Chapter-2 Data Model (4hrs) (7-12)

2.1 Logical, Physical and Conceptual Model.
2.2 ER Model
2.3 Relation with UML class diagram
2.4 Alternate data Model (Network Data Model, Hrerarc
Data Model

Data Model

- Data models define how the logical structure of database is modeled.
- They are useful tools for database ~~desing~~ design. Data model is a set of concepts to describe the structure of a data base and certain constraints that the database should obey.
- It defines how data is connected to each other and how they are ~~proces~~ processed and stored in the system.

- Stages of data models are

1. Conceptual Data Model
2. Logical Data Model
3. Physical Data Model.

Chapter-2

Data Model (4hrs) (7-12)

2.1 Logical, Physical and Conceptual Model.
2.2 ER Model
2.3 Relation with UML class diagram
2.4 Alternate data Model (Network Data Model, Hrerarchial Data Model

Data Model

- Data models define how the logical structure of database is modeled.
- They are useful tools for database design. Data model is a set of concepts to describe the structure of a data base and certain constraints that the database should obey.
- It defines how data is connected to each other and how they are processed and stored inside the system.

- Stages of data models are

1. Conceptual Data Model
2. Logical Data Model
3. Physical Data Model.

GOOD MORNING
PAGE NO. :
DATE : / /

Data Model

Features	Conceptual	Logical	Physical
Entity name	Yes	Yes	
Entity Relationship	Yes	Yes	
Attributes		Yes	
Primary key		Yes	Yes
foreign keys		Yes	Yes
Table names			Yes
Column Names			Yes
Column Data Types			Yes

College Scenario

entities	entity relationship
student	student read subject
subject	
faculty	faculty teaches subject

Attributes

student	subject	faculty
name	subject code	
rollno	subject name	
department		
subject		
phone number		

primary key & foreign key : rollno, subject code

table name student_info table name student_info tablename fac

column_names
rollno | name | department | subject_code | phone_no |

Column_names
subject_code subject_name subject_marks

column_data_type	
roll_no	int
name	char
department	char
subject_code	int
phone_no	double

SQL query

Bus Ticket Reservation system

Bus — passenger buys ticket
ticket — ~~tickets are assigned~~
passenger — tickets are assigned to bus

Ticket

ticket_no	p_id	bus_id	~~ticketed~~ ticket_cost
502	101	6533	600

bus

bus_id	bus_name	bus_capacity	bus_destination
7855	XYZ travel	33	Pokhara

Passenger

p_id	p_name	p_email	p_mobile	p_age	p_addre
101	abc	abc@gmail.com	98653254	25	ktm

Virtual table

ticket_no	p_name	bus_name	ticket_cost	passenger

ticket_no	p_name	p_mobile	bus_staff

1. Conceptual Model

- It identifies highest level relationship among different entities.
- It is highly abstract ie We don't have much details
- It can be understood easily by both technical and non technical user
- Only 'entities' are visible. No attributes and primary key specified.
- Relationship among entities are abstract too. (That is even relationship details are hidden).
- It can be easily enhanced
- Not any software tools are required to define data models. (since, conceptual data model can be written on a piece of paper or white board, there is not any requirement of software tools.)

Conceptual Data Model.

Entities: Time
Product
Sales & store

- (Note: from the figure, entities Time, Product and Store have direct relationship with sales entity, due to which to of information can be obtained by looking at the conceptual data model. And since it is not a digital document, it can be easily enhanced).

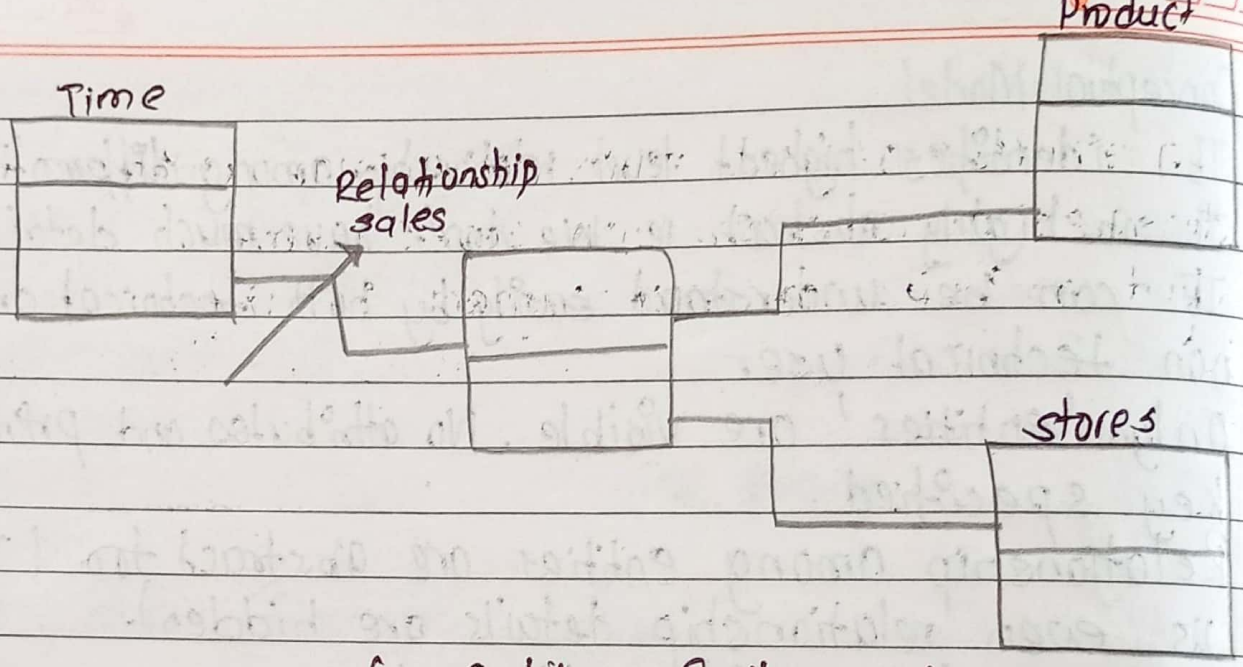


fig: Entities & their relationship.

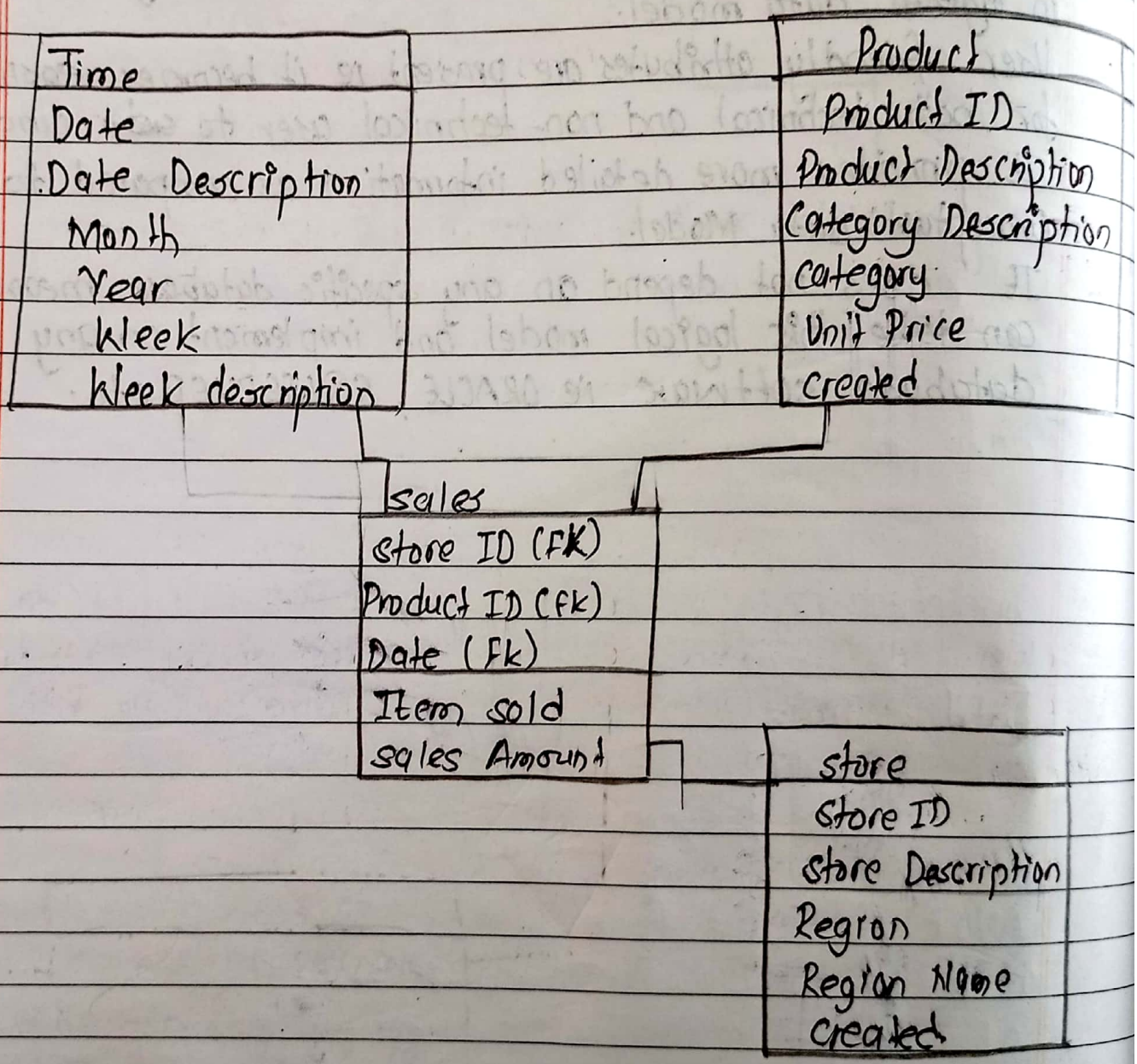
Logical Data Model

- Attributes of each entity are present, which are further identified as keyed and non-keyed attributes.
- keyed attributes define uniqueness of entity.
- Primary key - foreign key relationship are clearly defined in logical data model.
- User friendly attributes are present ie it becomes easier for both technical and non-technical user to understa
- It provides more detailed information as compared to Conceptual data Model.
- It does not depend on any specific database means yo can take this logical model and implement on any database software ie ORACLE, SQLSERVER etc.

GOOD MORNING
PAGE NO. :
DATE : __/__/____

Logical Data Mode

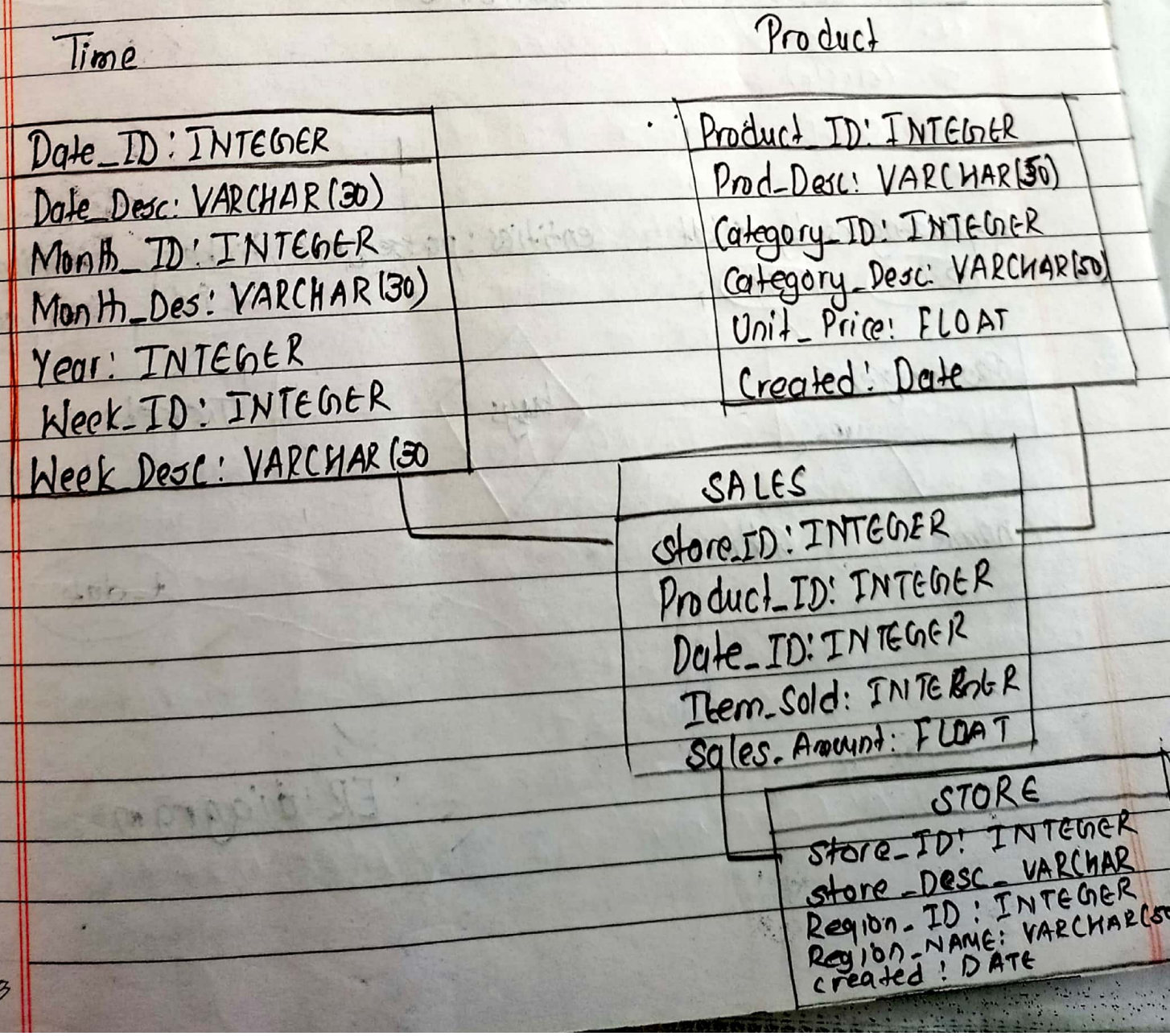
- In this example Date, Product ID, store ID are primary key for Time, product and store entity respectively and can also be foreign key for sales entity.



GOOD MORNING
PAGE NO.:
DATE: / /

Physical Data Model

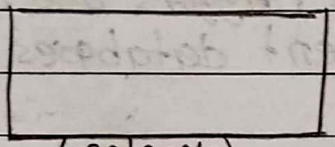
- In this model, entities referred to as tables and attributes are referred to as columns.
- It provides database compatible table name and database specific data types
- It is difficult for user to understand
- It requires more effort to be enhanced in comparison to logical data model
- It includes indexes, constraints, triggers and DB Objects
- It is difficult to port to a different databases once design is finalized.



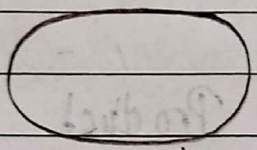
Data Model Standard

1976 peter chen
ER model
entity relationship
model.

(diamond) Relationship

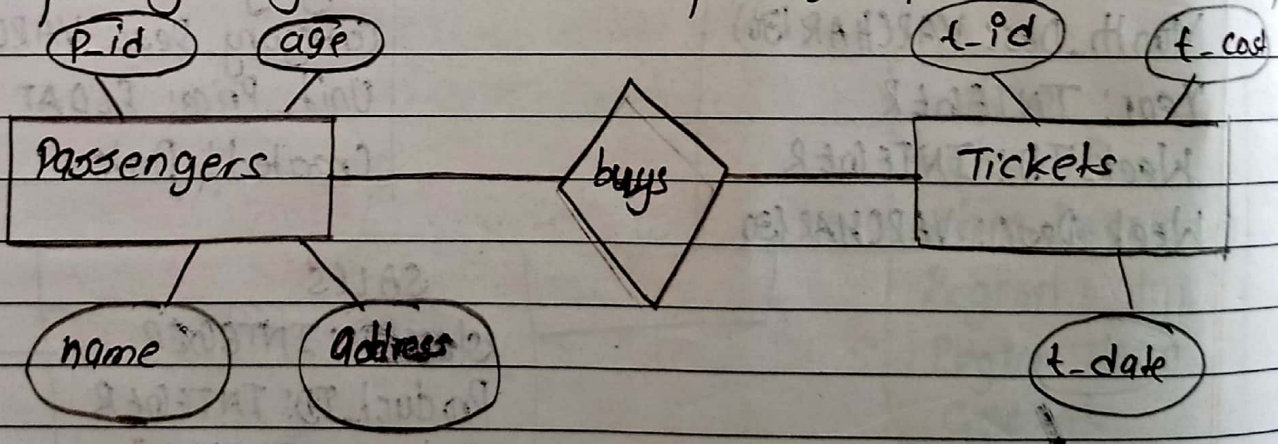


(rectangle) entity



(circle) attributes.

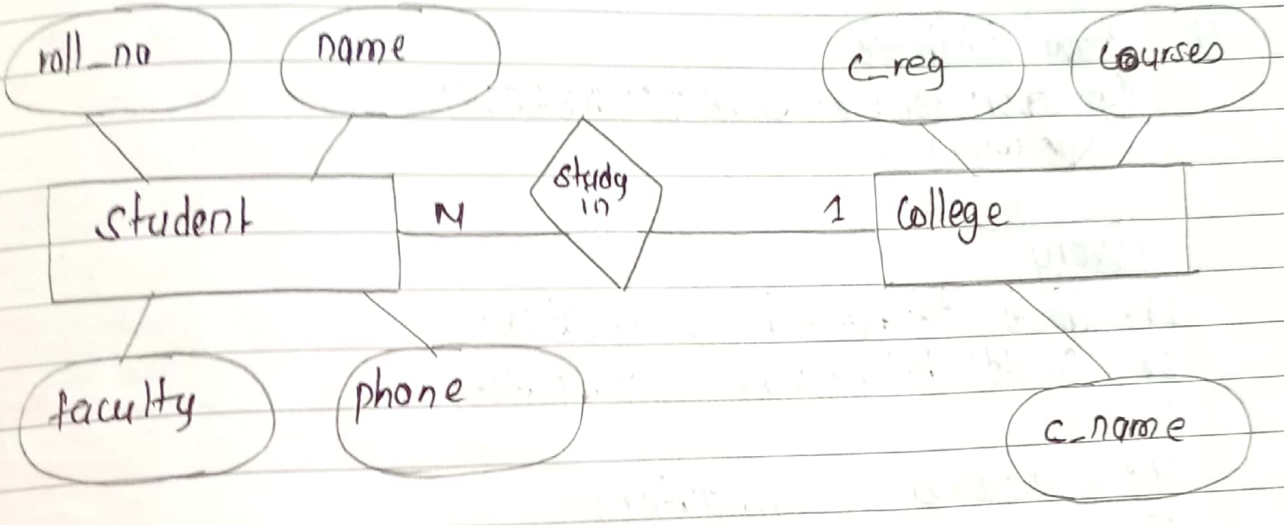
Ex-passengers by ticket entities: passengers, tickets, relationship: b



ER diagram

entities: college, student

college teaches student
student study in college



ER diagram

SQL (structure Query Language): 1970s; 1974 popular language

LAB 1: DDL & DML statements

1. Create Database:

Syntax: Create database (database_name;

Example:

CREATE DATABASE: test_db;
CREATE DATABASE bce_computer;

test database = test_ database

2. Use Database:

Syntax: USE database_name;

Example:

3. Drop Database
Syntax: DROP DATABASE database_name;
Example:
DROP DATABASE bce_computer;

4. Show database
Syntax or example: SHOW DATABASES;

LAB1
Query
1. Create database bce_Computer;
2. Create database beit_computer;
3
4. use bce_computer;
5
6. drop database beit_Computer;
7
8. Show databases;
9
10. create table student(roll_no int not null, student_name varchar(25), address varchar(20), primary key (roll_no);
11. select * from student;

5. Create table.
Syntax: CREATE TABLE table_name (column1 datatype, column2 datatype, ... ColumnN datatype. PRIMARY KEY (ColumnN);

Example: CREATE TABLE student(roll_no int not null, name varchar(25), faculty varchar(20), address varchar(20), sgpa decimal(3,2), PRIMARY KEY(roll_no));

Table name: student

roll_no	name	faculty	address	SGPA

6. Select statement
   Syntax SELECT column_name or * from table_name;
   *: all
   Example:
   select * from student;
   select roll_no, SGPA from student;

7. Drop table
   Syntax: DROP TABLE table_name;
   Example
   Drop table student;

8. Insert data into table
   Syntax: INSERT INTO table_name (column1, column2, ... column N)
   values (value 1, value 2 ..... value N);
   Example:
   Insert into student (roll_no, student_name, faculty, address, SGPA)
   values (17445, 'abcd', 'BCE', 'ktm', 3.56);
   OR
   Insert into student values (17445, 'abcd', 'BCE', 'ktm', 3.56);

10.

1. Drop table student
2. CREATE TABLE student (roll_no int not null, name varchar(25), faculty varchar(20), address varchar(20), SGPA decimal(3,2), PRIMARY KEY (roll_no));

3. Select * from student;

Code

1. Create database bce_database;
Use bce_database
DROP database bce_database;
CREATE TABLE student (roll_no int not null, student_name varchar(25), faculty varchar(20), address varchar(20), sgpa decimal(3,2), PRIMARY KEY (roll_no));
Drop table student;
select * from student;
select roll_no, sgpa from student
insert into student values (17256, 'abc', 'BCE', 'KTM', 3.22);
insert into student values (17257, null, null, null, null);
insert into student values (17258, 'abc', 'KTM', 'BCE', 3.22);

Question 1: Write SQL query for the following cases.

1. Create database named employee
2. Use that database to perform other operation that follow
3. Create a relation name employee_info with attributes emp_id as int emp_name as varchar, salary as decimal ⇒ age as int (emp_id is primary key ⇒ not null)
4. Insert at least five data of your choice into the relation employee_info
5. Display only emp_name ⇒ salary from the relation employee_info.

1 Create database Employee,

2. Use Employee;

3 Create table employee_info (emp_id int not null, emp_name varchar(25), salary decimal(10,2), age int, Primary key (emp_id));

4. Insert into employee_info values (111, 'abcd', 12536.5, 26);
Insert into employee_info values (112, 'abcd', 12536.5, 26);
Insert into employee_info values (113, '
Insert into employee_info values (114,
Insert into employee_info value (115,

5. Select emp_name, salary. from employee_info;

9. Add New column in an existing table:
Syntax: ALTER TABLE table_name ADD column_name datatype;

Table name: student

roll_no	student_name	faculty	address	SGPA	Phone no

Example: ALTER TABLE student ADD phone_no varchar(30);

10. Drop Column in an existing table:
Syntax: ALTER TABLE table_name DROP COLUMN column_name;
Example:
ALTER TABLE student DROP COLUMN phone_no;

Output

roll_no	student_name	faculty	address	SGPA

11. To change a Datatype of a column:

syntax:

ALTER TABLE table_name MODIFY COLUMN Column_name data type;

Example:

ALTER TABLE student MODIFY COLUMN phone_no varchar(10);

Insert into student values (1726D, 'abc', 'BCE', 'KTM', 3.22, '9845696...

ER MODEL

- In this database model, relationships are created by dividing Object of interest into entity and its characteristics into attributes.
- Different entities are released using relationships.
- E-R Models are defined to represent the relationships into pictorial form to make it easier for different stakeholders to understand.
- This model is good to design a database, which can then be turned into tables in relational model.
- While formulating real world scenario into the database model, ER model creates entity set, relationship set, general attributes and constraints.
- An Entity-relationship model (ER model) describes the structures of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram)
- An ER model is a design or blueprint of a database that can later be implemented as a database.

ER Model

- first published in 1976 by ~~peter~~ Peter Chen
- The main components of E-R model are: entity set and relationship set.
- In the following diagram we have two entities student and College and their relationship.
- The relationship between student and college is many to one as a college can have many students however a student cannot study in multiple colleges at the same time.
- students entity has attributes such as roll_no, name, faculty and phone entity has attributes such as C_reg, courses & C_name.

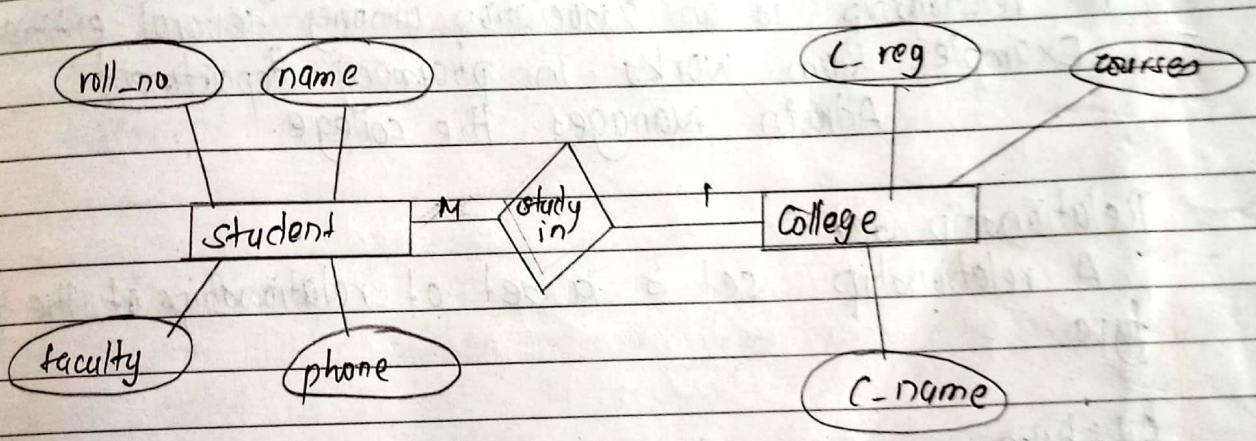


fig. ER diagram.

Entity & Entity Set.

Entity

- An entity is a "thing" or "object" in the real-world that is distinguished from all other objects.
- for example a specific person, company, a particular events etc.

GOOD MORNING
PAGE NO.:
DATE: __/__/__

Entity set
- An entity set is a set of entities of the same types that share the same properties
- It consist of same set of attribute.
- Each entity set has a key.
- Each attribute has a column
- Example: set of all department, student, admin. etc

Relationship & Relationship Set

Relationship
- The connection among two or more entity set.
- A relationship is an association among several entities
- Example: Ram works in pharmacy department
  Admin Manages the college.

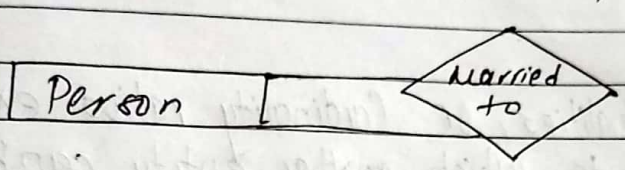
Relationship set.
- A relationship set is a set of relationships of the same type

Attributes
- Attributes are descriptive property ~~possed~~ possessed by each member of entity set.
- Degree of a relationship set: The number of different entity sets participating in a relationship set is called as degree of a relationship set.

GOOD MORNING
PAGE NO.:
DATE: __/__/____

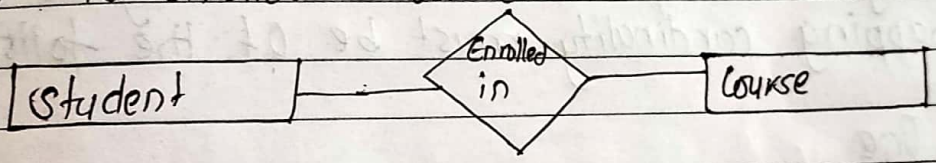
1. Unary Relationship

When there is only ONE entity set participating in a relationship is called as ~~a~~ unary relationship. for example one person is married to only one person.



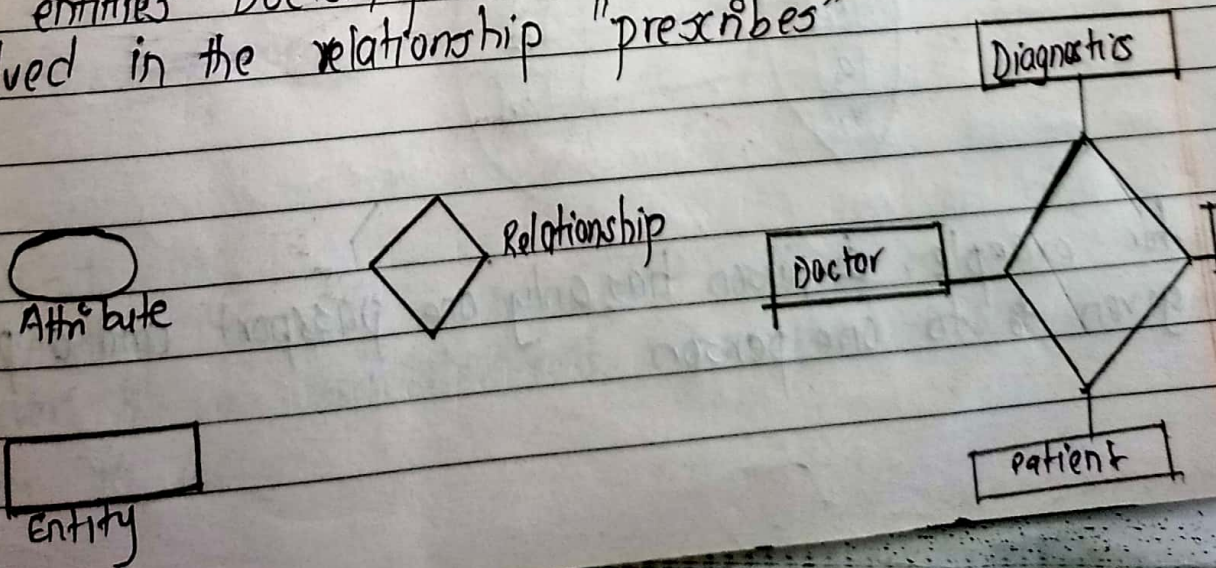
2. Binary Relationship

When there are Two entities set participating in a relationship is called as binary relationship. for example, student is enrolled in course.



3. n-ary Relationship

- When there are n entities set participating in a relation the relationship is called as n-ary relationship
- In the real world, a patient goes to a doctor and doctor prescribes the medicine and diagnosis to the patient, four entities Doctor, patient and medicine, diagnostics are involved in the relationship "prescribes"



GOOD MORNING
PAGE NO.:
DATE: __/__/____

Constraints in Data Models

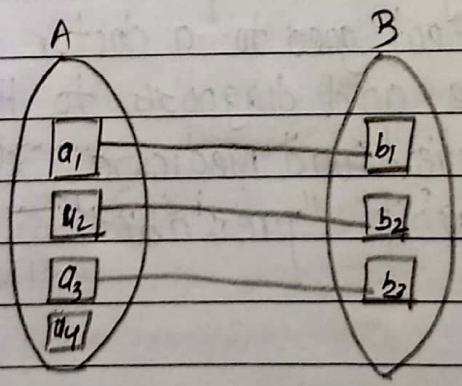
- An ER enterprise schema may define constraints to which the contents of a database must confirm. We have mapping cardinalities and participation cardinalities.

- Mapping Cardinalities, or Cardinality ratios express the number of entities to which another entity can be associated via a relationship set. Or, the number of times an entity of an entity set participates in a relationship set is known as cardinality.

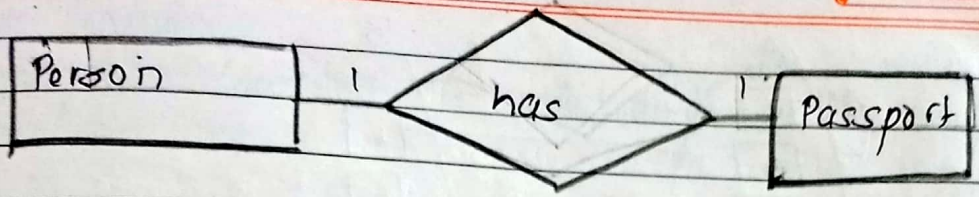
- for Binary relationship set R between entity set A and B, the mapping cardinality must be of the following.

- One to One

An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.

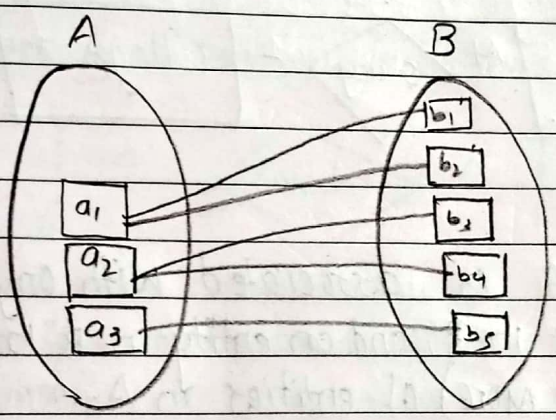


for example, a person has only one passport and a passport is given to one person

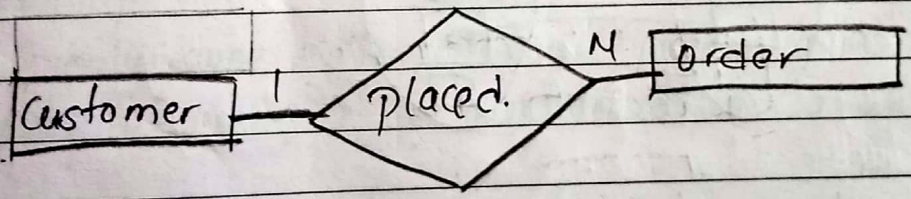


One to many

- An entity in A is associated with any number (zero or more) of entities in B. An entity in B, however can be associated with at most one entity in A.

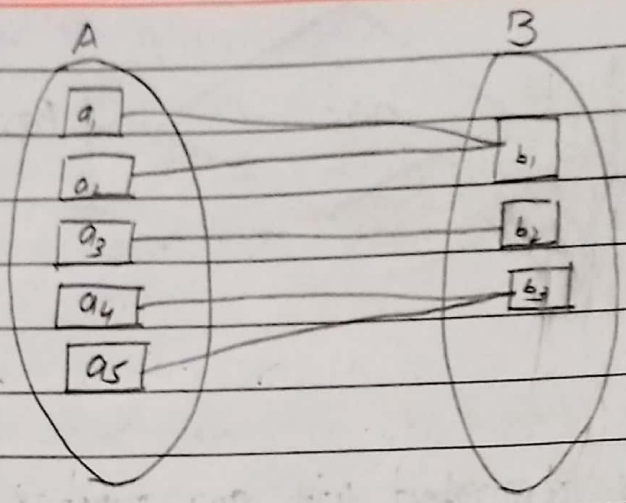


for example: a customer can place orders but a order cannot be placed by many customers.

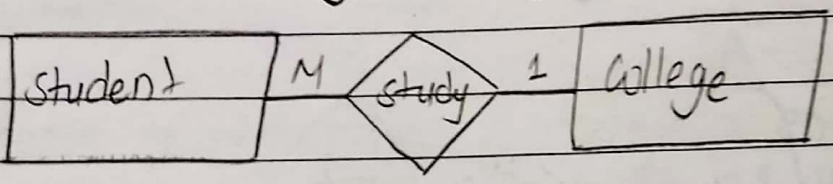


Many-to-One

- An entity in A is associated with at most one entity in B, however can be associated with any number (zero or more) of entities in A.

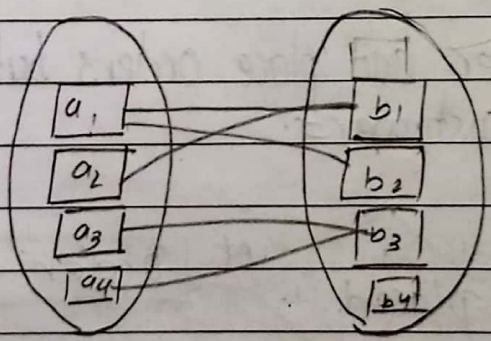


for example - many students can study in a single college but a student cannot study in many colleges at the same time.

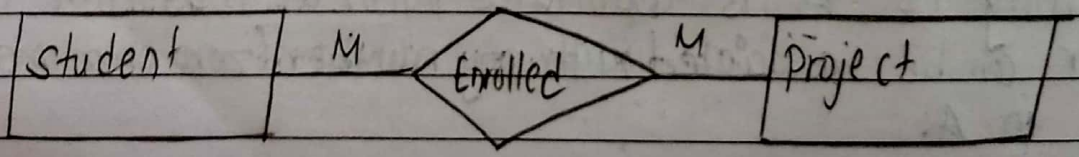


Many to Many

- An entity in A is associated with any number (zero or more) of entities in B, and an entity in B is associated with any number (zero or more) of entities in A.



- for example, a student can be assigned to many project can be assigned to many students.

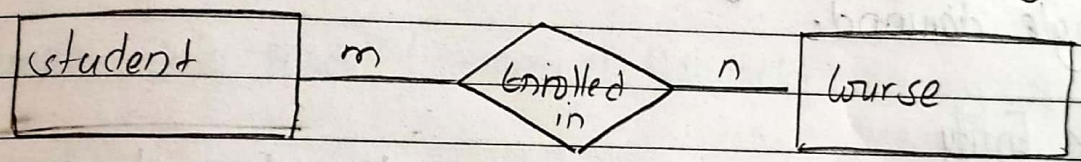


GOOD MORNING
PAGE NO.:
DATE: / /

Participation Constraints:

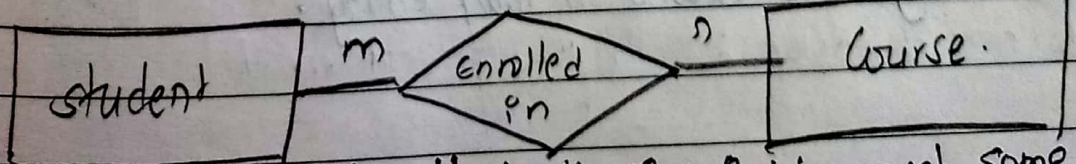
participation Constraint is applied on the entity participating in the relationship set.

1. Total participation
- Each entity in the entity set must participate in the relationship. That is, the participation of an entity set E in a relationship set R is said to be total if every in E participates in at least one relationship in R. Also called mandatory participation.
- If each student must enroll in a course, the participation of student will be total.
- Total participation is shown by double line in ER diagram.



Example: It specifies that each student must be enrolled in at least one course.

2. Partial participation.
- The entity in the entity set may or may NOT participate in the relationship.
- That is, if only some entities in E participates in relationships in R, the participation of entity set E in relationship R is said to be partial
- Single line is used to represent partial participation
- Also called optional participations.



Example: It specifies that there might exist some courses for which no enrollments are made.

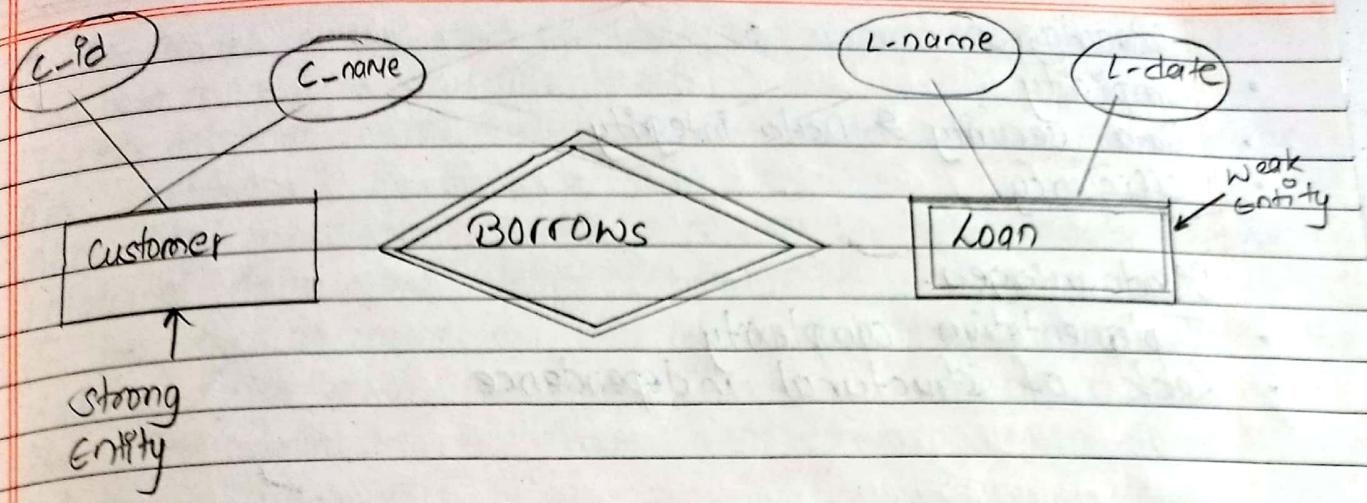
Strong And Weak Entity

Strong Entity

- The strong entity is the one whose existence does not depend on the existence of any other entity in a schema.
- It is denoted by a single rectangle
- A strong entity always has the primary key in the set of attributes that describes the strong entity.
- Set of similar type of strong entities together forms the strong Entity set.
- strong entity always consists of a primary key
- Two strong entity's relationship is represented by single diamond.

Weak Entity

- A weak entity is the one that depends on its owner entity ie a strong entity for its existence.
- A weak entity is denoted by the double rectangle
- Weak entity do not have the primary key instead it has a partial key that uniquely discriminates the weak entities
- The collection of similar weak entities is called Weak Entity Set
- The relationship between a weak entity and a strong entity is always denoted with an Identifying Relationship. ie. double diamond.
- Let us understand this concept with the help of an example; a customer borrows a loan. Here we have two entities first a customer entity, and second a loan entity.



Question: Create a new example that shows strong & weak entity & its relationship.

Alternate Data Model (Hierarchical, Network)

Hierarchical Data Model

- This database model organizes data into a tree-like-structures, with a single root, to which all the other data is linked.
- The hierarchy starts from the Root data, and expands like a tree, adding child nodes to the parent nodes
- In this model, a child node will only have a single parent node.
- This model efficiently describes ~~many~~ many real-world relationship like index of a book, recipes etc.
- In hierarchical model, data is organized into tree-like structure with one one to many relationship between two different type of data, for example, one department can have many courses, many professors and of-course many students.

Advantages
- Simplicity
- Data security & Data Integrity
- Efficiency

Disadvantages.
- Implementation complexity
- Lack of structural independence

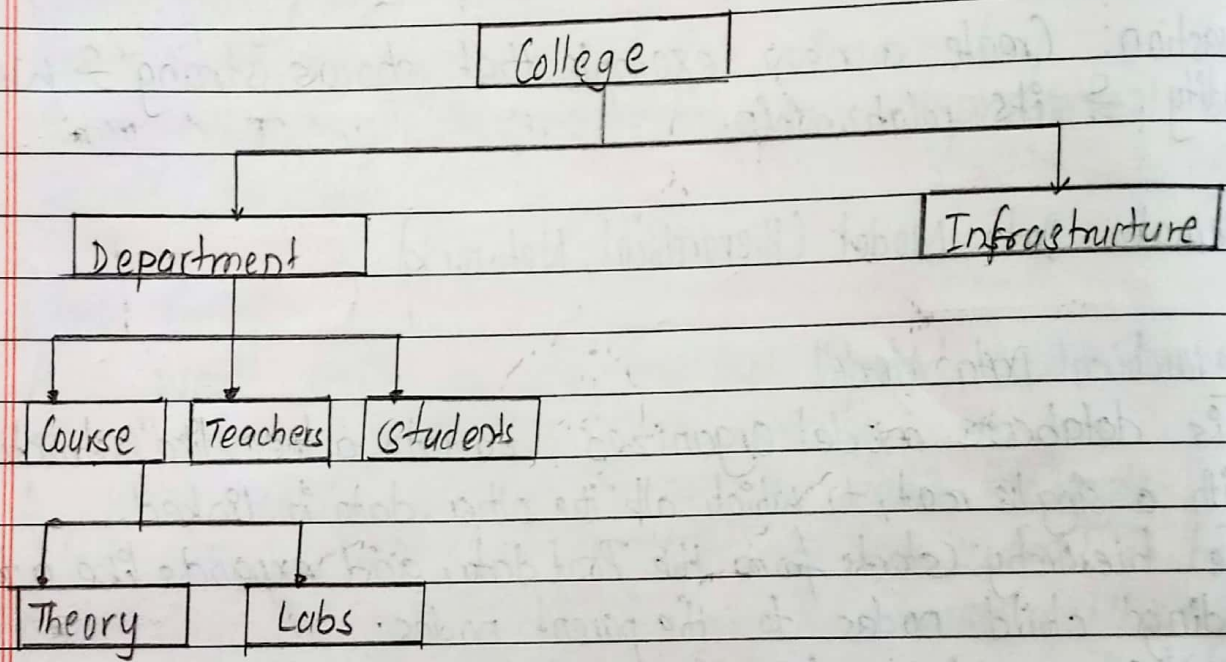


fig. Hierarchical Data Model

Network Data Model
- Network Model is same as hierarchical model except that it has graph-like structure rather than a tree based structure.
- Unlike hierarchical model, this model allows each record to have more than one parent record.
- This is an extension of the Hierarchical model

- In this model data is organized more like a graph, and are allowed to have more than one parent node.
- In this database model data is more related as more relationships are established in this database model.
- After its introduction in 1960s it was widely popular before relational data model were introduced.
- In this type of model the data is more related, hence accessing the data is also easier and fast.
- This database model was used to map many-to-many data relationships.

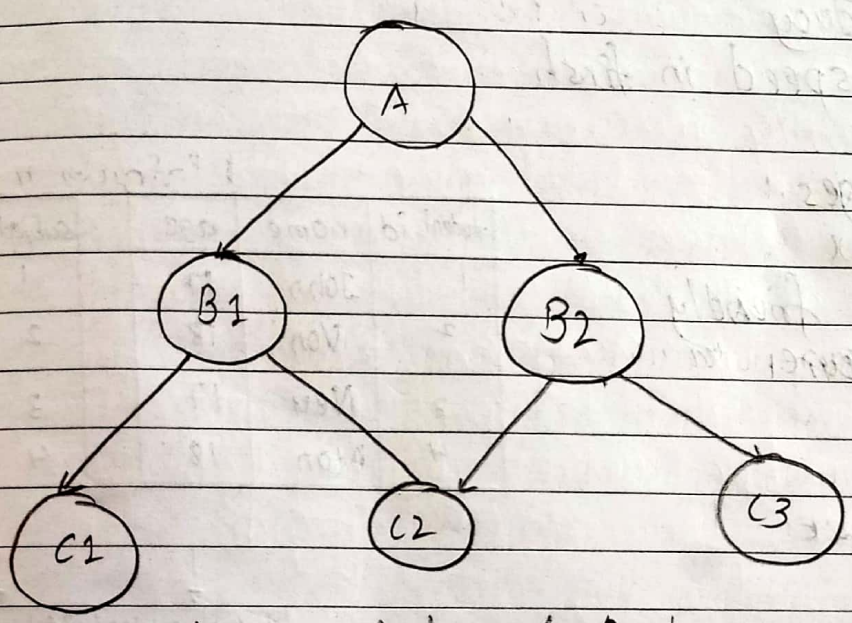


fig: Network Data Model.

Relational Model

- In this model, data is organized in two-dimensional tables and the relationship is maintained by storing a common field.
- This model was introduced by E.f. Codd in 1970, and since then it has been the most widely used database model, in fact we can say the only database model used around the world.

- The basic structure of data in the relational model is tables
- All the information related to a particular type is stored in rows of that table.
- Hence, tables are also known as relations in relational model.

Advantages
- Flexible & efficiency
- Relationship many to many
- Table diagram
- Good management
- Less redundacy
- Searching speed in first

Disadvantages:
- Very complex
- Not user friendly
- less secure.

student_id	name	age
1	John	17
2	Von	18
3	Neu	17
4	Man	18

subject_id	name	teacher
1	Java	Mr.J
2	c++	Miss C
3	C#	Mr.C(Hash)
4	php	Mr.PHP

student_id	subject_id	marks
1	1	98
2	2	78
3	1	76
4	2	88

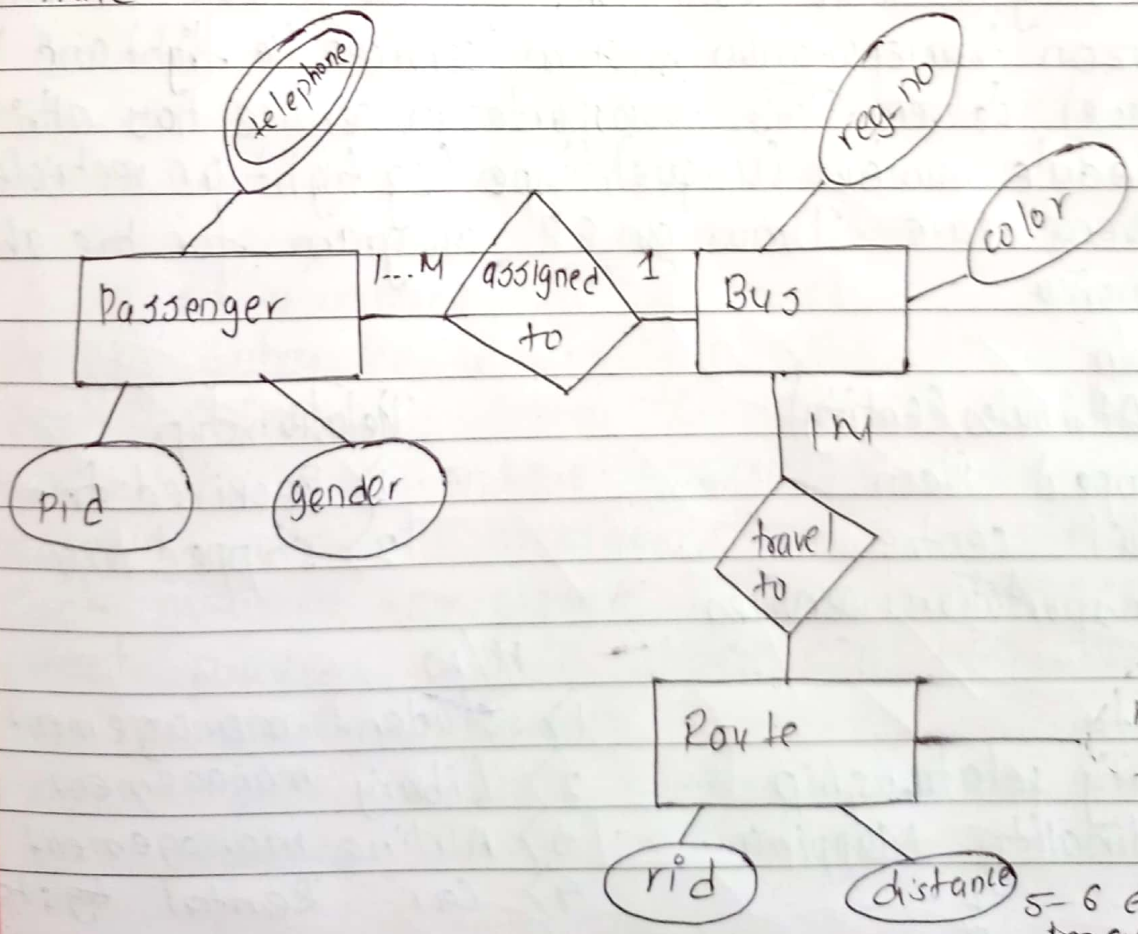
fig: Relational Model.

Consider a bus ticketing system that record information about passenger bus & route. passengers assigned to a bus and bus travel to route. A bus is assigned to many passenger and the passenger can be assigned to only one bus. Many buses travel in ~~only one~~ to same route but a bus can travel in only one route the atribute of passenger are pid (unique) gender & telephone (multivalued). Similarly bus contain reg_no (unique) and color as attribute. root contain rid (unique), distance, rate (base on distance as attribute s draw the ER diagram for the cenario given

Soln

Entities
passenger
bus
route

Relationship
assigned to
travel.



5-6 entities imagine

DOB, age (based on DOB)
derived
attribute

1 ... M — R — 1 many to one relationship.

1) Amazon prides itself on having upto date on the processing and current location of each shipped item IT realise on a company with information system. shipped item can be have attribute as item number (unique), weight, dimensions, insurance_amount, destination and final_delivery_date shipped item are received into the amazon system at a single retail centre. Retail centre have attribute as type (unique_id (unique) and address, shipped item make their way to their destination with one or more standard amazon transportation system (services (ie flight and vechi vehicle) delivery. This transportation service has attribute schedule number (unique), type (eg flight or vechicle) & delivery route Draw an ER diagram for the above cenario

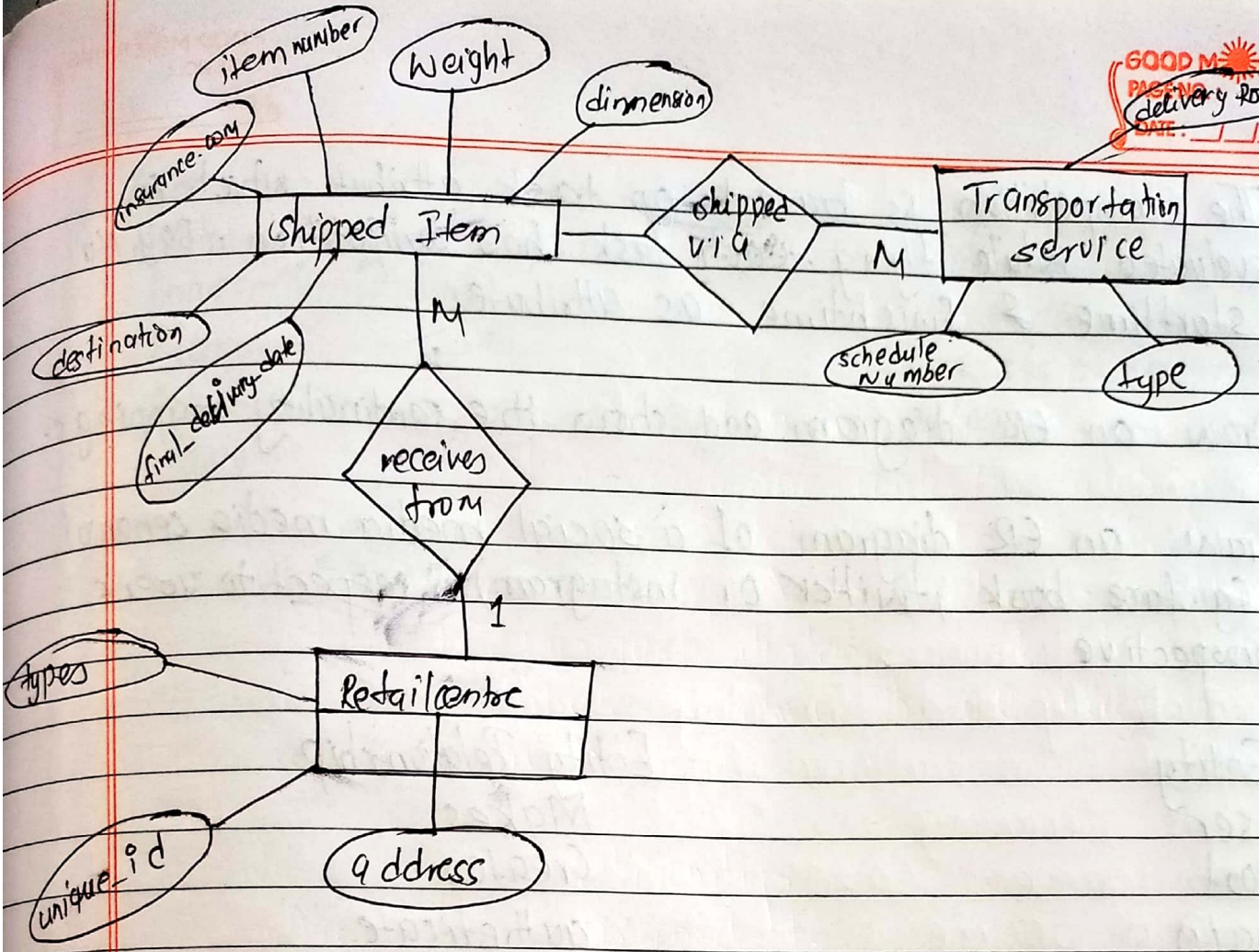
Soln:

Attributes (Entity)
1) shipped item
2) Retail centre
3) Transportation services

Relationship
1) Received from
2) shipped via

* Entity
* Entity relationship
* Cardinality Mapping
  1 ... 1
  1 ... M

H/W
1) Student management system
2) Library management system
3) Airline management system
4) Car Rental System



1 - - M → One to many

M - - M → Many to many

2) Draw an ER diagram for an E-commerce system showing cardinality mapping

* You are organizing an Inter university gliding competitio

→ A no. of university have participated as a team and need sto pay Entry fee, university name as attribute.

→ Each university team consists of no. of players who takes part in the competition, everybody who competes must be member of one of the teams.

→ Each players has player_id name and experience_le

→ each players flies a glider and has crew capacity as constraint while flying

→ Each glider consists of type, call sign & num_seat as attributes

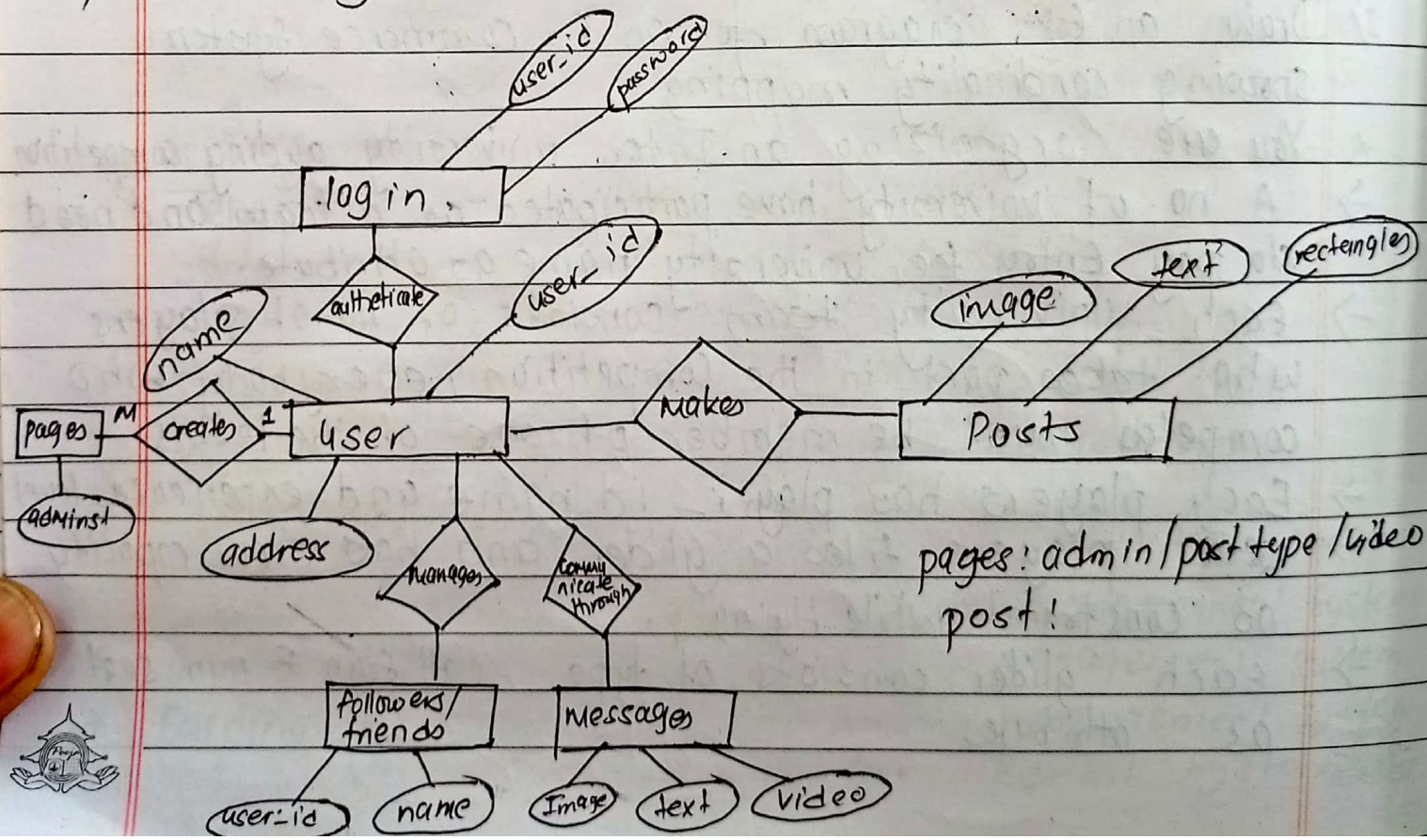
GOOD MORNING
PAGE NO.:
DATE: / /

- The competition is based on task attribute which is evaluated while flying. Each task has competition · Day No starttime & Finishtime as attribute.

Draw an ER diagram and show the cardinality mappings.

3) Draw an ER diagram of a social media media cenario Eg. face book, twitter or instagram) in respect to users prospective

	Entity	Entity Relationship
1)	User	Makes
2)	post	Creates
3)	login	authenticate
4)	pages	manages
5)	followers / friends	add / delete (Manipulate)
6)	Message	communicate through



pages: admin / post type / video post!

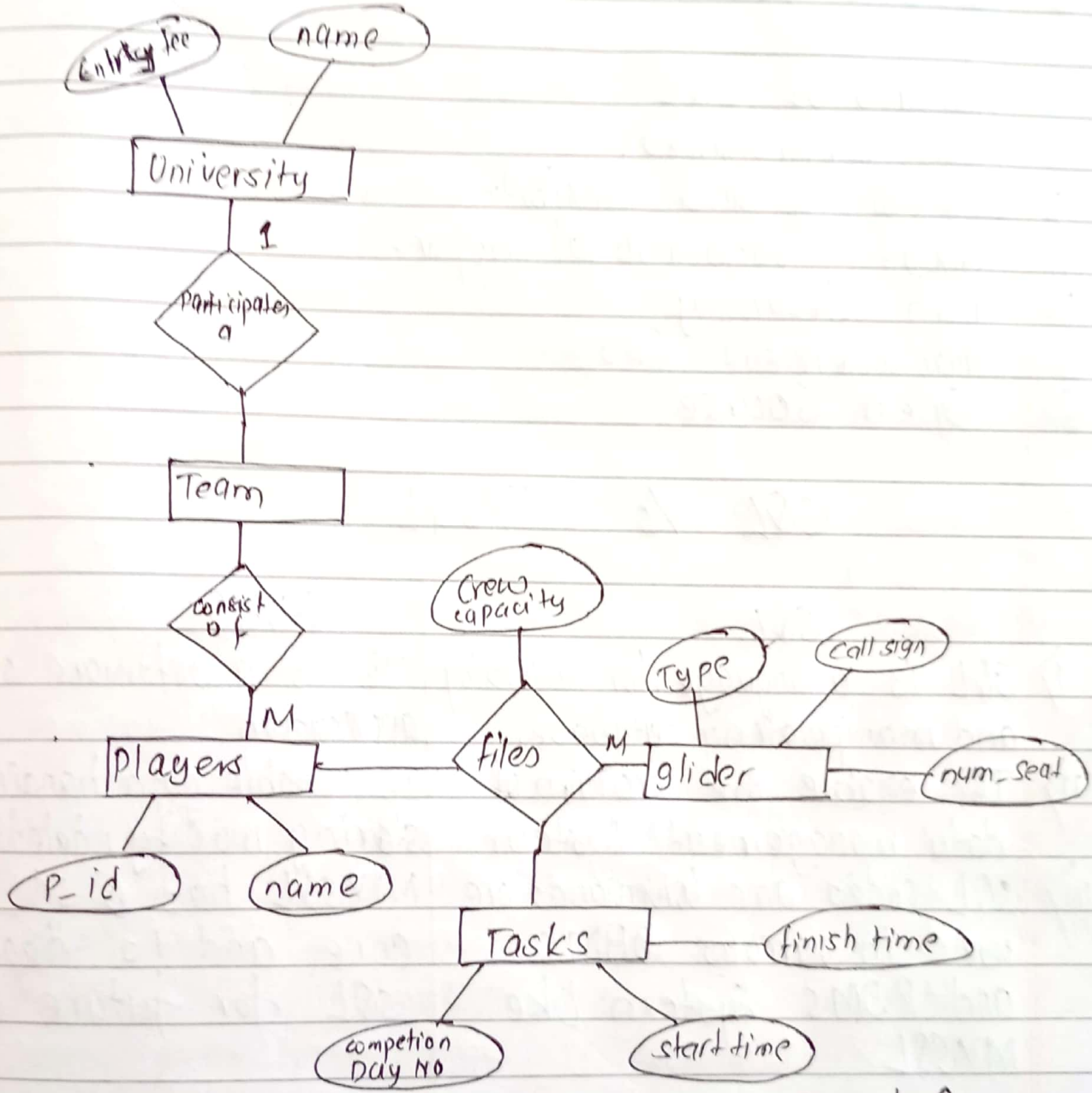
Ans

Entities

University Team
Player
Glider
Task

Relationship

participated as
consists of / member of
files.



1 ... 1 ⇒ One to One

M ... M ⇒ many to many.