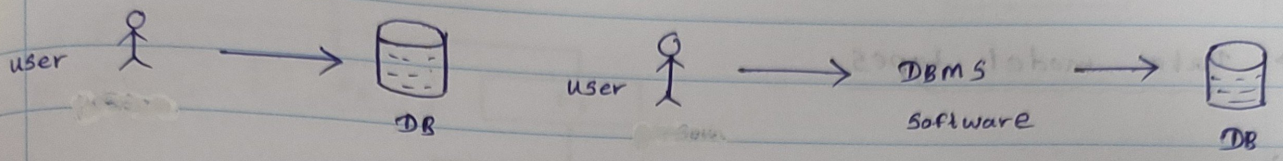
No:

Date: / /

Database management (DBMS)

07/06/2022

Database → Collection of related data.
DBMS → Software tool to manage database.
DB systems = DB + DBMS



Tutorial 1

Should pay - A Security facility, crash control.
Should not pay - crash control, concurrency control

* Scrooge should pay for the security facility because to protect the data by hacking, malware or viruses. And he should pay for the crash control because, he is making payments.

a) what is view mechanism and query language.
View mechanism -

22.06.21

Data modelling & Data models

What is an entity?
- unique and distinct object used to collect and store data.

ex:- Database of university → Entity (Students)
(university is not an entity)

No: Date: / /

Student ← (entity)

ID	name	Gender	m.n.	← attributes
10072	Tom	male	077-....	← peels or Columns

> Data model types

1) Hierarchical modle.

Database
↑
Table / file
↑
record (row)
↑
field (column)
↑
Bite
↑
Bit (binary)

- Data Hierarchi -

Bussines — Technical
cm RM pm

cm -
RM -
pm -

• Data request through → Database.

login
user name p.w.
Data request

client address
c No.
c city
c street

functional -
non-functional -

what is redundant data ?

* redundant data is the same data unnecessarily stored at different places.

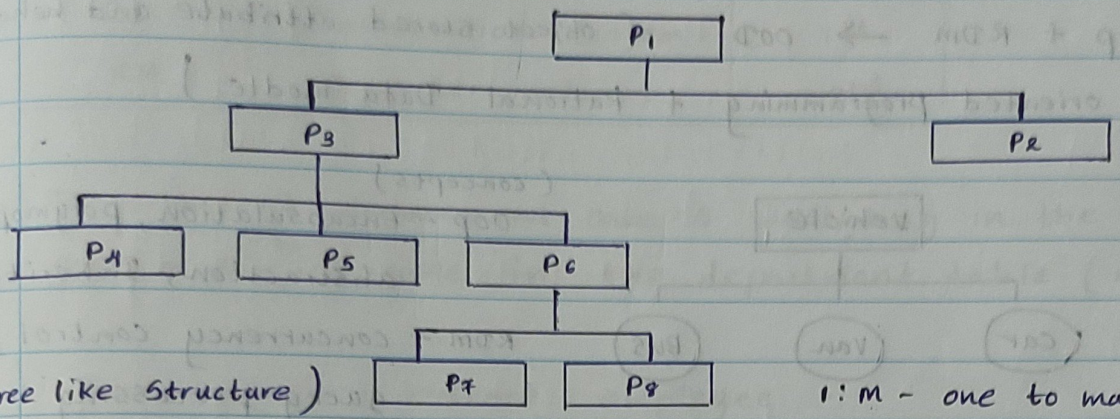
* we have only one path to reach any node to access data. So in that case it can anser only very limited set of queries / questions.

Atlas

No :

Date: __/__/____

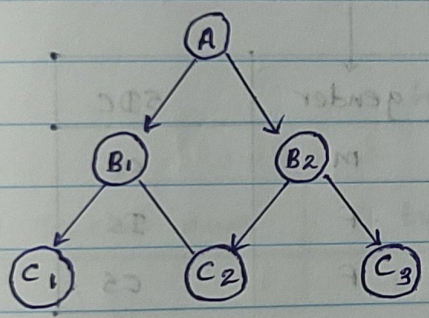
example Hierarchical modle :-

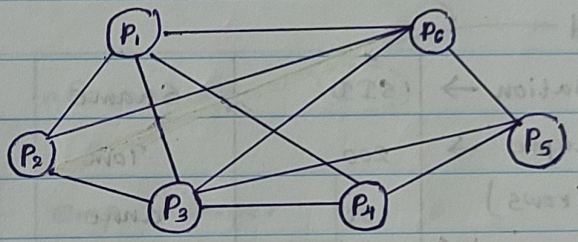


(tree like Structure)

1 : M - one to many relationships

2) Network modle - (graph Data Structure)





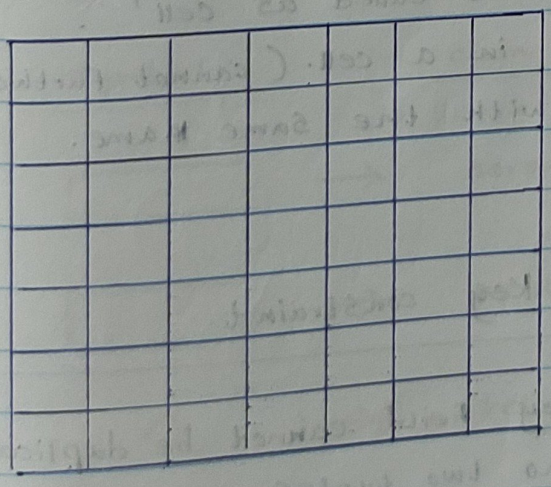
1 : m - one to many

m : N - many to many relationships

* There are so many paths to reach the data. This modle can answer complex queries.

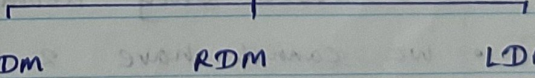
28.06.22

3) 2D / Flat modle (flat modle)



2D modle.

Data model



- HDM - High level Data model.
- RDM - representational » »
- LDM - Low Level » »

Atlas

No: Date: __/__/__

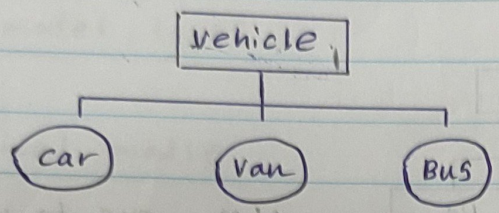
(*) to page no: 001

4) Object oriented data modle (OOD)

OOP + RDM → OOD (object stored attribute and behaviours)

(object oriented programming + Rational Data modle.)

ex:-



(concepts)

OOP - Encapsulation, polymophism, abstraction, Inheritence.

RDM - concurrency control, query processing.

5) The relational data modle.

ex:- 1) Student table.

attribute / Field → (columns)

Key Field →

(Table) Relation →	SID	Sname	city	gender	SDC
(record) tuple → (row)	200	Tom	col.	m	cs
	100	anne	Jaf.	f	Is
← cell →	300	Jane	gam.	f	cs

- In order to have a unique record, a key feild is created.
- The coloumn order is not important (attribute)
- The Tuple order isn't important. (tuple = row)
- mandatory to have key feild in a relation.
- key feild is important and there arn't be two same rows (tuple), can't be duplicated. key field deatails can't be duplicated.
- Intersection of column and row is called as 'cell'.
- we store only atomic values in a cell. (cannot further divide)
- we cannot have 2 columns with the same name.

➤ constraint.

Domain constraint	Key constraint.
• Specifies the value of each attribute. ex:- 'A' is an atomic value.	• key feild cannot be duplicated. (no two tuples should have the same combination of values for the attributes) (no equal records)
• Command set of boundaries.	

No:

Date: / /

2) Department table

DID	Dname	floor
CS	computer science	G
IS	Information System	1st
CN	Computer network	2nd

- The student department code's including in the Student table. are must be Include in the department table. (records must same)

Q) Store employee.

emp. number	First name	Second name	Emp. Depart. num.

hardware information.

Serial num.	type.	emp. number

- an employee can have multiple hardware devices so the Serial number column in the hadware info. table can be duplicated.

> Entity relation diagram. (ERD)

Symbol — chen notation.

3 main elements ERD.

1) Entity - ▭
2) Attributes - ⬭
3) Relationships - ◇

▭ → strong entity (Key field)

▭ → weak entity (can't identify Key Field - no Key)

◇ ···→ relationship (To connect 1 or more strong entities)

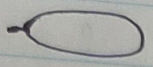
◈ → Identifying relationship (To connect strong entity and a weak entity)

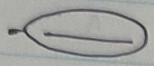
Atlas

* attributes can attach to an Entity or either relationship.

No:

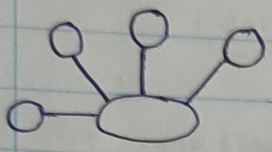
Date: __/__/__

 → Symple attribute (atomic value) (The values can't further Divide)

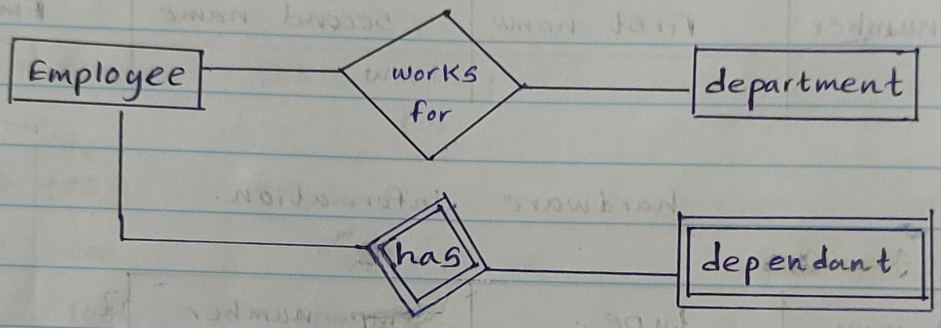
 → key attribute (The main / key value)

 → multi value attribute (can store more than 1 value)

 → Derived attribute (A generated value by us)

 → composite attributes (having multiple components)

ex:-

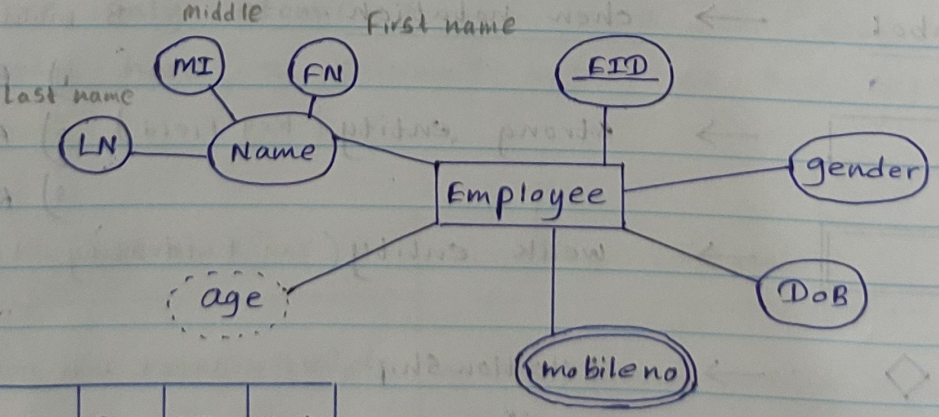


employee

EID	Ename	Egender
100	John	M
200	amy	M
300	Ravi	M
400	Rani	F

Dependent

Dname	Dgender	EID
Tom	M	100
Tom	M	200
Finn	F	200
Mary	F	100



ERD to table

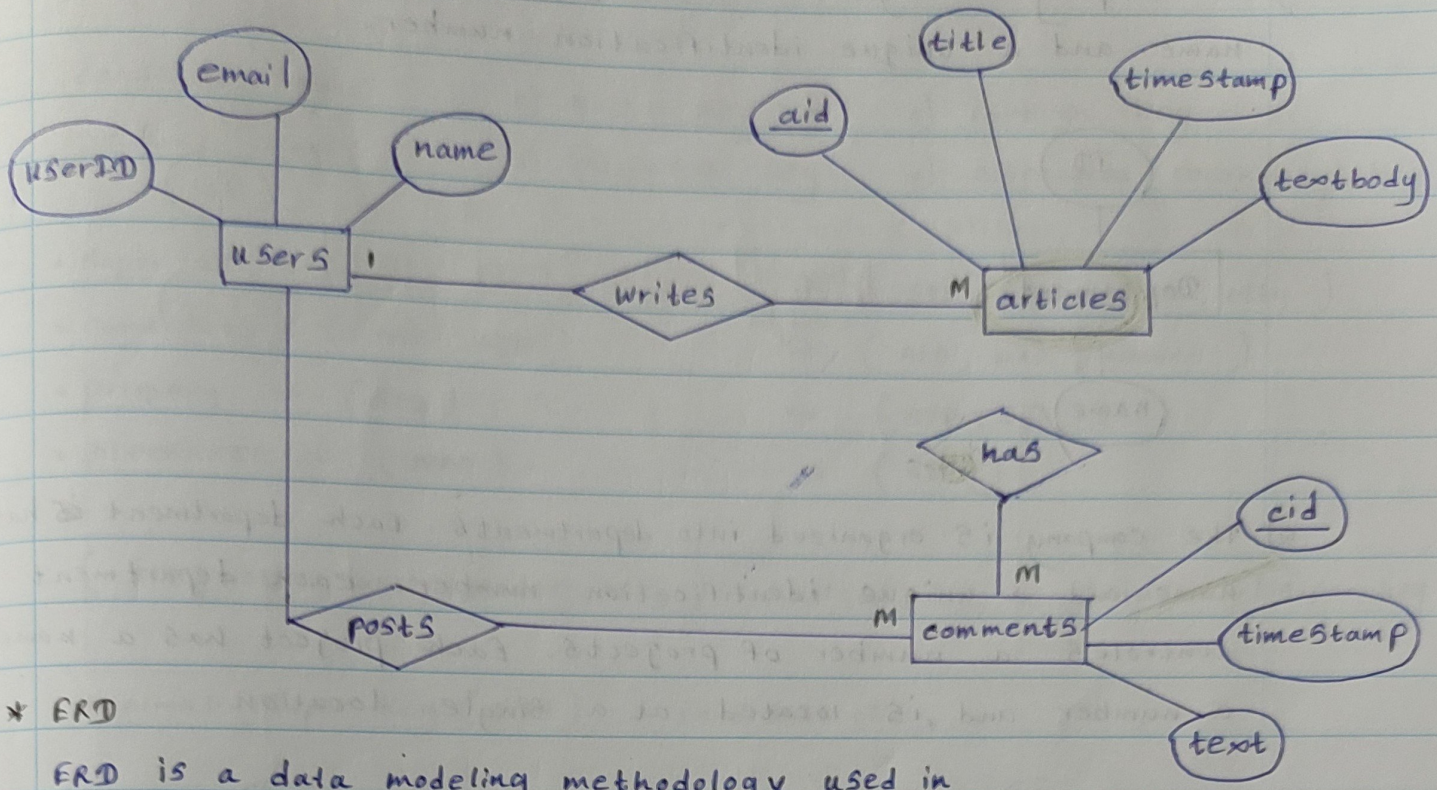
EID	gender	DoB	FN	MI	LN

Atlas

No:

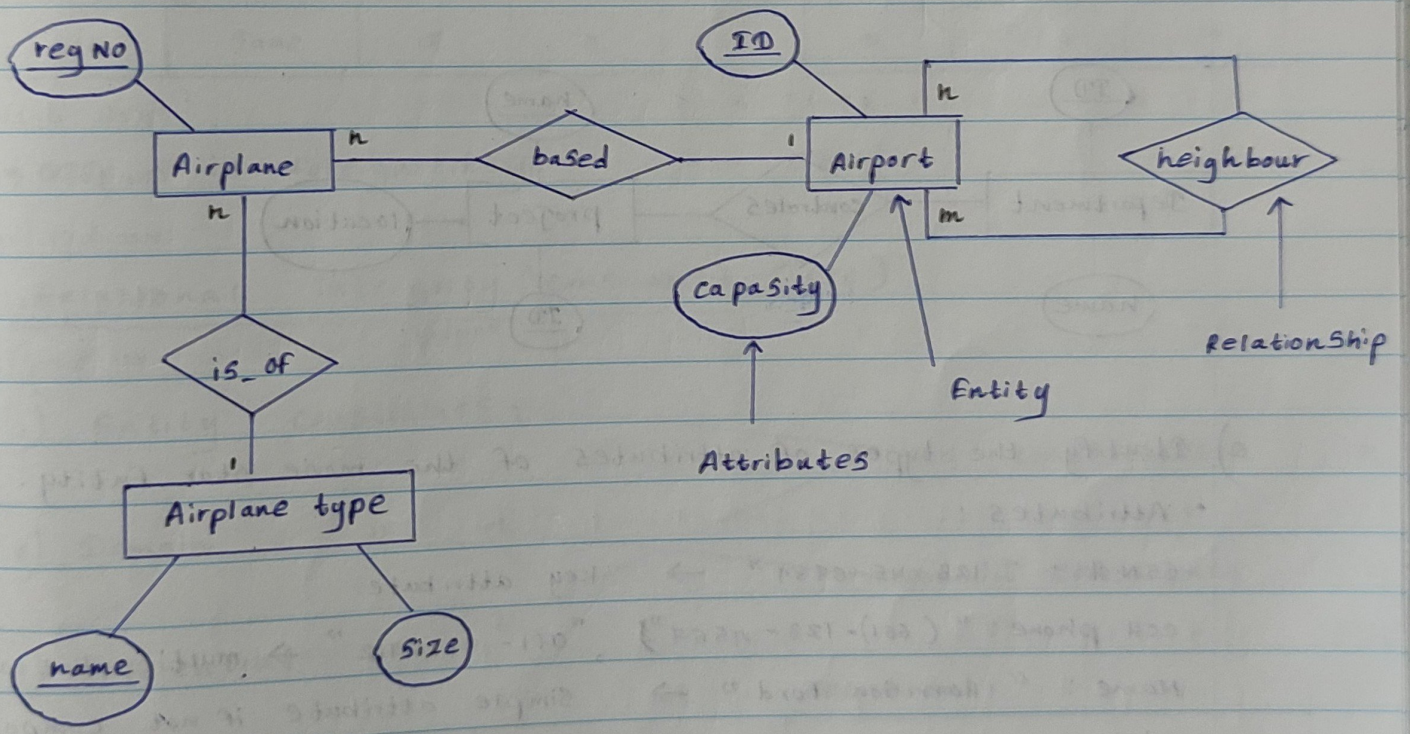
Date: __/__/____

- Sample ERD



* ERD

ERD is a data modeling methodology used in Software engeneering to produce a conceptual data model of a database System.



* Identify what data is important
* Identify what data should be maintained.
* Data modeling is an 'Iterative process' / not 100%. complete one.

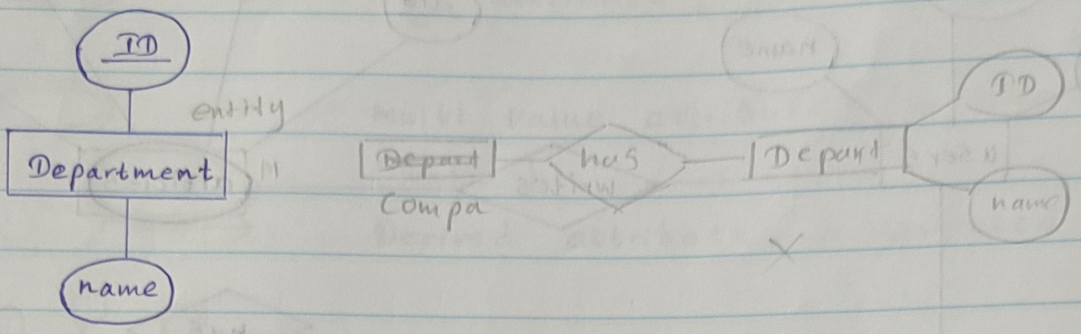
Entity identification → Relationship identification → Cardianility identification → identify attributes → create ERD.

No :

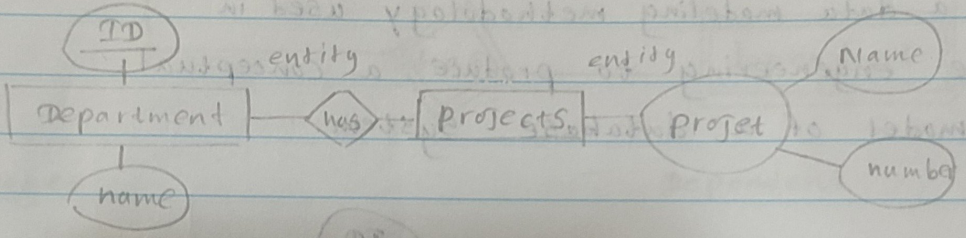
Date: __/__/____

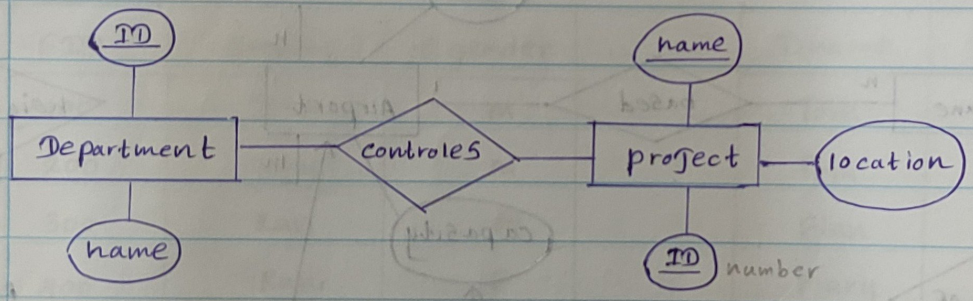
05.07.22.

Q) The company is organized into departments. Each department has a name and a unique identification number.



Q) The company is organized into departments. Each department is has a name and a unique identification number. each department controles a number of projects. Each project has a name, a number and is located at a single location.





"each department controles a number of poject" This part is missing

Q) Identify the types of attributes of the movie star Entity.

- Attributes :

SSN # : "123-45-6789" → key attribute.

cell phone : "(661)-123-4567", "071-8716912" → multi value attribute

Name : "Harrison Ford" → simple attribute if not composite.

Address : "123, main street, LA, CA" → simple or composite.

Gender : male → simple

Birth date : "02.05.1989" → simple

Age : 24 → derived attribute.

- Entity : "movie star"

Atlas

No:

Date: / /

Concept of keys

1) candidate key
2) primary key
3) super key
4) composite key
5) foreign key
6) Alternate / secondary key.

ex:-
- Super key - (.....) → (SID, age, gender, city)
- candidate - (NIC, SID) → (SID, age, gender)
- primary - (SID) → (SID, age)
- Alternate - (NIC) → (SID)

(SID, NIC, age, gender, city) - unique key combination / super key

minimal super key (MSK) ∘ candidate key.

ex- Student

SID	name	age	Dcode
100	Tom	23	1
200	Jane	17	2

Department

DID	name	floor
1	MIS	2nd
2	CS	1st

- DID - is foreign key.

Relational intergrity constraints. (RIC)

1) Entity constraints → Key constraints
2) Domain " → Domain constraints
3) Referential "

Student

SID	name	Sage	DID
100	John	20	01
200	mary	30	01
300	John	20	04

Department

DID	Dname	DFloor
01	MIS	2nd
02	CS	1st
03	SE	01

Atlas

(next page) ★

Date: __/__/____

• exersice 1

1) for each table, identity the primary key and the foreign keys.
2) Do the table exhibit entity intergrity ? answer yes or no, and explain answer ?
3) Do the table exhibit referential intergrity ? answer yes or no, and then explain your answer ?

1)

table	primary key	foreign key
Employee	EMP_code	STORE code
Store	STORE code	REGION code | EMP_code
Region	REGION code.	Region code.

2) yes, every table primary keys are unigue not null.
3) yes, The entries in all foreign keys are entries of the respective primary keys.

• exersice 2

1) primary keys - TRuck_num (Truck table)
Base_code (Base table)
Type_code (type table)

2) Foreign keys - Base_code (Base table)
Type_code (Truck table)

★ from previous page.

• Relational intergrity constraint. (RIc)

1) Entity intergrity constraint - all primary entries are unigue, and no part of a primary key maybe null.
2) Domain intergrity - all the values of an attribute must bound to a specific range.
3) Referential intergrity - requires that a foreign key must have a matching primary key or it must be null.

No :

Date: __/__/____

(*) from 28.06.22. (page 001)

> Features of object oriented data model.

- object - contains data and their relationships
- Attributes - describe the properties of an object.
- methods - describes the behavior of an object.
- class - collection of similar objects with shared structure.
- Inheritance - object inherits method and attributes of parent class

(whatever common to the data entered)

ex:-

activity code ⟶ activity code
activity code ⟶ activity name.

> concept of keys

12.07.2022.

1) composite key

normally key is a single attribute. composite key can store multiple attributes. it means there is an attribute combination in the key ($A_1 + A_2$)

ex:- mobile no:

SID	mobile no.	
100	071-65....	Same SID
100	077-	diff. mobile no.
200	071-55....	
300	072-5653666	diff. SID
400	072-5653666	Same mobile no.

* So the primary key of this table is (SID + mobile)
* we called as the ' composite primary key '.

No:

Short note.

Date: __/__/____

18.07.2023

1) ER Diagrams.

2) ERD components → entity
→ attributes
→ Relationship.

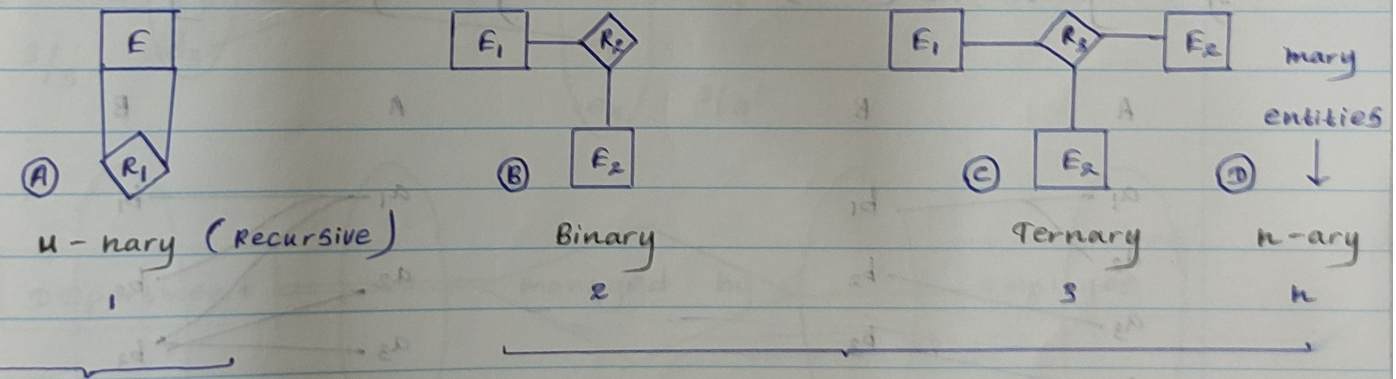
3) RIC → Entity intergrity constraint
→ Domain " "
→ Referencial " "

4) Attribute
- type
- key - SK, PK, AK, FK, com. K, can. K

- Degree of relationship (DOR)
- connectivity
- cardianility
- attribute.
- participation.

Relationship

number of entity types that participate in a relationship.

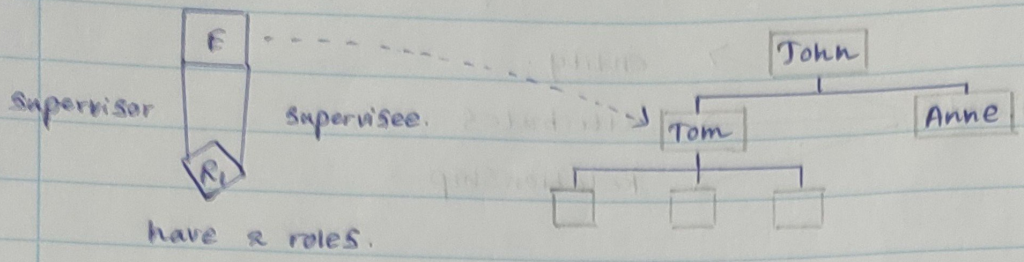


Entity has two roles

Every single Entity has only one role.

No: _____ Date: __/__/____

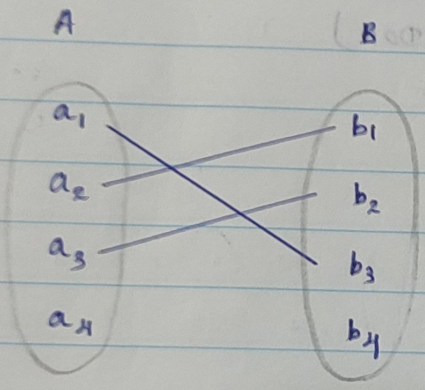
* defining the role in a 'u-nary' relationship.

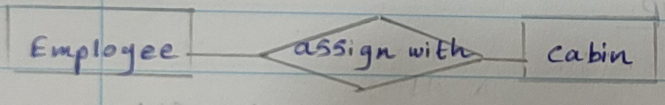


Connectivity

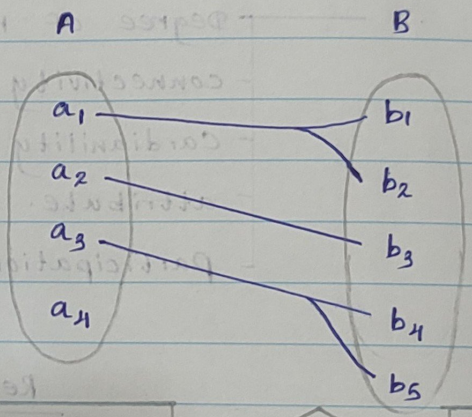
'the number of instances of one entity that can or must be associated with each instance of another entity.'

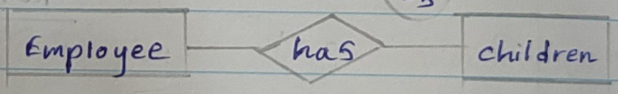
1) one to one (1:1)



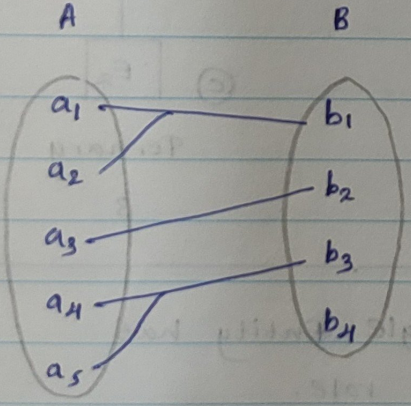


2) one to many (1:m)

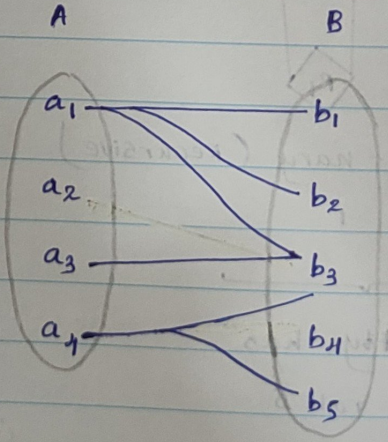


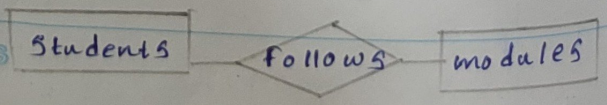


3) many to one (m:1)



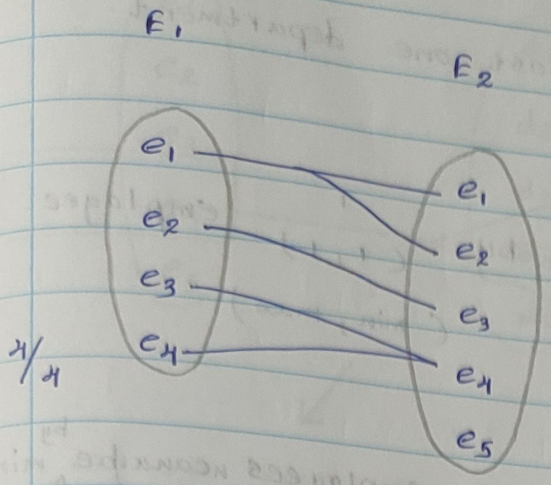
4) many to many (m:N)





Date: __/__/____

* participation

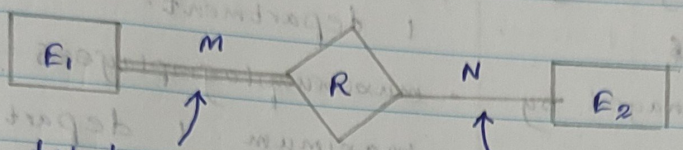


4/4 4/5

Single line - Partial participation

Double line - total participation.

(many to many - connectivity)



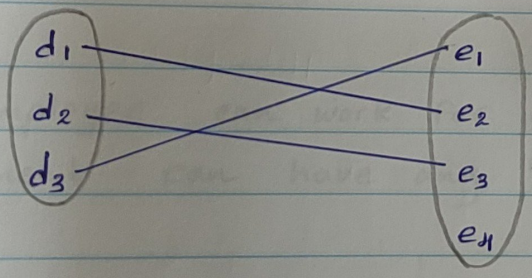
total participation

Partial participation

example :-

1) A department is managed by an employee. All the department are managed by employees.

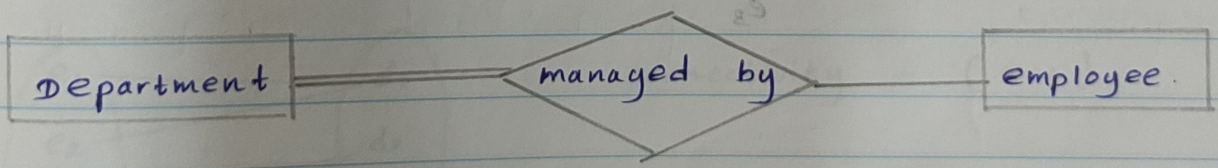
department employee



3/3 3/4

1 : 1

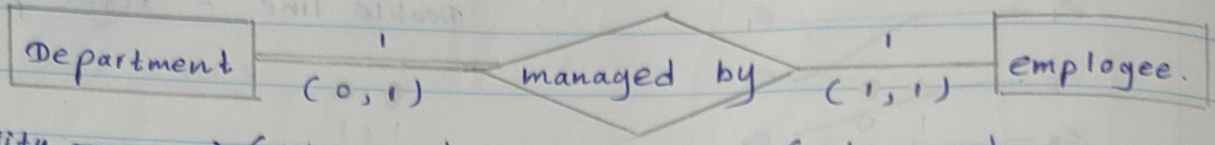
one to one



No: Date: / /

Cardianility

* An employee can manage at most one department.



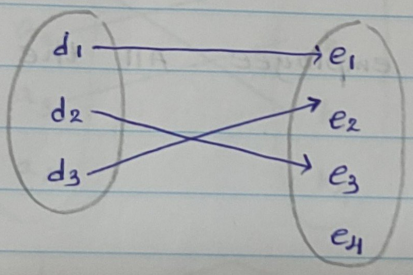
cardianility ⟶ (min, max)

(min, max)

- department can managed minimum 0 emplogee.
- department can managed maximum 1 employee.

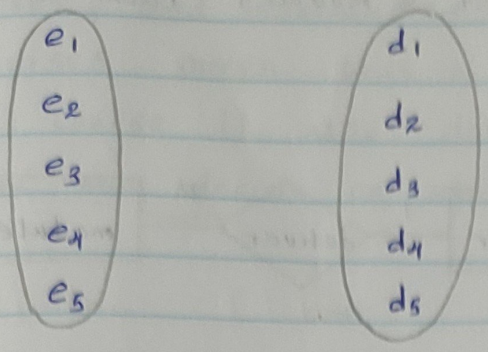
- an employee can manage by minimum 1 department.
- an employee can manage by maximum 1 department.

dep. Empl.



No: Date: / /

Q2) employee department



connectivity

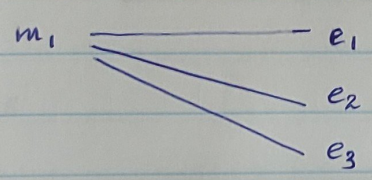
cardianility

এভাবে complete করবা.

1) questions related to the above answer.

:) A department has exactly one manager and one employee can mange at most one department.

manager emplo.

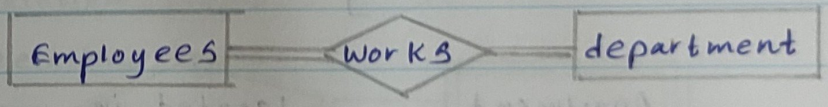


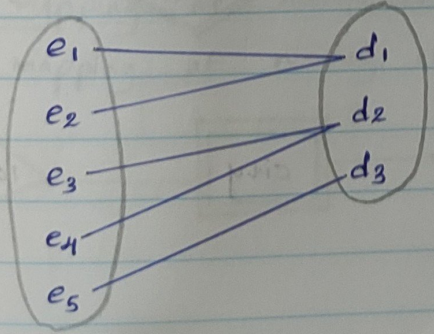
connectivity - (1:m)

cardianility - (1,*) (1,1)

ii) An employee can work for exactly one department but a department can have any number of employees.

emp dep





connectivity - (m:1)

cardianility - (1,1) (1,*) many, any

- minimum 1 employee can work for a department.
- maximum 1 " " " " " "
- A department can have minimum 1 employee
- A department " " maximum many "

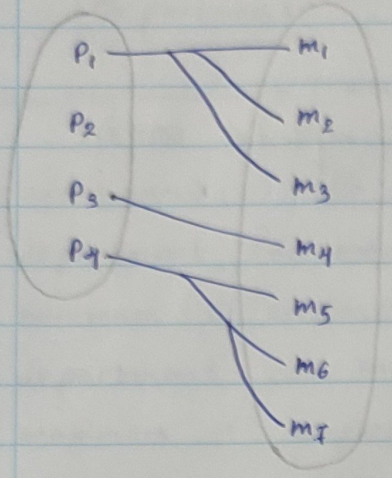
Date: / /

No:

19.07.2022

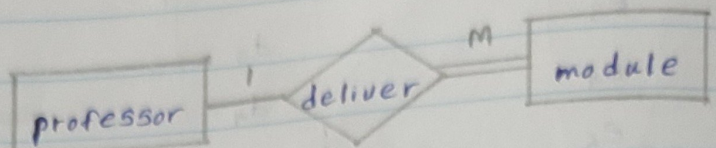
a) A professor can deliver at most 3 modules. Each module is assign with single professor.

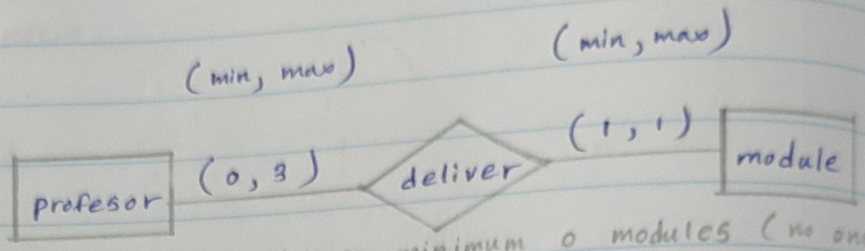
prof. module.



3/4

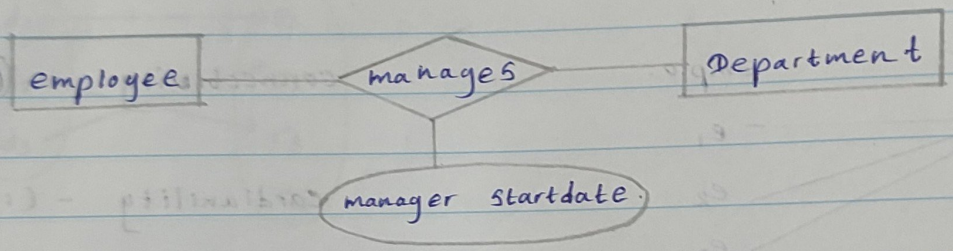
7/7





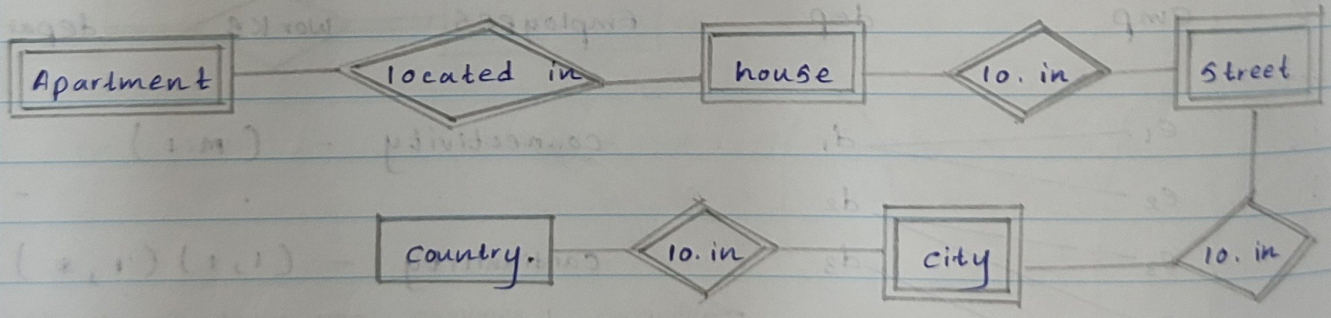
- professor can deliver minimum 0 modules (no one
  professor " " maximum 3 modules
- minimum modules can be delivered by 1 professor
  maximum " " " " " "

a) An employee manages a department. The start date as a manager needs to store in a database.



- Exercise 01

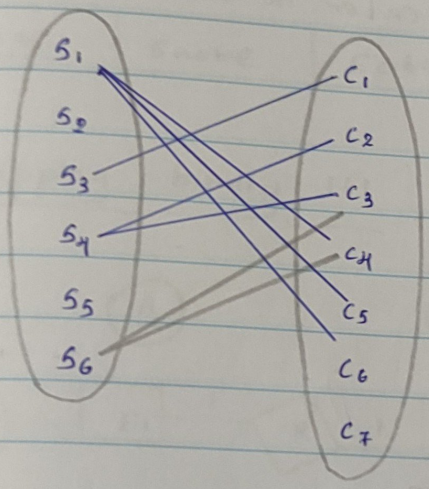
a) An apartment is located in a house in a street in a city in a country.

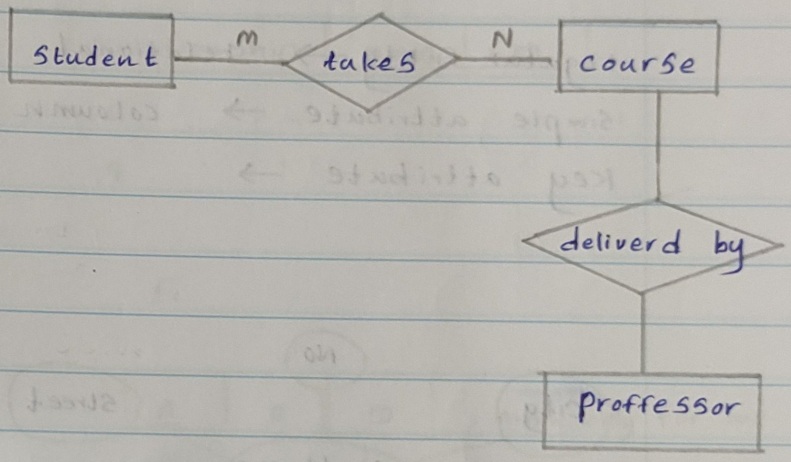


No:

Date: / /

a) Design a database to keep track of what courses a student takes, and information on what professors teach the courses. Given that a student should take many course, a course is also deliverd by many professors. you may decide on attributes.





> Relational Schema.

26.07.2022.

ER - to - Relational mapping Algorithm

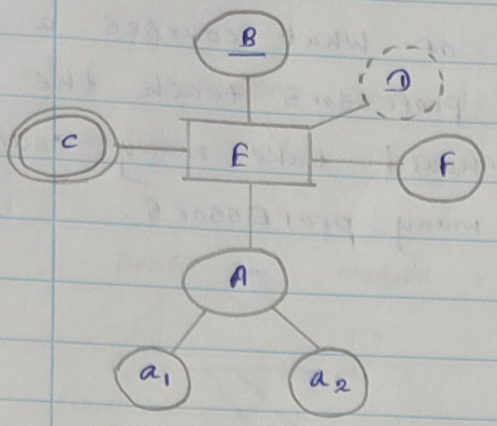
Steps

1) mapping of regular entity types
2) mapping of weak entity types
3) mapping of binary 1:1 Relation Types.
4) mapping of binary 1:N relation types.
5) mapping of binary M:N relation types.
6) mapping of multivalued attributes.
7) mapping of n-ary relationship types.

No :

Date: __/__/__

1) mapping of regular entity types.



E

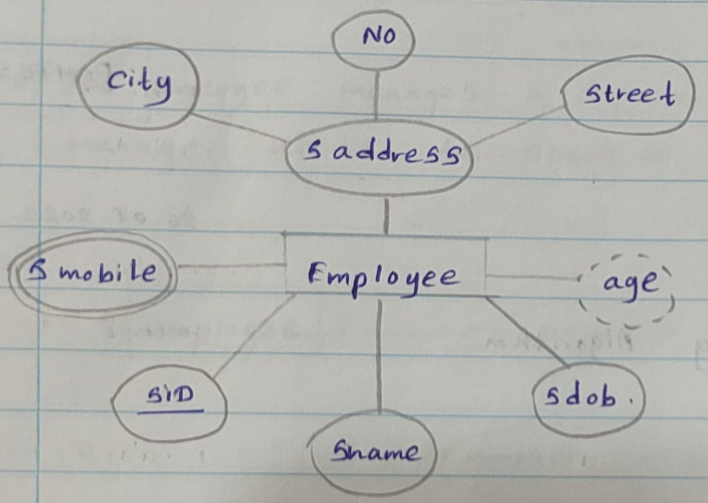
a1	a2	B	F

- derived attribute
- multivalued attribute and composite attri. didn't enter in relational schema

regular Entity → relational

Simple attribute. → coloumn

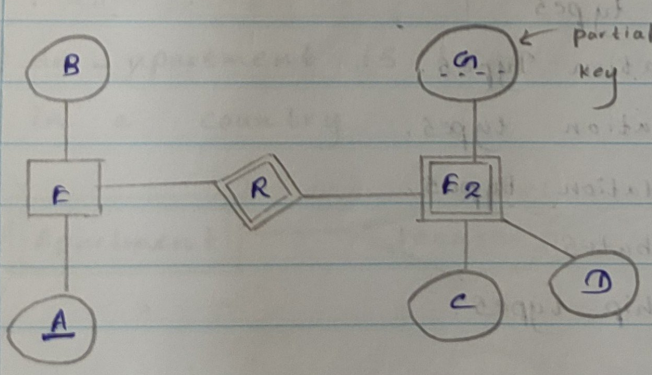
key attribute → P1



E

city	NO	street	SID	Sname	Sdob.

2) mapping of weak entity types.



E

A	B

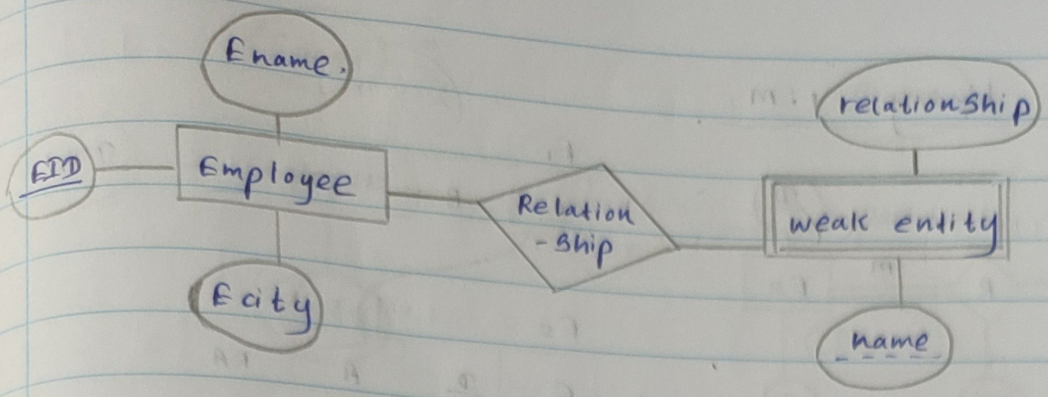
E2

C	D	G	A

partial key :-

- where the values are unique for a certain entity such as name.

Date: __/__/____



Employee

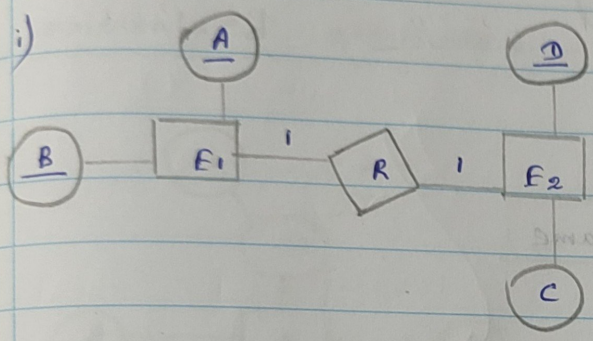
EID	Ename	Ecity.

weak en.

relationship	name	EID

8) mapping binary 1:1

i)



E_1

A	B	D

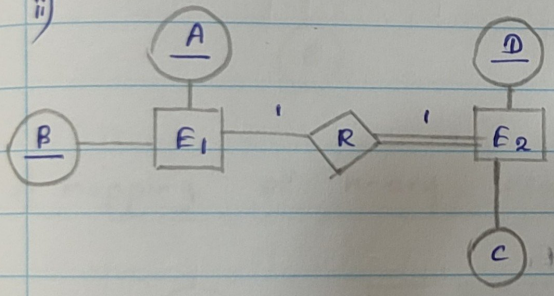
E_2

C	D

Arrow only for one side to represnt 1:1 relationship

we get the foreign from the second side 'D'

ii)



E_1

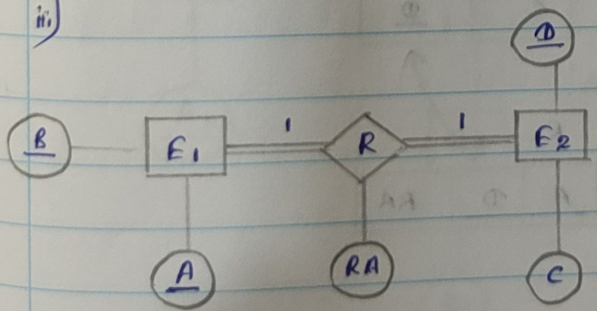
A	B

E_2

C	D	A

to the foreign key doesn't take from the total participants side so 'D' is not the foreign key.

iii)



E_1

A	B

E_2

C	D

R

RA	A	D

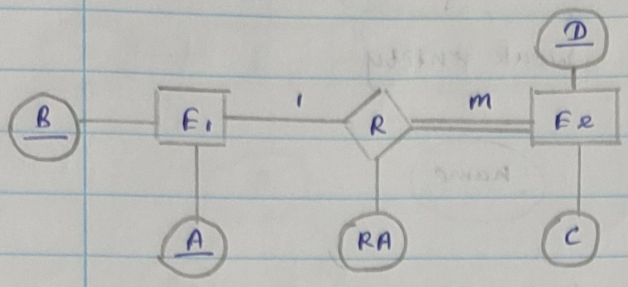
We take the foreign forom the both two sides. 'A', 'D'

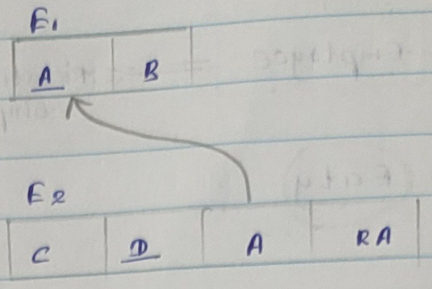
RA - Relationship attribute.

No :

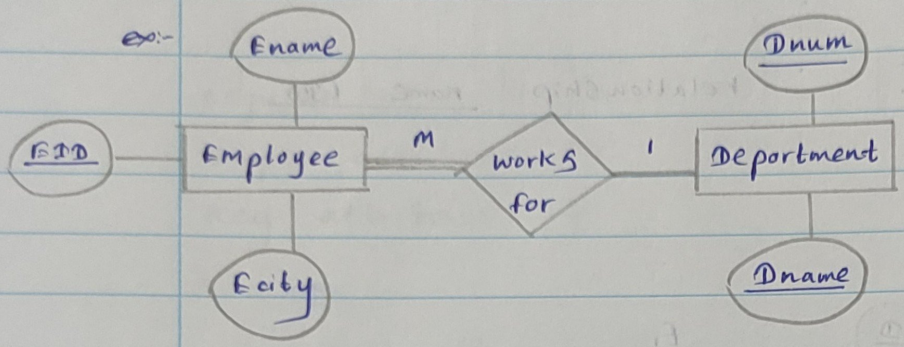
Date: __/__/____

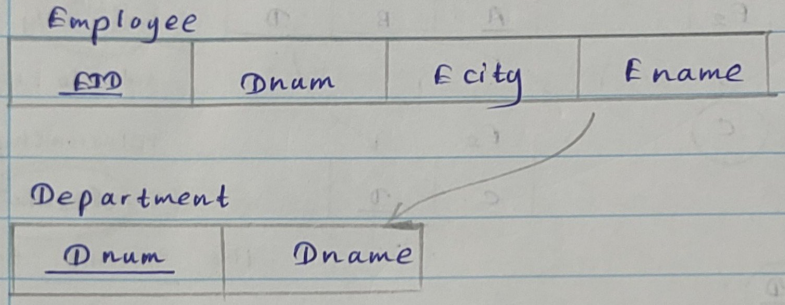
4) mapping of binary 1 : M.



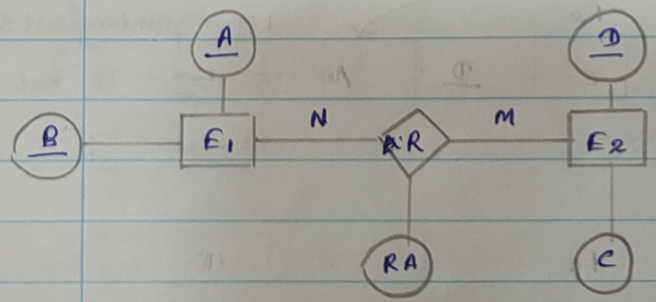


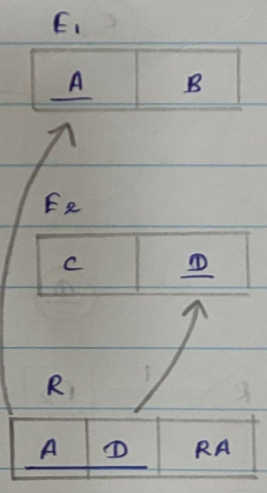
ex:-





5) mapping of binary M : N.

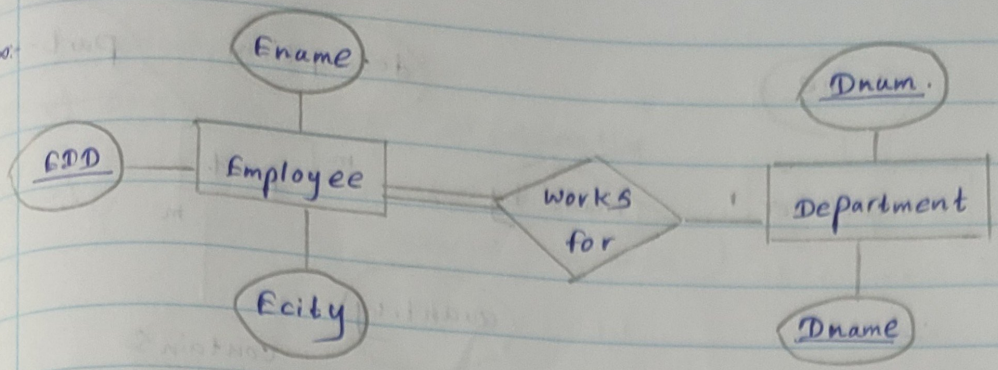




No :

Date: __/__/____

eg:



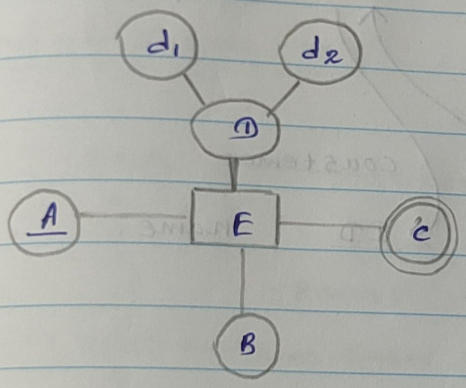
employee

EID	Dnum	Ecity	Ename

Department

Dnum	Dname

6) multivalued attributes.



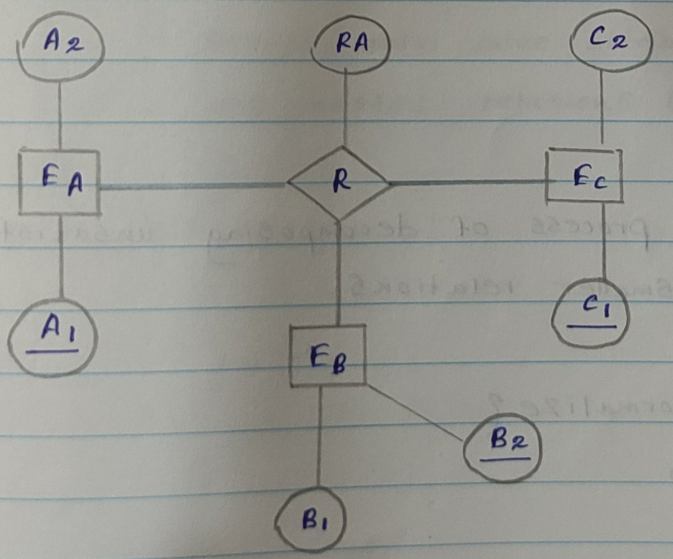
E

A	B	d1	d2

C - E

C	A

7) mapping of n-ary relationships.



E_A

A_2	A_1

E_B

B_1	B_2

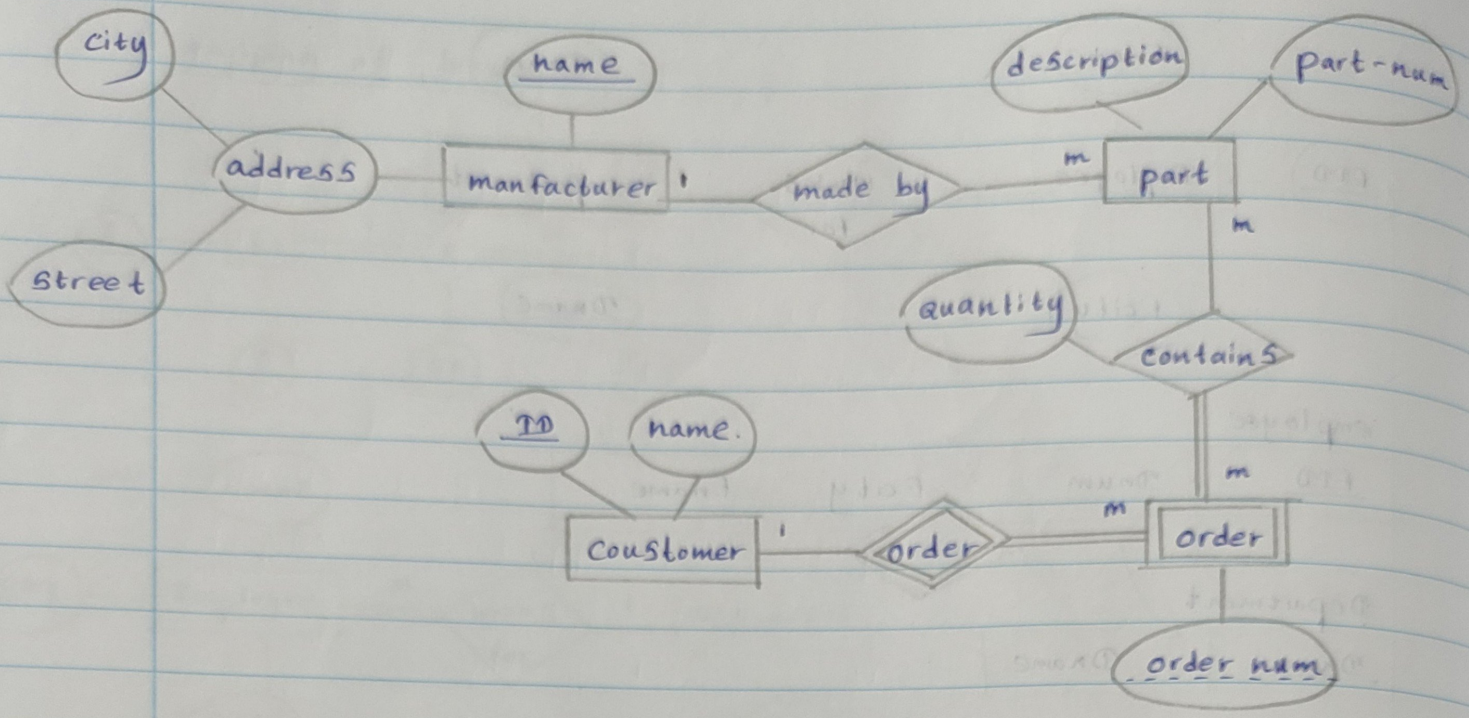
E_C

C_2	C_1

R

A_1	B_2	C_1	RA

No: _____ Date: __/__/____



02.08.2022.

➤ Normalization

* what is normalization?

normalization is the process of decomposing unsatisfactory bad relations into smaller relations.

* why we need to normalize?

It helps to,

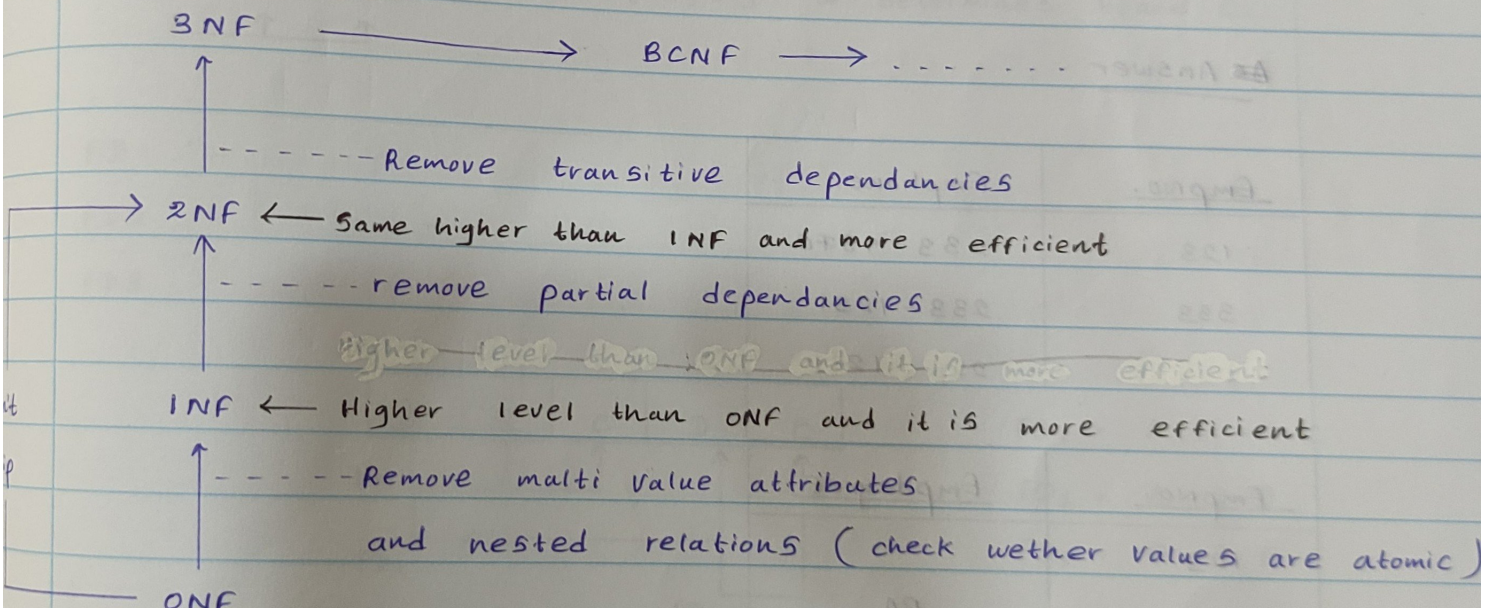
No :

Date: __/__/____

1) reduce duplicates
2) organize data efficiently
3) Reduce data anomalize.
   (Insertion , Delete , update)

- How to apply normalize (NF - normal form)

> Normalization rale.



No: _____ Date: __/__/____

Example 1

Emp NO.	Emp phone	Empdegrees
123	233 - 9876	
333	233 - 1231	BA , BSc. , PHD
679	233 - 1231	BSc. , MSc.

employee table.

Emp NO.	Empphone	Empdegree
123	233 - 9876	
333	233 - 1231	BA
333	233 - 1231	BSc.
333	233 - 1231	PHD
679	233 - 1231	BSc.
679	233 - 1231	MSc.

← wrong answer explaining.

Answer

EmpNO.	Emphone.
123	233 - 9876
333	233 - 1231
679	233 - 1231

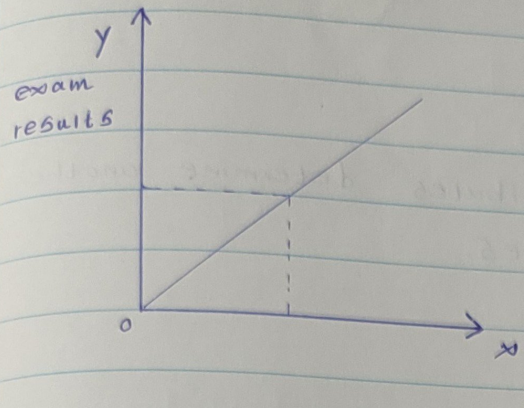
EmpNO.	Empdegree
123	
333	BA
333	BSc.
333	PhD.
679	MSc.
679	MSc.

No: _____

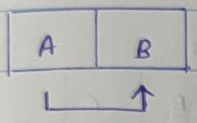
Date: __/__/____

Functional dependants

1NF → 2NF → 3NF



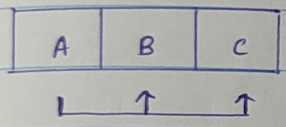
y functionally depends on x



A determines B (A → B)

or

B depends on A

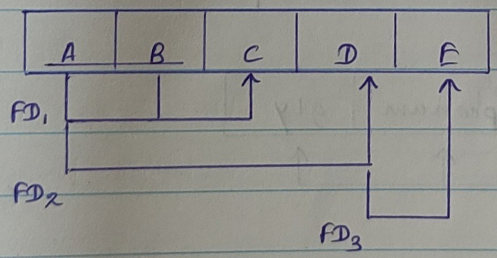


A → { B, C } a determines B and C.

FD 1 -

FD 2 -

FD 3 -



FD_1 = A, B → C

FD_2 = A → D

FD_3 = D → E

Full functional Dipendency

* The complete prime attribute set determines a non-prime attribute or attributes.

Ex:- FD_1

➢ Partial dependancy

* part of the prime attribute set determines a non-prime attribute or attributes.

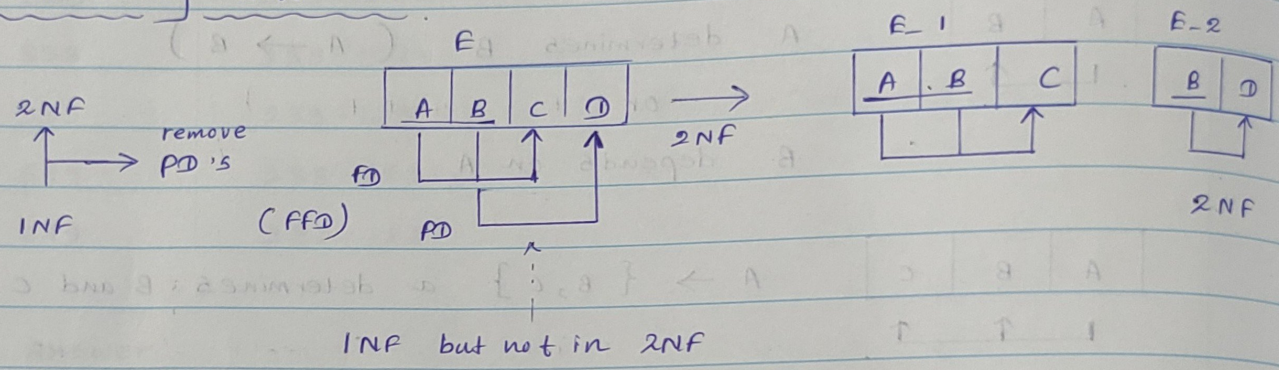
ex: FD_2

➢ Transitive dependancy

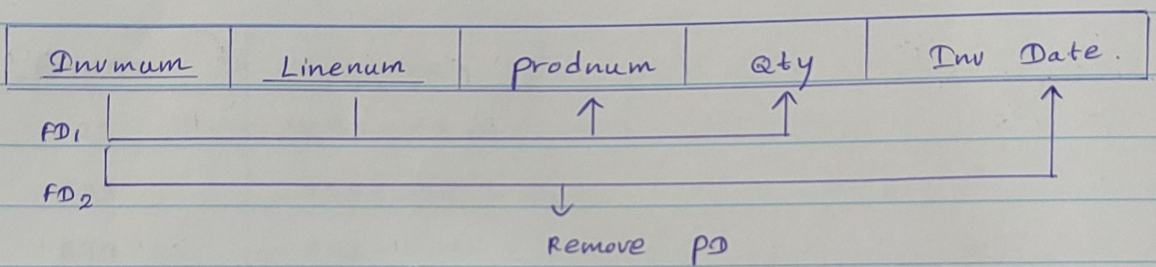
* A non-prime attribute or attributes determine another non-prime attribute and attributes.

ex: FD_3

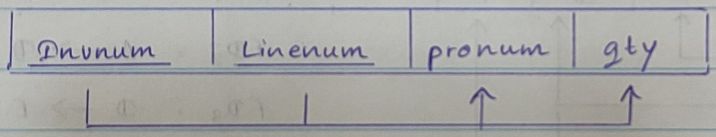
• Normalizing into 2NF.



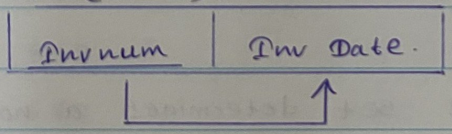
Q)



E_1



2NF (E_2)



• only have 1 primary key in 2NF.
• No partial dependancies.

No :_____ Date: ___/___/_____

a) purchase order relation

po	po NO.	po. DATE	Emp code	supp. NO.	par Supp Name.	part NO.	Part Desc.	part qty.
						P1		
						P		

po-1

po. NO	po. DATE	Emp code.	sup p-no	Supp num

← This table is in 2NF

po-2

po. NO.	part NO.	Part DEsc.	Part qty.

new relation's primary key is — po NO. + part No.

← This Table is in INF

No: _____ Date: __/__/____

1NF ⟶ 2NF (removing partial dependencies)

bring po-2 to 2-NF

po2-1

po no.	part no.	part qty.

po2-2

part no	part. Desc.

bring po-1 to 3-NF

po-1

po. NO.	po. Date	Emp code.	S. NO.

po-1-1

S. NO.	S. name.