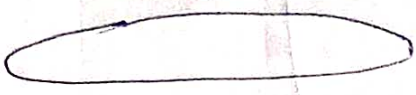
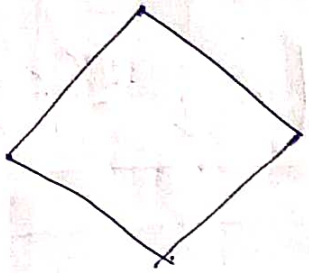
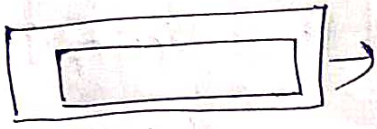
→ Strong Entity



→ Strong Attribute

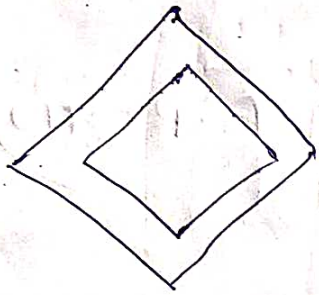


→ strong relationship

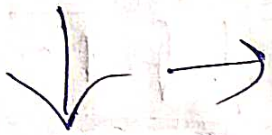


→ weak entity

→ weak attribute

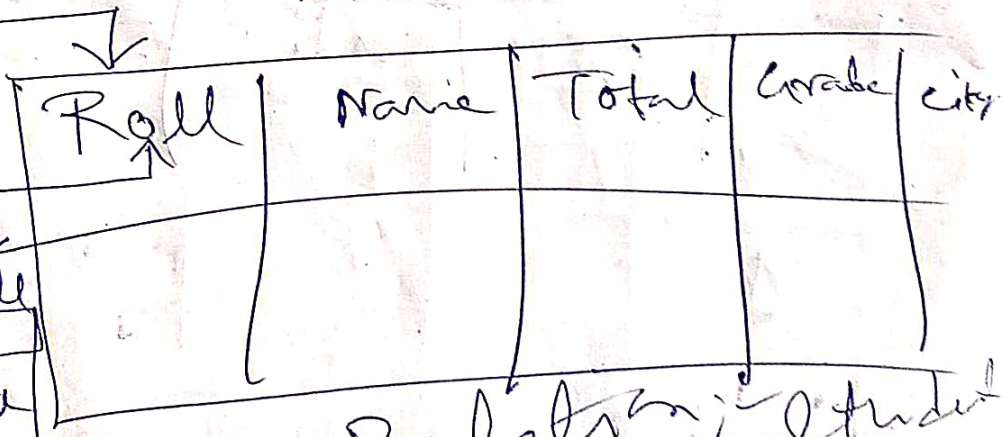


→ weak relationship

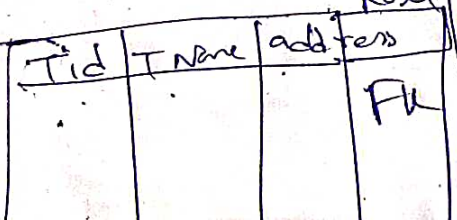


→ linker

PK



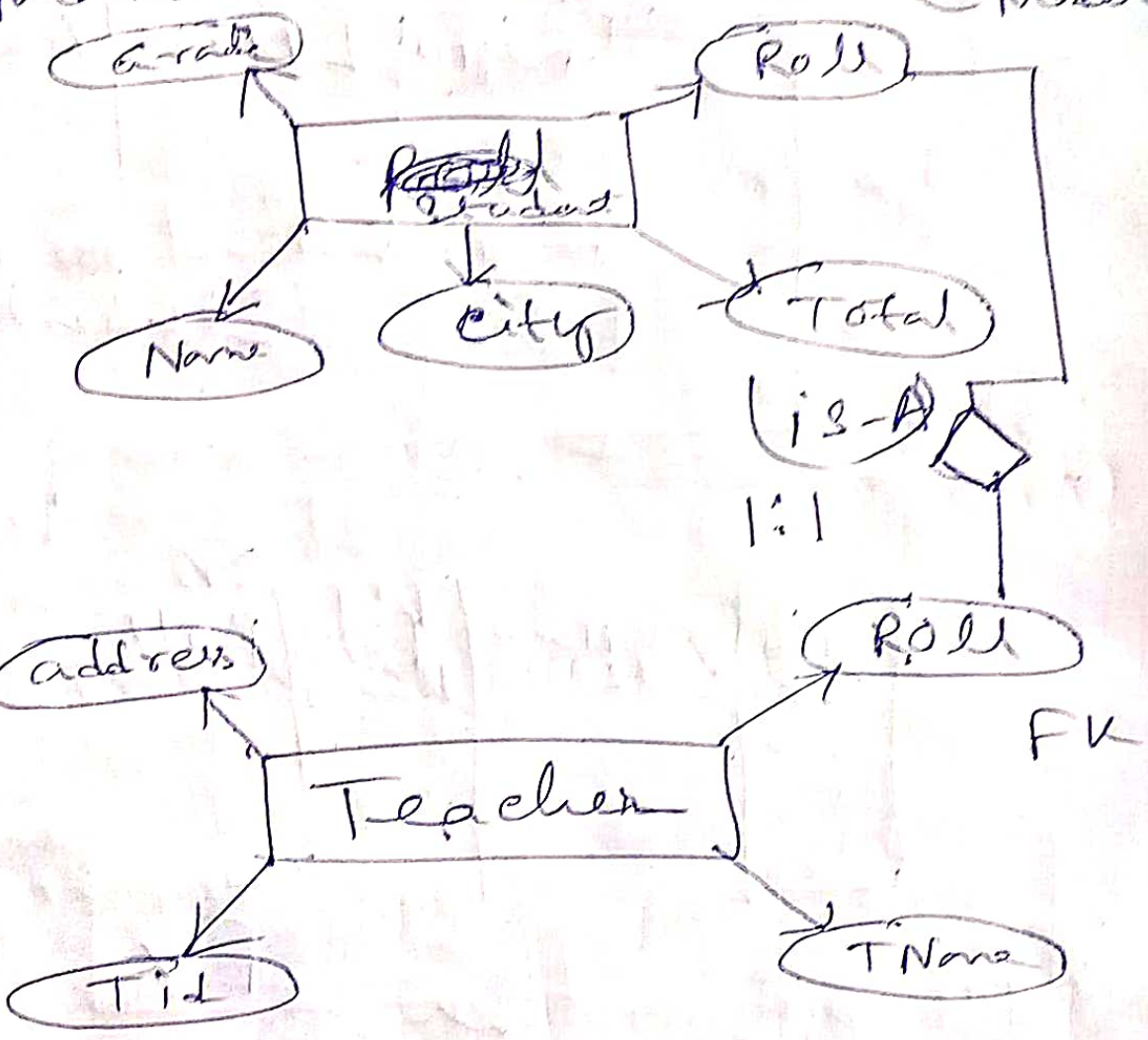
Teacher



Relation; Student

Data Model

- ER-Model: Entity Relationship Model
- Hierarchical Model
- Network Model
- Relational Model

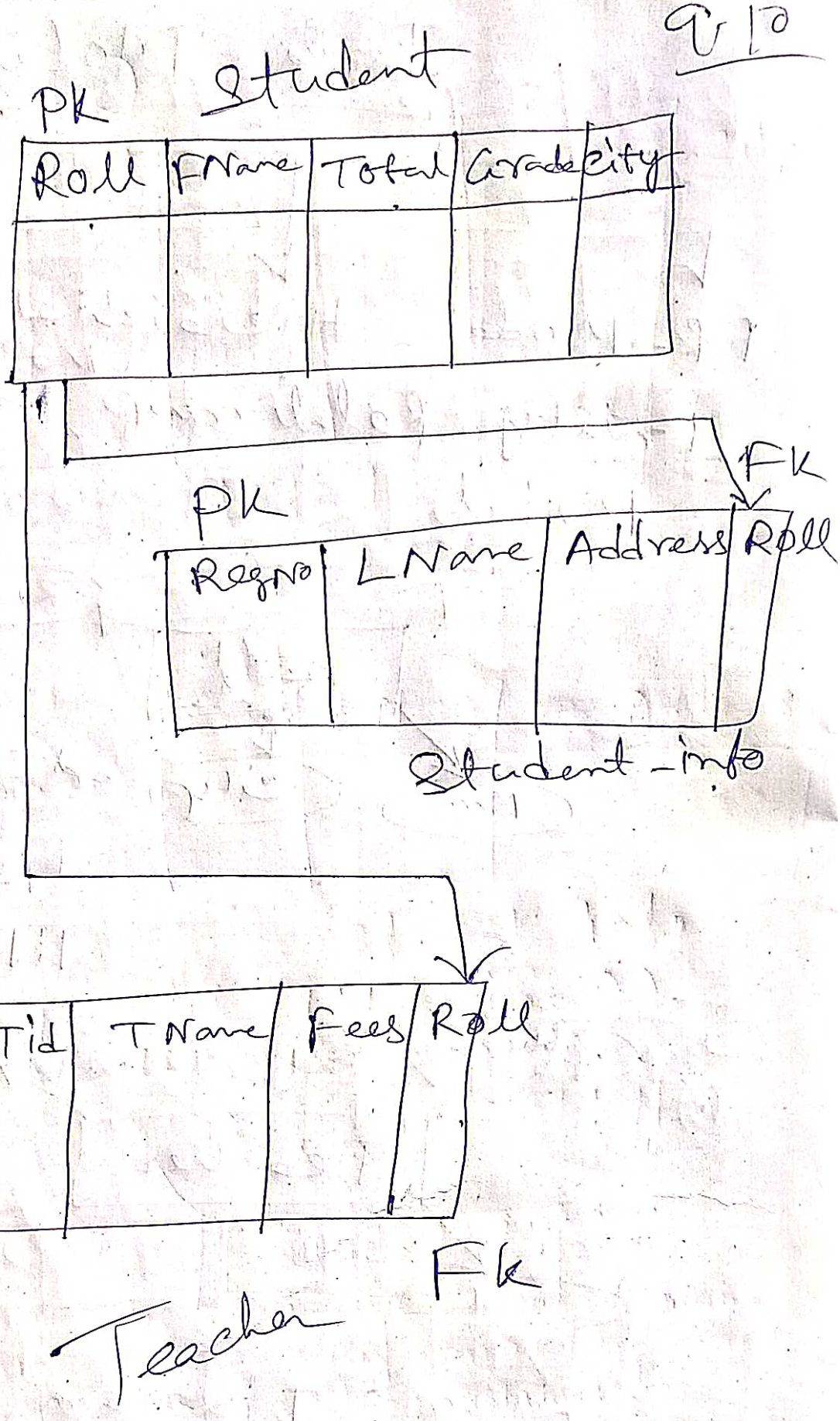


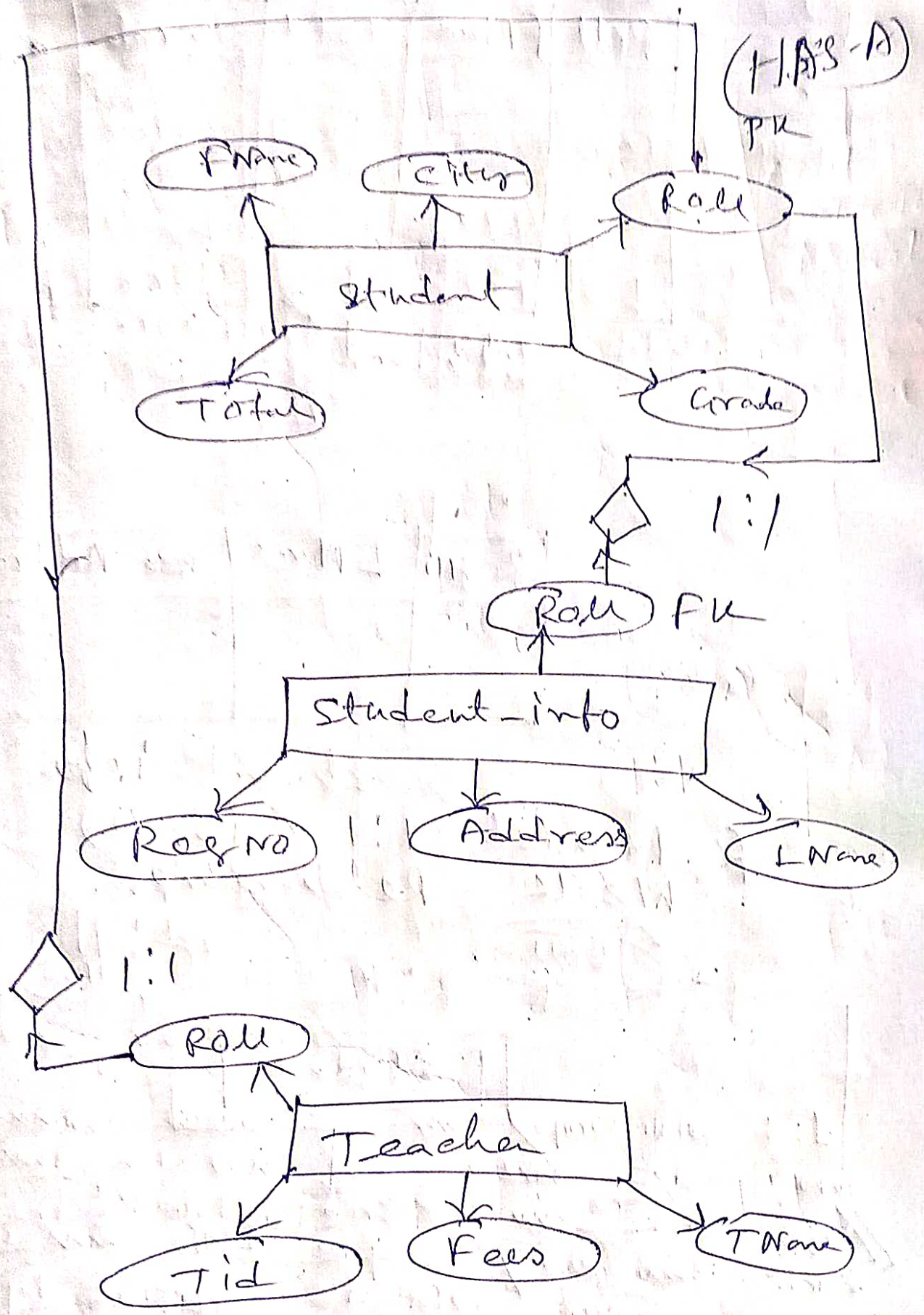
E-R Diagram 1:1

(IS-A) strong Relationship

(is-a)

Q 10


Student
PK
Roll
FName
Total
Grade
City
FK
PK
RegNo
LName
Address
Roll
Student-info
Tid
TName
Fees
Roll
FK
Teacher



E-R Model 1:M Relationship

Strong entity

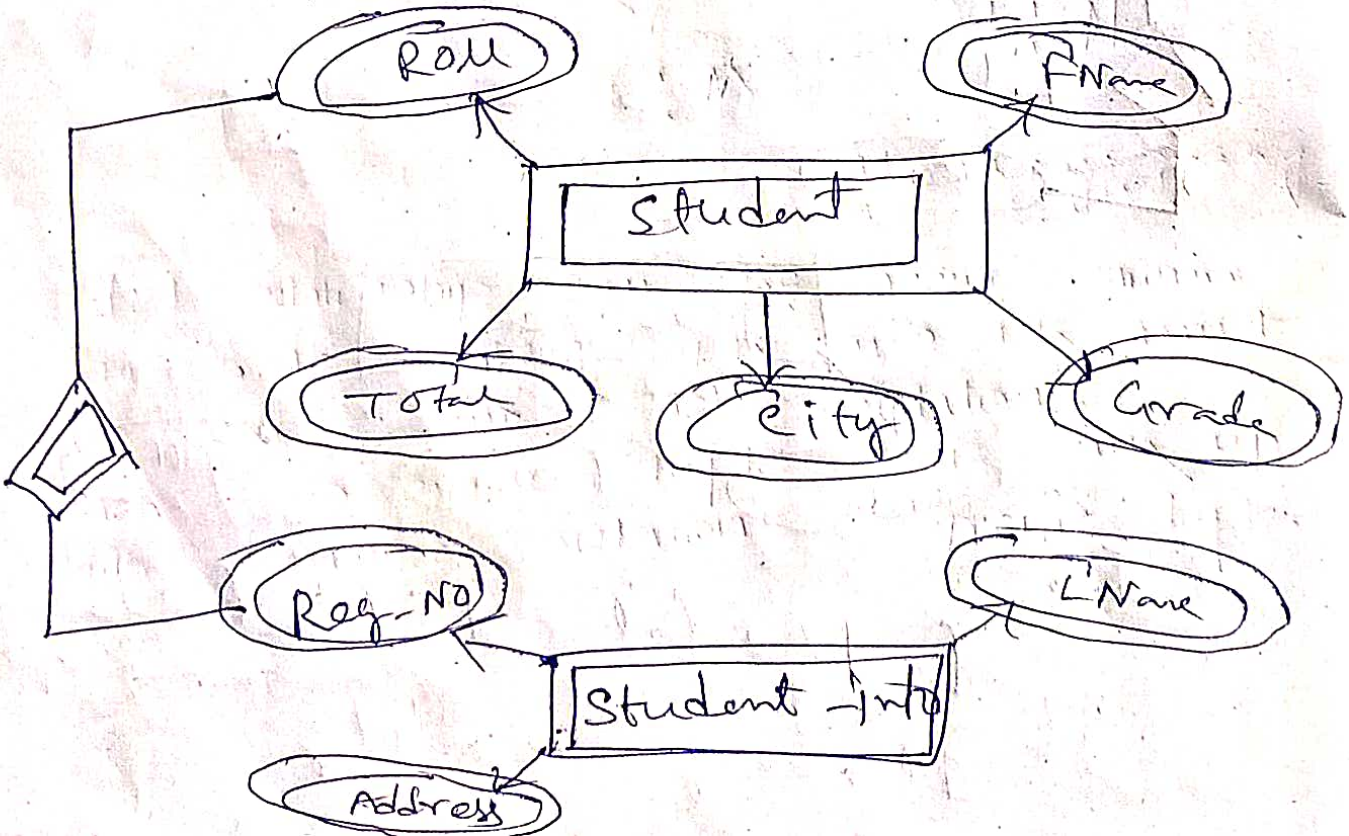
Roll	FName	Total	Grade	City

Student

RegNo	LName	Address

Student_info

E-R Model 1:1 (is-A)
Weak - Relationship..



Weak relation

with in a relation if primary key is not set then that column in which primary key can be there. That attribute is known as weak attribute and the relation is known as weak entity.

If we set relationship between two table where primary key and foreign key does not exist. or we can set relationship between two table where in both the ~~key~~ table primary key exist. is known as weak relationship.

E-R Model :-

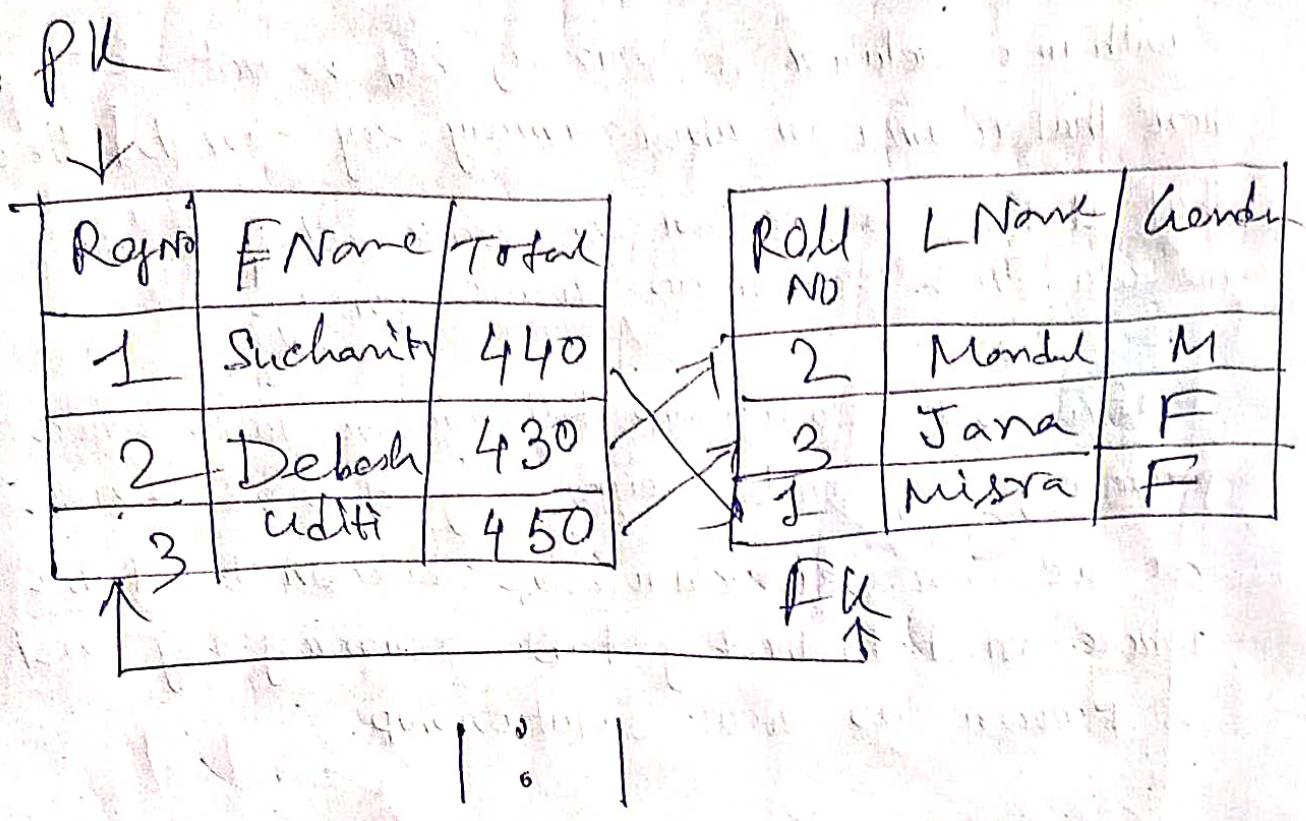
~~Adv~~ Adv :-

(1) 1:1 Relationship support ✓

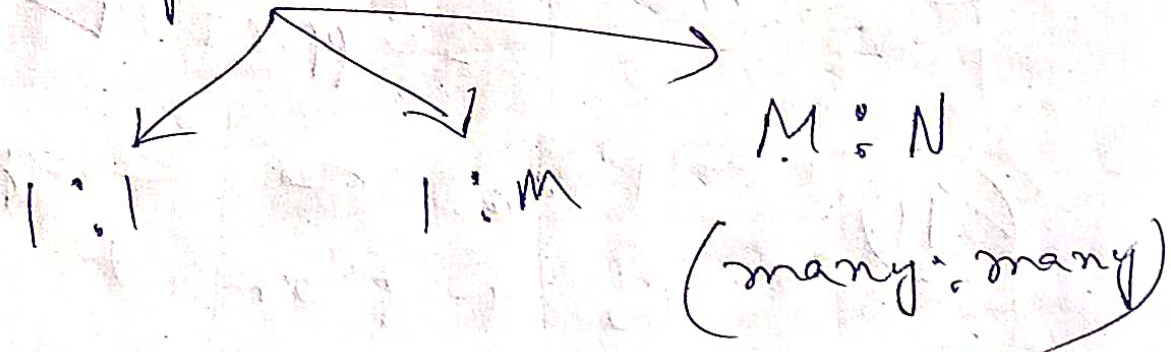
(11) It is easy to implement

disadv :-

(1) 1:M, M:1, M:N does not support

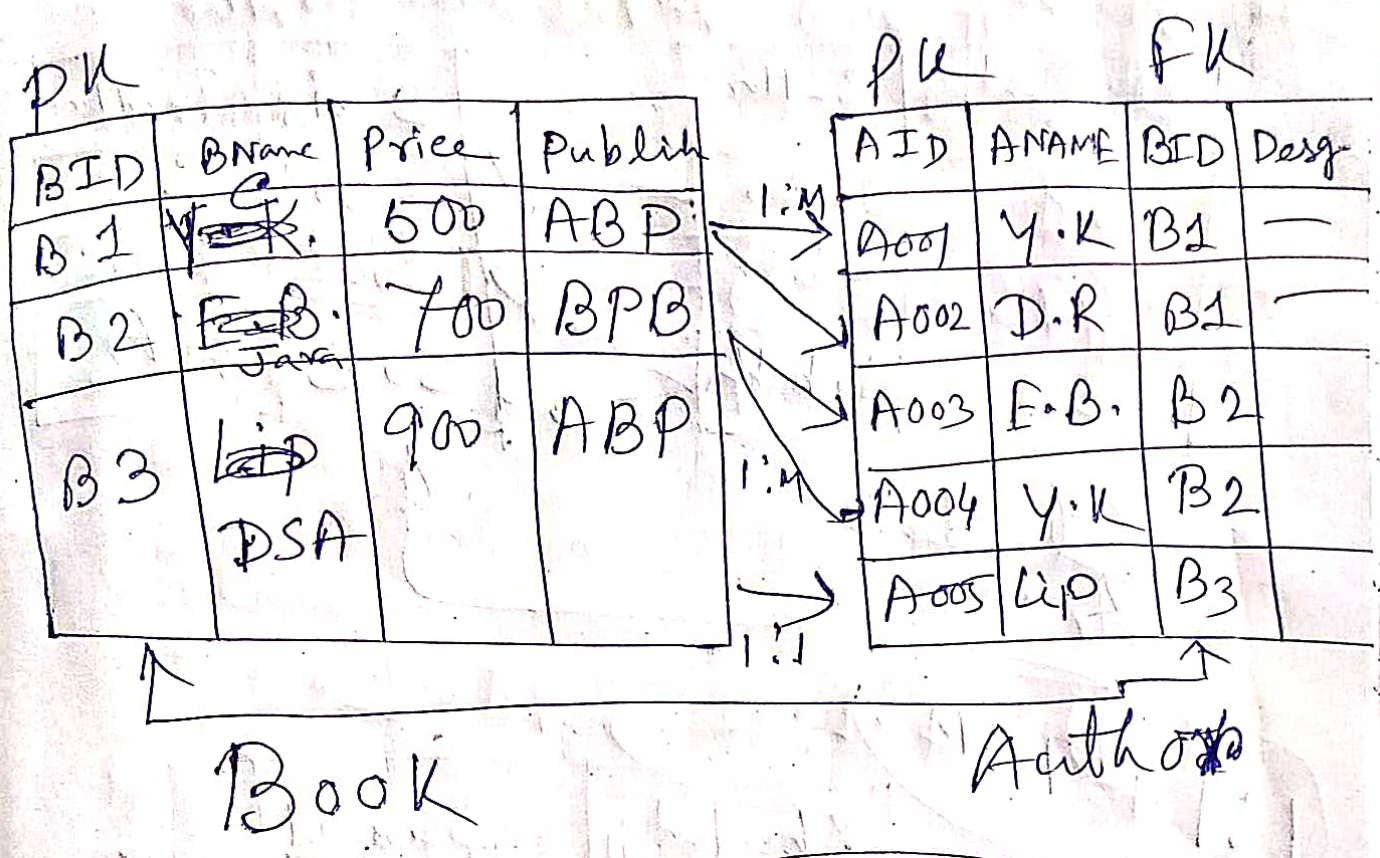


Relationship



1:1 relationship can ~~be~~ occur within two table where PK and FK must be exist and each relation's each and every tuple is link up with another relation's specific tuple.

1:M / M:1

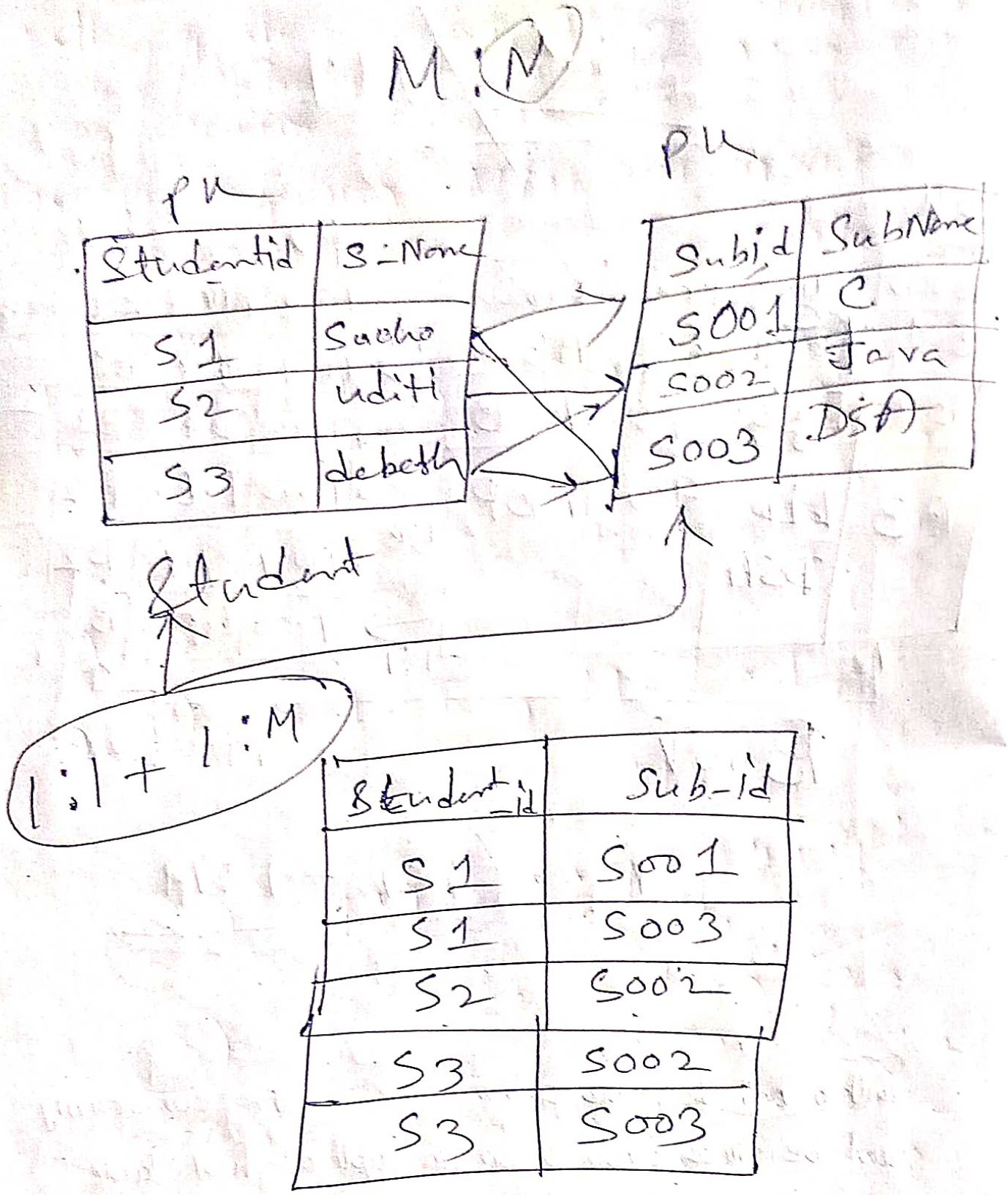


1:M = 1:1 + 1:M

Within both the table ~~prima~~ PK and FK connectivity must be there. and a single tuple of first table must be connect with one or more than one tupple of foreign key or second table, Then that relationship known as 1: Many.

With in 1: Many relationship 1:1 and 1: Many relation-ship must be present

M:N

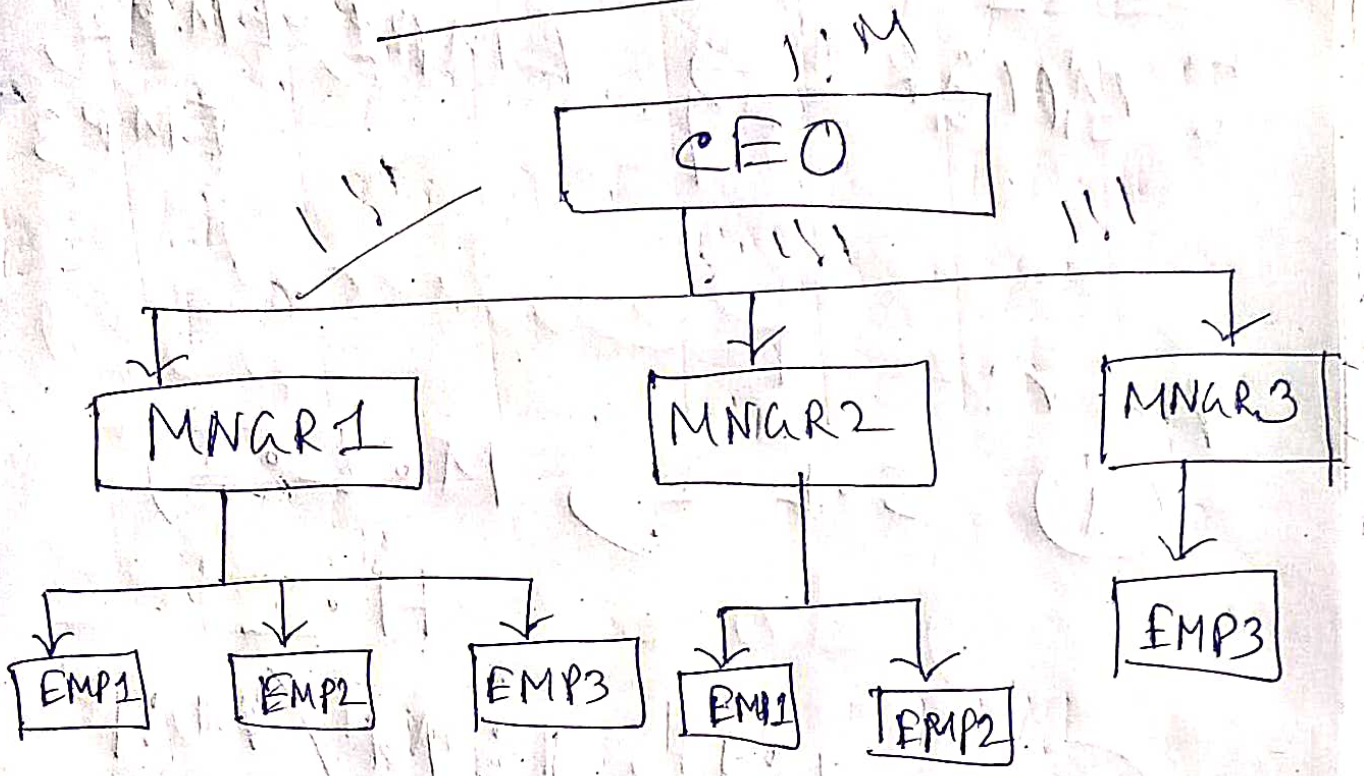


Within the both the table PK and FK concept must not be present. We can connect both the table PK and PK relationship. We have to make a relationship in our own. More than one tuple of the first table can be connect

with one or more than one tupple of second table tuple. then that relationship known as many: many.

In the above case weak Realationship must be present.

Hierarchical Model



Multilevel inheritance:-

CEO → MNGR1 → EMP1
" → " → " 2
" → " → " 3
" → " → " 3

S. L. I →

CEO → MNGR1
" → " 2
" → " 3

MNGR1 → EMP1
" → " 2

Hierarchical In.

CEO → MNGR1, MNGR2, MNGR3

MNGR1 → EMP1, EMP2, EMP3

Adv:-

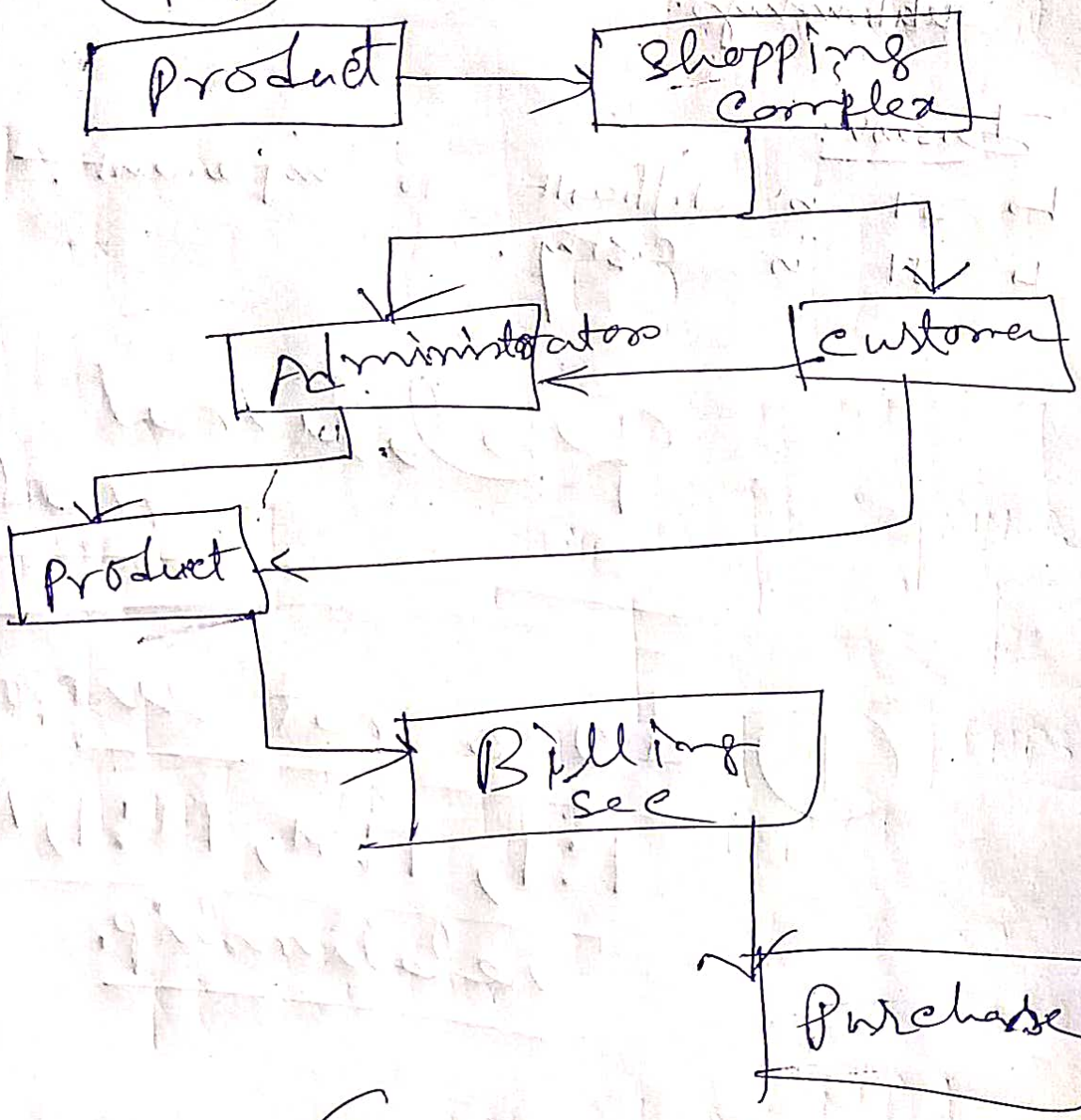
1. S.L.I, M.L.I, H.I
2. 1:1, 1:M/M:1
3. ~~Implementation cost~~ ~~Lower~~ upper layer node search is easier than lower layer

Disadv:-

1. M:N ✗
2. Implementation difficult.
3. Lower layer searching is time consuming

Network Model

1:M
M:1
M:M
1:1



Hybrid

S.L.I

M.L.I

M.P.I

Hierarchical I.

1:1 ✓

M:1 / 1:M ✓

M:N ✓

Adv:

↳ It supports S.L.I, M.L.I, Multiple I. and Hierarchical Inheritance as well as Hybrid Inheritance.

↳ It supports 1:1, 1:M, M:1, M:N relationship.

Disadv:

↳ It is difficult to implement.

↳ It is costly.

Relational Model

(1) does not support 1:1, 1:M, M:1, M:N relationship.