

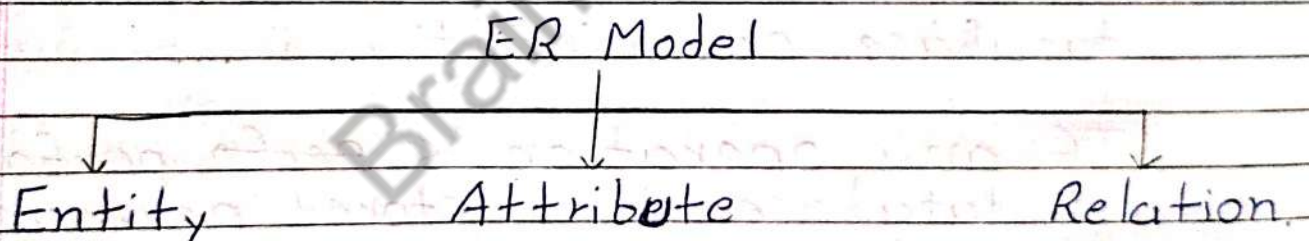
Unit - 2 Data Models

1 Explain ER Model.

ER Model is called Entity and Relationship model.

ER Model is used to define data elements and relationship between the system.

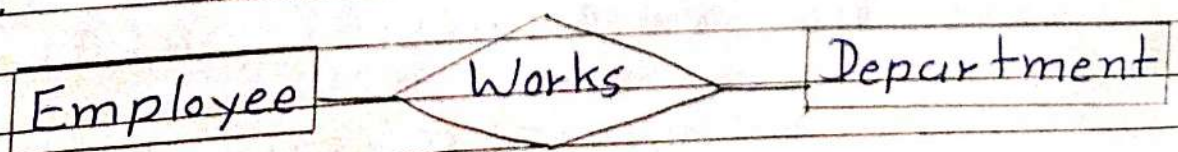
ER Model consist three parts.



(1) Entity : Entity can be any object, class, person or place.

Entity is represented by Rectangles.

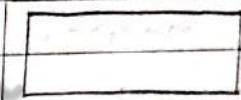
Ex.



Entity has one type - Weak Entity

- Weak Entity: Weak Entity means the entity which depends on another entity.


Weak Entity does not contain any key attributes of its own.

It is represented by 

Ex.



(2) Attribute: Attributes consist properties of Entity.

It is represented by 

There are Four types of Attributes.

- a) Key
- b) Composite
- c) Multivalued
- d) Derived

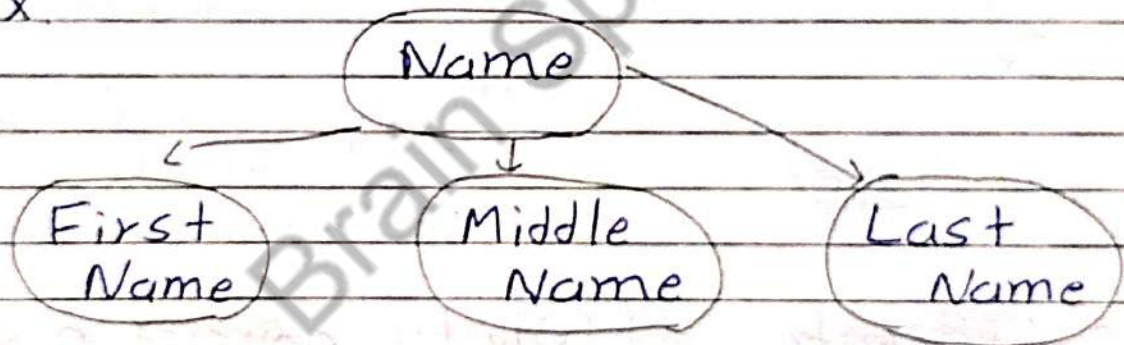
Date _____
Page _____

(a) Key Attributes: It is used to represent unique attributes.

It is represented by attribute name

(b) Composite Attributes: It is used to represent attributes that consist of many attributes.


Ex.




(c) Multivalued Attributes: An attribute that has more than one value, that is known as multivalued attributes.

It is represented by

(d) Derived Attributes: An attribute can be derived by one or more other attributes.

It is represented by 

(3) Relation: This is used to show the relationship with entity.

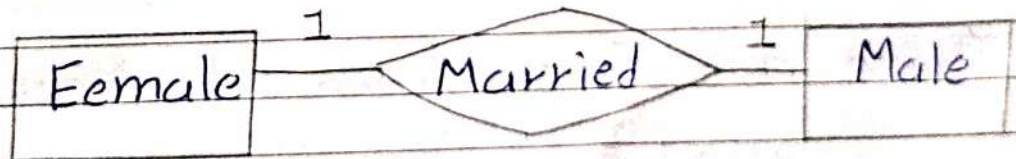
It is represented by 

Relation has four type

- a) One to One
- b) One to Many
- c) Many to One
- d) Many to Many

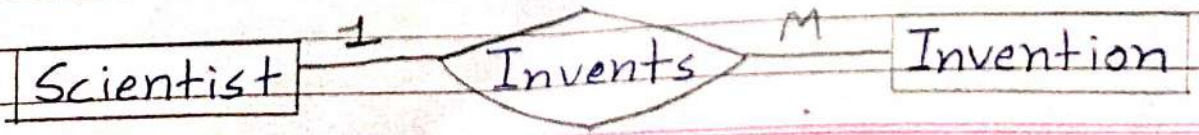
(a) One to One: Left side of entity is connected with only one right side of entity.

Ex.



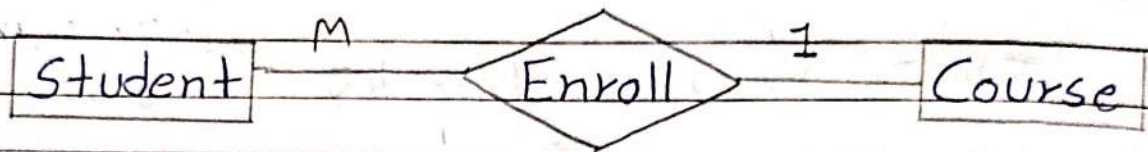
(b) One to Many: Left side of entity is connected with many right side of entity.

Ex.



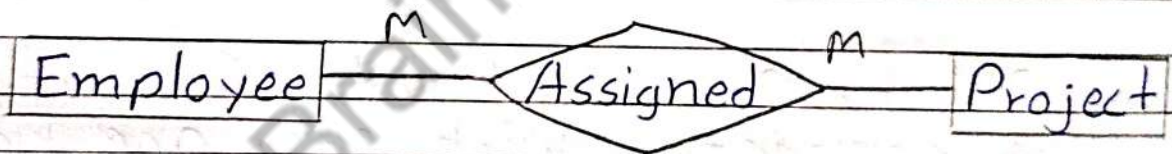
c) Many to One: Many Left side of entity is connected with only one right side of the entity.

Ex.



d) Many to Many: Many Left side of entity is connected with many right side of the entity.

Ex.



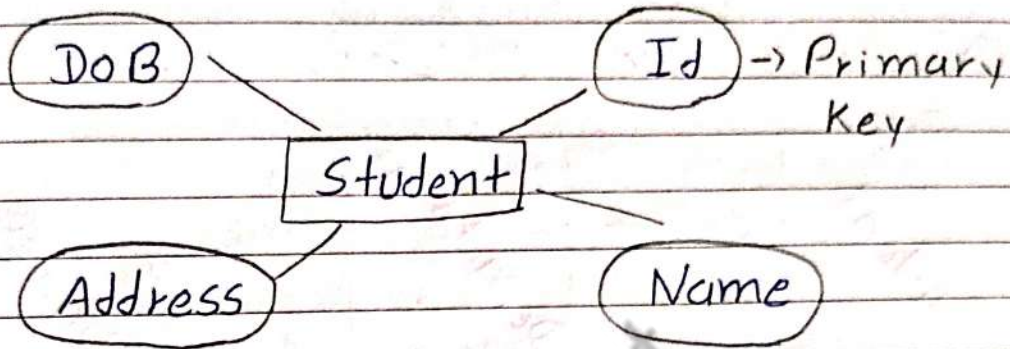
2 Explain Types of Keys.

There are Five types of Keys.

- 1) Primary Key
- 2) Composite Key
- 3) Super Key
- 4) Foreign Key
- 5) Candidate Key.

(1) Primary Key: Primary key is used to identify the entity which is unique.

Ex.



(Fig-1)

(2) Candidate Key: A candidate key is an attribute set that can uniquely identify.

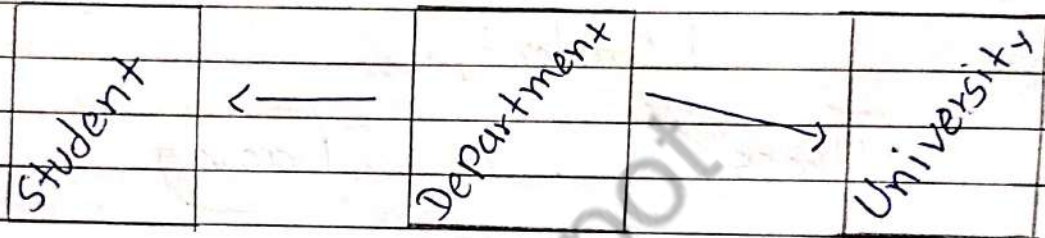
Ex. In Fig-1, DoB, Address and Name are the candidate Key.

(3) Super Key: Super key is an attribute set that can uniquely identify by tuple.

Ex. In Fig. 1 - Id and Name are the super key.

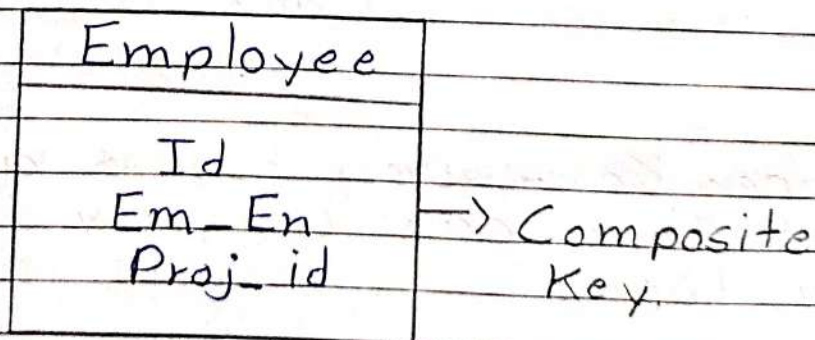
(4) Foreign Key : Foreign Key are the column of table used to point to to the primary key of another table.

Ex.



(5) Composite Key : Whenever primary key consist of more than one attribute this is called Composite Key.

Ex.



3 Explain Relational Algebra with its Operation.

→ Relational Algebra: It is procedural query language.

Relational Algebra gives step by step process to obtain result of query.

These are the Basic Operations of Relational Algebra.

1 Select Operation:

Symbol - σ

Syntax - σ condition (Relation)

Select operation is used to select required tuples of the relations.

2 Projection Operation:

Symbol - π

Syntax - π attribute set (Relation)

Projection Operation is used to select specify attribute set from the data.

~~3 Cartesian~~

3 Cartesian Operation:

Symbol: \times

Syntax: Relation (R_1) \times Relation (R_2)

Cartesian Operation is used to multiply the each tuple of R_1 to each tuple of R_2 .

4 Union Operation:

Symbol: \cup

Syntax: $R_1 \cup R_2$

Union Operation is used to combine the records from two or more query in single result.

5 Intersection Operation:

Symbol: \cap

Syntax: $R_1 \cap R_2$

Intersection Operation is used to select common value from two or more query in single result.

6 Difference Operation:

Syntax: $R_1 - R_2$

Symbol: -

Difference Operation is used to returns all the records from first query.

7 Rename Operation:

Symbol: ρ

Syntax: $\rho_A(x_1, x_2, \dots, x_n)(\text{Relation})$

Rename Operation is used to changed table and Attributes name from data.

This Operation perform three type of Query.

- (i) $\rho_x(CE)$
- (ii) $\rho_{(A_1, A_2, \dots)}(CE)$
- (iii) $\rho_x(A_1, \dots)$

Example :

Student

Branch

No.	Name	CPI	Name	Branch
1	A	8	B	CE
2	B	9	C	IT

1 Select Operation : σ Name = "A" (Student)

1	A	8
---	---	---

2 Projection Operation : π No, Name (Student)

No.	Name
1	A
2	B

3 Cartesian Operation : Student X Branch

No.	S-Name	CPI	B-Name	Branch
1	A	8	CE B	CE
1	A	8	C	IT
2	B	9	B	CE
2	B	9	C	IT

4 Union Operation : π name (student) \cup π name (Branch)

Name
A
B
C

5 Intersection Operation:

$$\Pi \text{ name}(\text{student}) \cap \Pi \text{ name}(\text{Branch})$$

Name
B

6 Difference Operation: Student - Branch

$$= A$$

7 Rename Operation:

- (i) $P_{(S-D)}$ Student - Table name changed
- (ii) $P(N-S, Name-S, C-S)$ Student
- (iii) $P_{(CS-D)}$ student
 $(N-S, Name-S, C-S)$

4 Explain Join Operation with its types:

Join Operation is used to combine related tuples from different relations.

There are three types of Join Operation.

(1) Natural Join

(2) Outer Join

(3) Equiv Join

(1) Natural Join:

Symbol: \bowtie

Syntax: $R_1 \bowtie R_2$

For Natural Join, we have to perform these three steps.

(i) Perform Cartesian Operation

(ii) Remove Inconsistent tuples

(iii) Remove Duplicate Attributes.

(2) Outer Join:

There are three types of Outer Join:

- ci) Left Outer Join
- cii) Right Outer Join
- ciii) Full Outer Join

ci) Left Outer Join: Display all the tuple in left relation even if it does not match in the right side.

cii) Right Outer Join: Display all the tuple in right relation even if it does not match in the left side.

ciii) Full Outer Join: Display all the values of tuple.

Example:

Student Result

R.no	Name	Branch	R.no	SPI
1	A	CE	1	8
2	B	EE	2 3	9

1 Natural Join: Student \bowtie Result

(i)

S-R-no	Name	Branch	R-R-no	SPI
1	A	CE	1	8
1	A	CE	2	9
2	B	EE	1	8
2	B	EE	2	9

(ii)

S-R-no	Name	Branch	R-R-no	SPI
1	A	CE	1	8
2	B	EE	2	9

(iii)

R-no	Name	Branch	SPI
1	A	CE	8
2	B	EE	9

2 Outer Join:

(i) Left Outer Join: Student \ltimes Result

R-no	Name	Branch	SPI
1	A	CE	8
2	B	EE	Null

(ii) Right Outer Join: Student \rtimes Result

R-no	Name	Branch	SPI
1	A	CE	8
3	Null	Null	9

ciii) Full Outer Join: Student XI Result

R-no	Name	Branch	SPI
1	A	CE	8
2	B	EE	Null
3	Null	Null	9

5 Explain Integrity Constraints with its types.

Integrity Constraints is one types of set of rules.

Integrity Constraints is used to maintain the quality of information.

There are four type of Integrity Constraints.

- (a) Domain Constraint
- (b) Entity - Integrity Constraint
- (c) Referential Integrity Constraint
- (d) Key Constraint

(a) Domain Constraint:

Domain Constraint is use to allow only vaild set of values for an attribute.

Ex.

ID	Name	Age
1	A	10
2	B	11
3	C	(B)

Age is number not a ch.

(b) Entity - Integrity Constraint:

This constraint is used to primary key value can not be null.

In this constraint, Primary key is not null value.

Ex.

* ID	Name	Age
1	A	10
2	B	11
○	C	20

ID can not be null.

(c) Referential - Integrity Constraint:

In this constraint one table contain Foreign key and second table contain Primary key.

Foreign key in one table refers to primary key in second table, then every value in table one must be available in table two.

Ex,

Table-1			Table-2	
F.K	E_id	Name	D_id	*
	1	A	24	A
	2	B	25	B
	3	C	26	C

25 is not valid value it is null value or 27 value.

(d) Key Constraint:

In this constraint, Primary key is always constant unique value.

Ex,

*Id	Name	Age
1	A	10
2	B	11
①	C	21

Primary key must be unique.