Date: 01/09/2021 TOPIC: DBMS. GFG Placement 100 course.

DATABASE Management System.

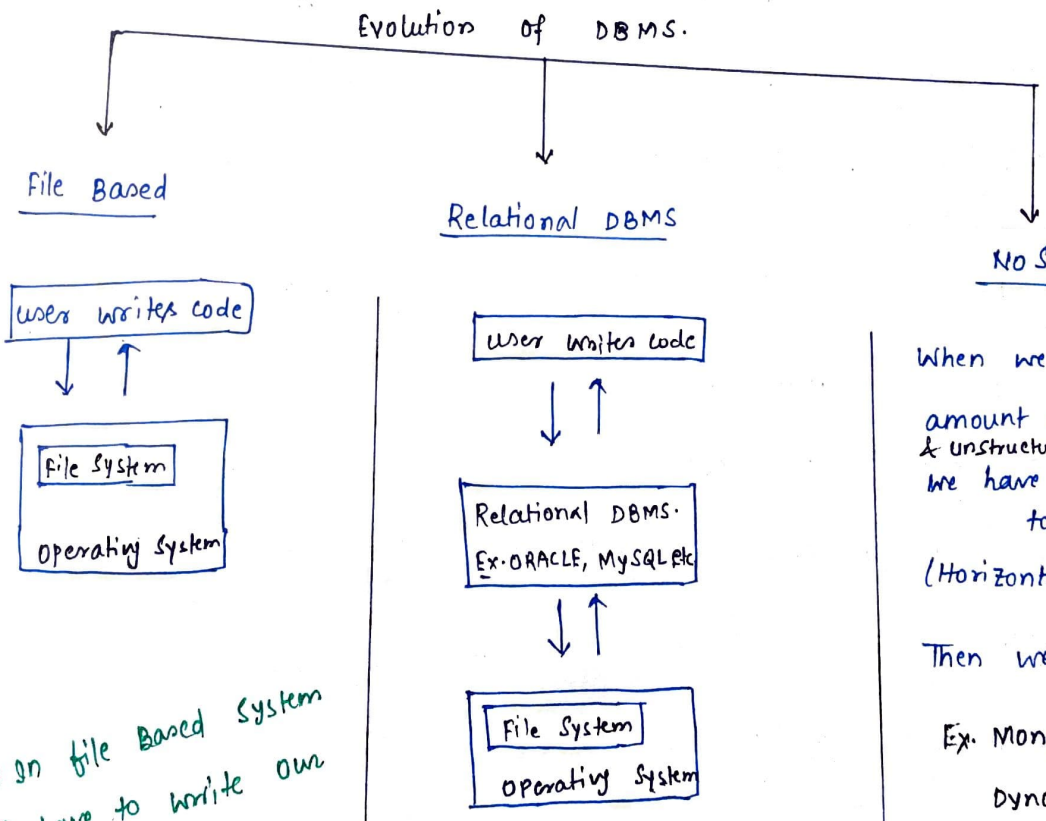
01. DBMS: Introduction.

Most of the websites like Amazon, IRCTC, and many more use database in their backend to store information.

* Database Management system is a collection of software that provides you the quick way to access and modify the data.

Ex. Oracle, MySql, Microsoft sql server and many more.

02. Evolution of DBMS.



* In file Based System we have to write our own DBMS code to access the data.
But via SQL. we don't need to worry just we have to write the Query and it will give us the data.

Uses: Before 1960.

* Relational DBMS provides
  - SQL (Structured Query Lang.).
  - It provides security, concurrency, and many more that is not provided in file Based System.

Uses: 1960 to till now.
(most used and famous DATABases).
i.e. Relational DBMS.

When we have large amount of data and & unstructured data. then we have ~~the~~ scalability to maintain (Horizontal Scalability).

Then we use NoSQL.

Ex. MongoDB,
DynomoDB by Amazon

Uses: Recent Days but not too much.

03. Entity Relationship Model.

Relational DBMS: A Relational DBMS stores data in the form table. that is how we implement Databases.

But before implemention of Databases we must first design the Databases.

* In Databases we have ER (Entity Relationship) Model to Design the Databases.

ER Model.

The ER Model contains 3 things. mainly:

1) Enitity Set.	: Basically Noun.	: Ex: Student, Teacher, ...
2) Relationship Set	: Basically Verb.	: Ex: teaches, gives, join...
3) Attributes.	: Attributes of Entity & Relationship	: Ex: Students ... course joining ... etc.

1) Entity set:

These are basically Nouns.

Ex. (Teacher) teaches a subject.

Here teacher is a noun. so. it is Entity Set.

2) Relationship set:

These are basically verb.

Ex. Teacher (teaches) a subject.

Here teaches is a verb so. it is Relationship Set.

3) Attributes:

Attributes are those who are attributes of Entity and Relationship

Ex: A student's phone no, Address, etc.
course's joining date, ending date etc.] → These are Attributes

04. ER Diagrams:

ER Diagrams are those diagrams which represents the ER Model of the database.

1) Entity Set: It is defined by  or 

Rectangle.

Double Rectangle for weak Entity set.

2) Relationship Set: It is defined by  or 

Diamond.

Double Diamond for weak Relationship set.

ATTRIBUTES

3) Attributes: There are different types of Attributes.

i) Normal Attributes: These Attributes contains single field.

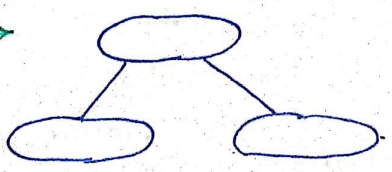
Ex. ~~Depart~~ Department. (Single value).

defined by → 

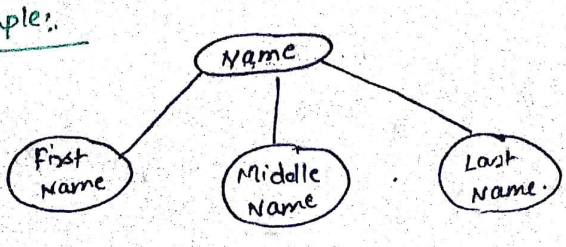
Ex :- Department

ii) Composite Attributes: These Attributes contains multiple field.

Ex. Name → (first name, middle name, Last name).
Address → (Street No, city, District, State, country).

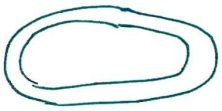
defined by → 

Example:



iii) Multiple Valued Attributes : Attributes those have multiple values for their entity. Ex. An entity may have multiple phone no. multiple Address and many more.

Represented by : Double ellipse,



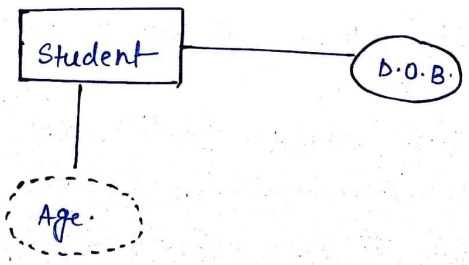
Ex:

iv) Derived Attributes : Attributes those can be derived from a given attributes.

Ex. If a attribute is given Date of Birth of an entity then its age can be derived from that Date of birth attributes.

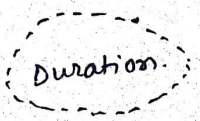
Represented by. Dotted ellipse.

Ex:



Other example: If a course have its Begin date and end as an attributes then we can derive its duration of course.

So, Duration is the derived attributes

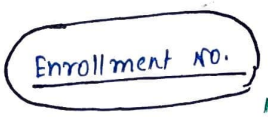


v) Key Attributes: Key attributes are those attributes that defines a particular entity set.

Ex. A student can be defined by it's Enrollment no.

Represented by: Ecllipse with underline.

Example:



↖ This is the key Attributes of a student.

---

05. Relationship sets.

1) Degree of Relationship sets.

i) unary

ii) Binary

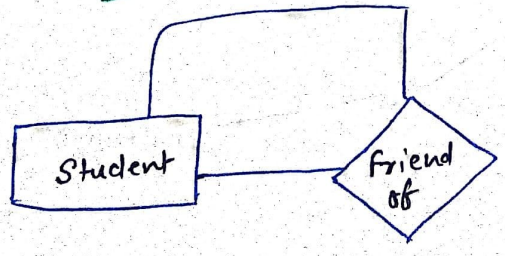
iii) n-Ary

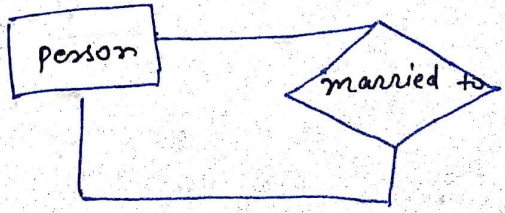
i) Uniary Relationship set: One Entity set related to itself. (Not very common).

Ex. student is a friend of itself.

Ex. A person is marry to a person.

Represented by.





ii) Binary Relationship Sets: (Very common).

These are the (Most common) Relationship Set.

An entity is related with other entity.

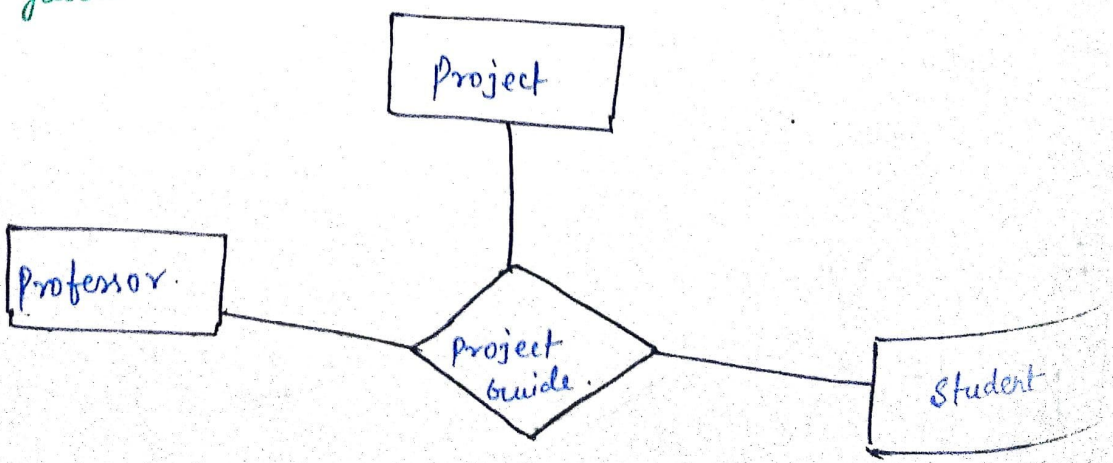
Ex. A student attends the course.

Ex: Supplier supplies items.

iii) n- ary Relationship sets: (Not very common).

Where two or more entity are participating in a relation are called n- ary Relationship sets.

Ex: Proffesor, project and student are connected with Project guide.



2). cardinality.

cardinality: How many entities of one side participates in a relationship.

These are of 3 types.

1) One to one
2) One to many (or many to one).
3) many to many.

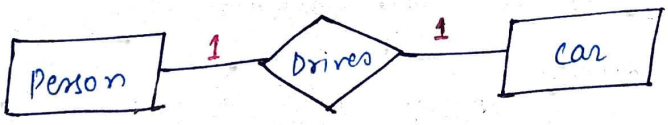
i) **one to one:** When only one entity of each side participates

ii) **one to many:** When one entity of one side participates an many entities of other side participates.

iii) **Many to many:** When many entities of both side participate in the relationship.

Examples of one to one Relationship.

Ex. A person drives a car.

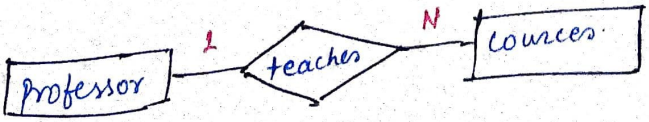


Ex. A person has Addhar card

Or, one Aadhar card is connected with one person

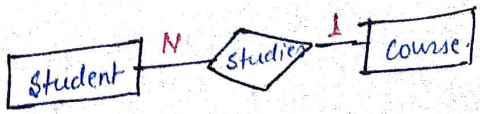
Examples of one to many Relationship.

Ex. 1 professor teaches N courses.



(1 to many)

Ex. N Student studies 1 course



(Many to one).

Both are same.

Example of Many to Many Relationship.

Ex. A student may studies many subjects.

Or, A subject may be studied by many students.

Ex. A customer orders many products.

Or A product is ordered by many customers.

---

06. ER Participation and weak Entity set

ER Participation.

Two types of participation are there.

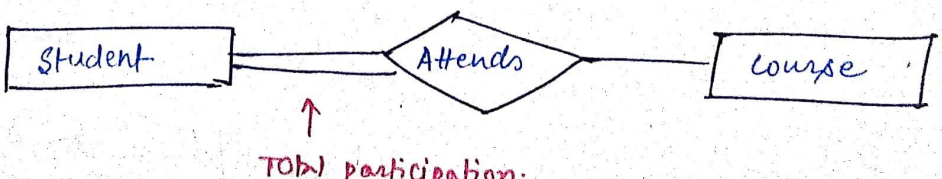
i) Total participation
ii) partial participation.

i) Total participation. Every entity of one side participates in relationship set.

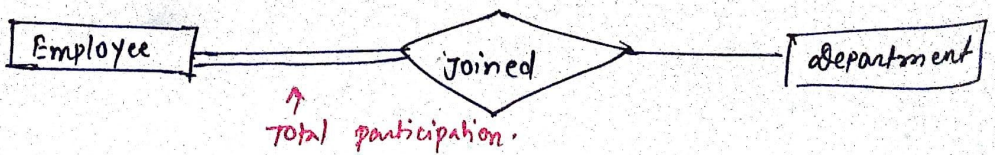
shown using ═══ (double)

Ex: We are assuming that,

A student must attainds one course.



Ex. Every Employee must be joined in a Department.



Weak Entity Sets.

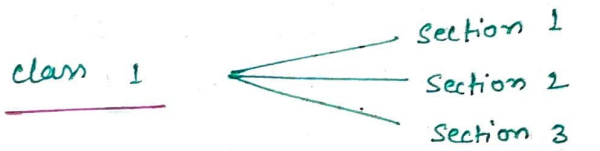
* Do not have their own key. are called weak entity and such sets are called weak Entity sets. Represented by 

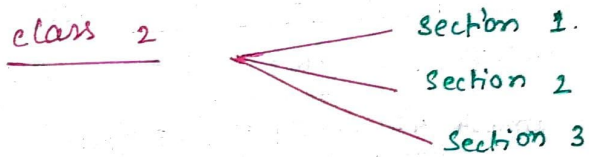
* weak entity sets always have total participation.

Ex. In a class their are different sections.

Every section is defined by a class (unique).

But in every class a section can't defined a particular class.



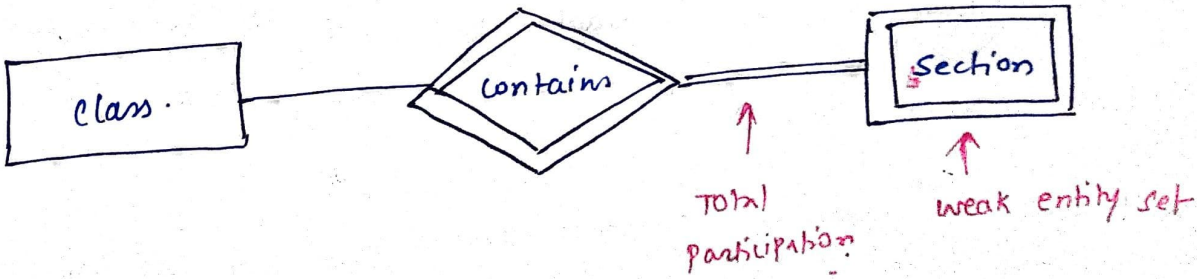


i.e. If we called only section-2.

class 2's section 2 then its a meaning.

But if we call section-2. Then it has no meaning

Which class's ??



Ex. Employee has dependents.

if the employee exits then only its dependents are valid otherwise - dependents are of no use.

Ex. hosts contains log ins.

Login exits only if a host exits.