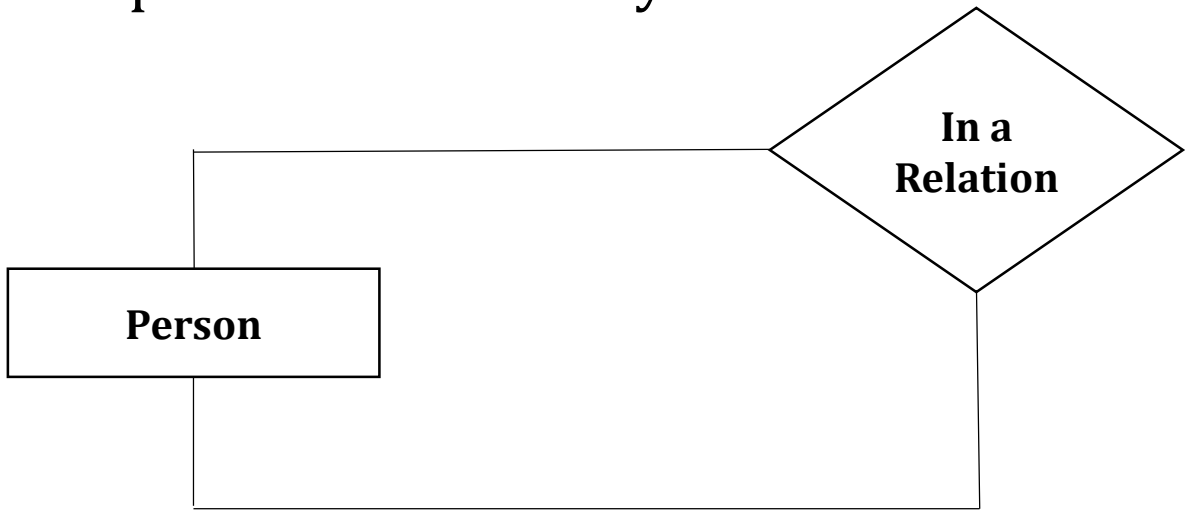


6.Key Attribute-The Key attribute is used to denote the **property that uniquely identifies** an entity and which is mapped to the **Primary Key** field in a database.



Relationship

- The association between two different entities is called as **relationship.** It is denoted by Diamond. 
- Line is used to links attributes to entity and entity set to relationship sets. 
- **A unary relationship** exists when there is **relation single entity.** A unary relationship is also known as recursive relationship in which an entity relates with itself.



Mapping Cardinality in E-R Diagram

- Mapping Cardinality in E-R Diagram

1. One to One
2. One to Many
3. Many to One
4. Many to Many

Component of ER Model

Rectangle: Represents Entity sets.
Ellipses: Attributes
Diamonds: Relationship Set
Lines: They link attributes to Entity Sets and Entity sets to Relationship Set
Double Ellipses: Multivalued Attributes
Dashed Ellipses: Derived Attributes
Double Rectangles: Weak Entity Sets
Double Lines: Total participation of an entity in a relationship set