MODULE 2

RELATIONAL MODEL

CO – Students will be able to model and design solutions for efficiently representing and querying data using relational model

Prepared By Mr. EBIN PM, AP

RELATIONAL MODEL CONCEPTS

- Relational data model is the primary data model, which is used widely around the world for data storage and processing.
- This model is simple and it has all the properties and capabilities required to process data with storage efficiency.

❖**CONCEPTS**

➢**Tables:** In relational data model, relations are saved in the format of Tables. This format stores the relation among entities. A table has rows and columns, where rows represent records and columns represent the attributes.

Prepared By Mr. EBIN PM, AP,CU EDULINE 2

- **Tuple:** A single row of a table, which contains a single record for that relation is called a tuple.
- **Relation instance:** A finite set of tuples in the relational database system represents relation instance. Relation instances do not have duplicate tuples.
- **Relation schema:** A relation schema describes the relation name (table name), attributes, and their names.
- **Relation key:** Each row has one or more attributes, known as relation key, which can identify the row in the relation (table) uniquely.
- **Attribute domain:** Every attribute has some predefined value scope, known as attribute domain.

Relation Name → STUDENT

Attributes

Tuples

Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

Fig: The attributes and tuples of a relation STUDENT

CONSTRAINTS (RESTRICTIONS)

- Every relation has some conditions that must hold for it to be a valid relation.
- These conditions are called Relational Integrity Constraints.
- Integrity means Reliability and accuracy of data.
- There are three main integrity constraints:

1. **Key constraints (Entity Constraints)**
2. **Domain constraints**
3. **Referential integrity constraints**

Prepared By Mr. EBIN PM, AP, CU EDULINE 5

1. Key Constraints

- There must be at least one minimal subset of attributes in the relation, which can identify a tuple uniquely.
- This minimal subset of attributes is called key for that relation.
- If there are more than one such minimal subset, these are called candidate keys.

Key constraints force that:

- In a relation with a key attribute, no two tuples can have identical values for key attributes.
- A key attribute cannot have NULL values.

Prepared By Mr. EBIN PM, AP , CU EDULINE 6

- The two properties of keys are unique identification and non-redundancy.
- Key constraints are also referred to as Entity Constraints.
- The entity integrity constraint states that no primary key value can be NULL. This is because the primary key value is used to identify individual tuples in a relation.

2. Domain Constraints

- Attributes have specific values in real-world scenario. For example, age can only be a positive integer.
- The same constraints have been tried to employ on the attributes of a relation.

Prepared By Mr. EBIN PM, AP ,CU EDULINE 7

- Every attribute is bound to have a specific range of values.
- For example, age cannot be less than zero and telephone numbers cannot contain a digit outside 0-9.

3. Referential Integrity Constraints

- Referential integrity constraints work on the concept of Foreign Keys.
- A foreign key is a key attribute of a relation that can be referred in other relation.
- Referential integrity constraint states that if a relation refers to a key attribute of a different or same relation, then that key element must exist.

Prepared By Mr. EBIN PM, AP ,CU EDULINE 8

- Referential integrity is typically enforced with a Primary Key (PK) and Foreign Key (FK) combination.
- A foreign Key is a column or combination of columns in one table (referred to as the child table) that takes its values from the PK in another table (referred to as the parent table).
- In order for referential integrity to be maintained, the FK in the child table can only accept values that exist in the PK of parent table.
- The primary objective of referential integrity is to prevent orphans; i.e.; records in the child table that cannot be related to a record in the parent table.

Prepared By Mr. EBIN PM, AP ,CU EDULINE 9

- Enforcing referential integrity means the relationship between the tables must be preserved when records are added (INSERT), changed (UPDATE) or deleted (DELETE).
- If we delete a tuple that is a target of a foreign key reference then three explicit possibilities exist to maintain database integrity.

✓ All tuples that contain references to the deleted tuple should also be deleted; it may cause the deletion of other tuples which result in domino or cascading deletion. (Deletion one after another).

✓ Only tuple that are not referenced by any other tuple can be deleted. A tuple referenced by any other tuple in the database cannot be deleted.

Prepared By Mr. EBIN PM, AP ,CU EDULINE 10

✓The tuple is deleted to avoid domino effect, foreign key attributes of all referencing tuples are set to null.

Foreign key (sid) — Primary key (sid)

cid	grade	sid
Carnatic101	C	53831
Reggae203	B	53832
Topology112	A	53650
History105	B	53666

sid	name	login	age	gpa
50000	Dave	dave@cs	19	3.3
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@ee	18	3.2
53650	Smith	smith@math	19	3.8
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

Enrolled (Referencing relation) Students (Referenced relation)

Fig: Referential integrity

Prepared By Mr. EBIN PM, AP ,CU EDULINE 11

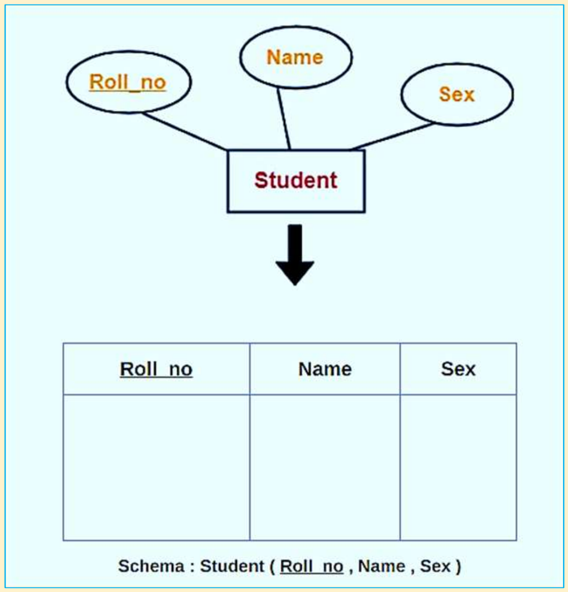
Convert ER Diagram to Relational Database

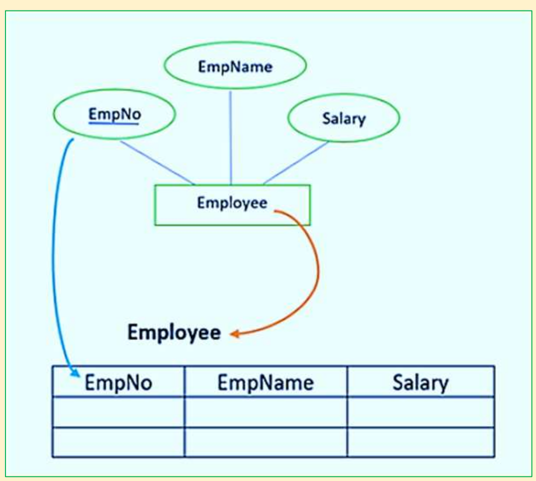
- Conversion of an E-R diagram into a relational model is required because E-R diagrams represent the conceptual level of the database design while the relational model represents the logical level. The relational models can be easily implemented using RDBMS like Oracle, MySQL, etc.
- In general conversion of E-R diagram into a relational model involves the following:

1. Mapping of an entity set into relation (tables) of the database.
2. The attributes of a table include the attributes of an entity
3. The key attribute of an entity becomes the primary key of the relation

Prepared By Mr. EBIN PM, AP EDULINE 12

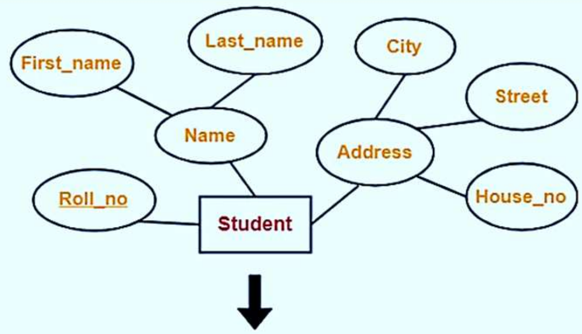
❖Rule-1: For Strong Entity Set With Only Simple Attributes





- Here we map entity set into a relation Employee and attributes of an entity set will become the attributes inside the table.
- The key attribute will become the primary key of the table.

❖Rule-2:
For Strong Entity Set With Composite Attributes

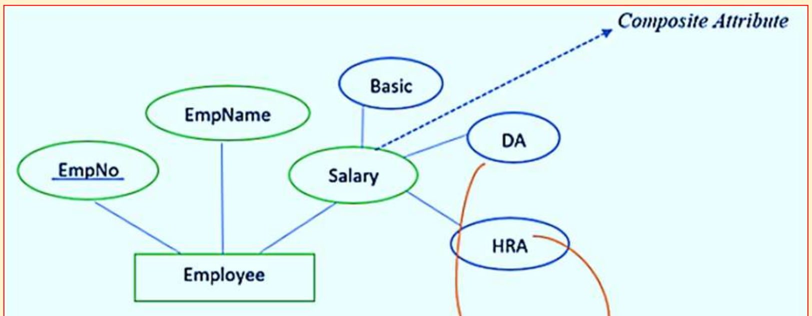


Roll_no	First_name	Last_name	House_no	Street	City

Schema : Student (Roll_no , First_name , Last_name , House_no , Street , City)

Prepared By Mr. EBIN PM, AP EDULINE 15

- While converting an E-R diagram consisting of a composite attribute we do not include the composite attribute in the relational model.

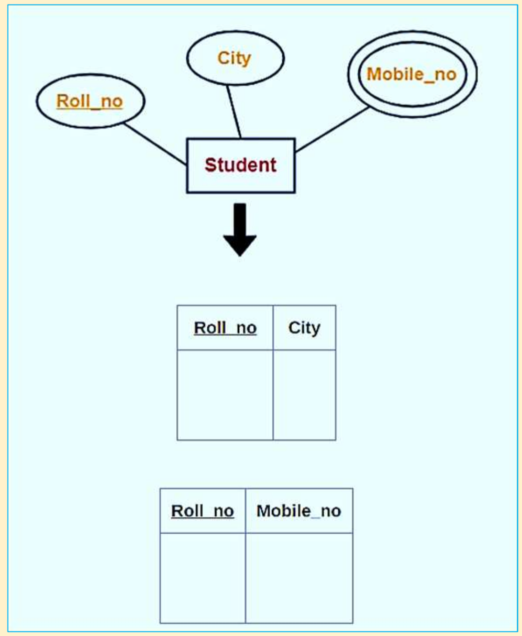


Employee

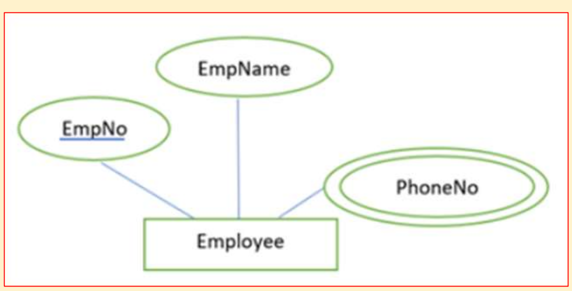
EmpNo	EmpName	Basic	DA	HRA

Prepared By Mr. EBIN PM, AP EDULINE 16

❖**Rule-3: For Strong Entity Set With Multi Valued Attributes**



- If an entity contains a multivalued attribute, we split the attributes into two relations in the relational model.
- One with key attribute and all simple attributes and other with key attribute and all multivalued attributes.



Employee

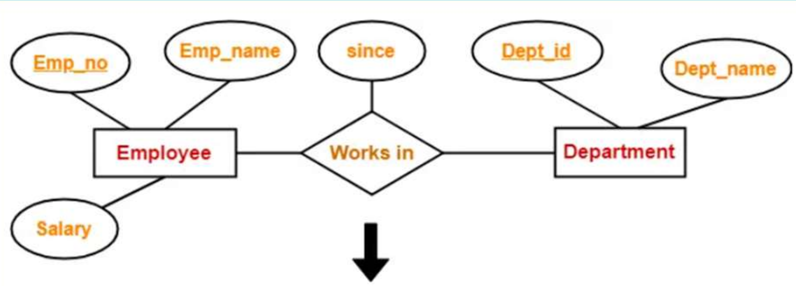
EmpNo	EmpName	PhoneNo
1	A	9821
1	A	9780
2	B	1234
….		

- If we include the PhoneNo in the table with all other attributes, then for a single-valued tuple we may have multiple entries as shown in the table above.
- However, to avoid duplicate values in the table, we split the attributes into two different relations as shown in the figure below.

EmpNo	EmpName

EmpNo	PhoneNo

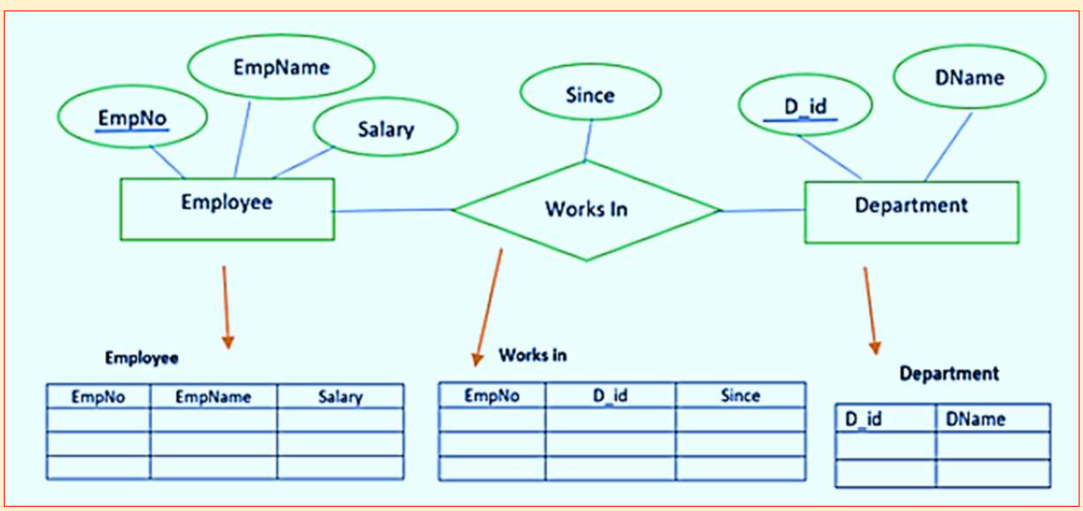
❖Rule-4: Translating Relationship Set into a Table



Emp_no	Dept_id	since

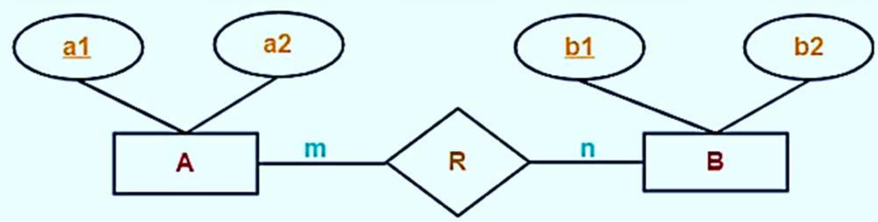
Schema : Works in (Emp_no , Dept_id , since)

- we can also map a relationship set into a relation.
- The attribute of such a relation includes key attributes of the participating relations.
- The attributes are will become a foreign key.
- For example, in the figure given below, there are two entity sets Employee and Department.
- These entity sets are participating in a relationship works in.
- The relationship set is converted into relation with attributes EmpNo from Employee relation, D_id from Department relation and Since, the attribute of the relationship set itself.



❖Rule-5: For Binary Relationships With Cardinality Ratios

1. For Binary Relationship With Cardinality Ratio m:n

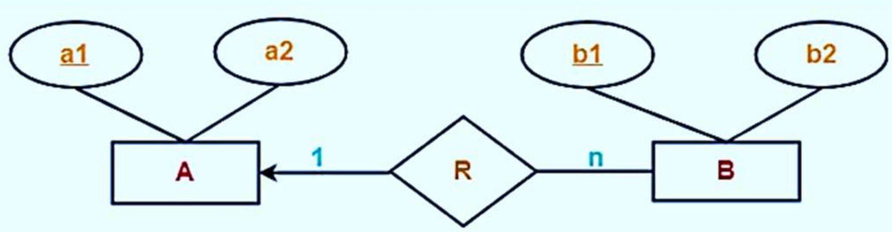


Here, three tables will be required-

1. A (a1 , a2)
2. R (a1 , b1)
3. B (b1 , b2)

2. For Binary Relationship With Cardinality Ratio 1:n

- Here, combined table will be drawn for the entity set B and relationship set R.

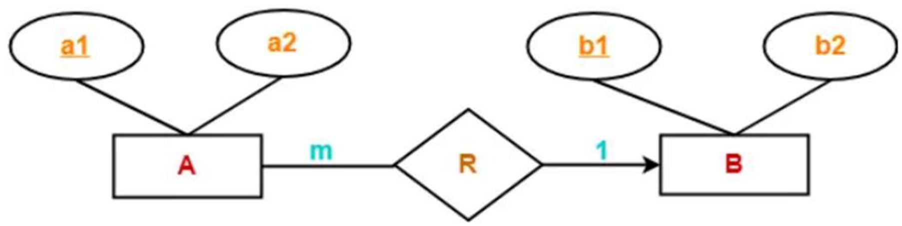


Here, two tables will be required-

1. A (a1 , a2)
2. BR (a1 , b1 , b2)

3. For Binary Relationship With Cardinality Ratio m:1

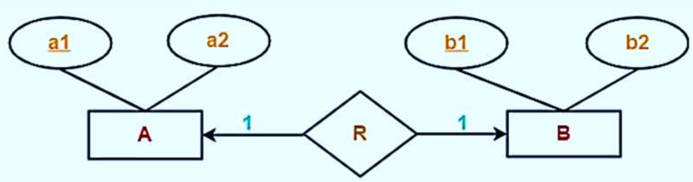
- Here, combined table will be drawn for the entity set A and relationship set R.



Here, two tables will be required-

1. AR (a1 , a2 , b1)
2. B (b1 , b2)

4. For Binary Relationship With Cardinality Ratio 1:1



Here, two tables will be required. Either combine 'R' with 'A' or 'B'

Way-01:

1. AR (a1 , a2 , b1)
2. B (b1 , b2)

Way-02:

1. A (a1 , a2)
2. BR (a1 , b1 , b2)

❖Rules to Remember

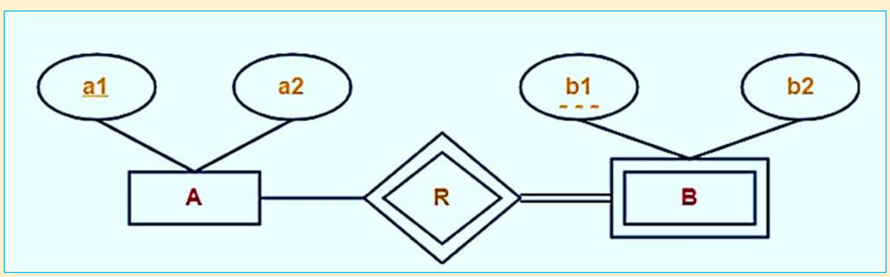
- While determining the minimum number of tables required for binary relationships with given cardinality ratios, following thumb rules must be kept in mind-

➢For binary relationship with cardinality ration m : n , separate and individual tables will be drawn for each entity set and relationship.

➢For binary relationship with cardinality ratio either m : 1 or 1 : n , always remember "many side will consume the relationship" i.e. a combined table will be drawn for many side entity set and relationship set.

➢For binary relationship with cardinality ratio 1 : 1 , two tables will be required. You can combine the relationship set with any one of the entity sets.

❖Rule-5: For Binary Relationship With Weak Entity Set



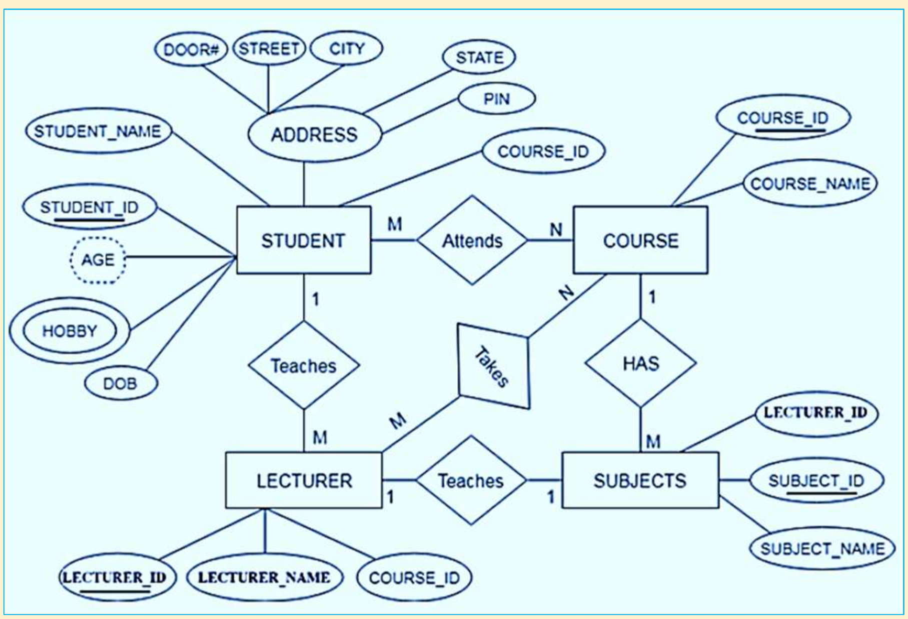
- Weak entity set always appears in association with identifying relationship with total participation constraint.
- Here, two tables will be required

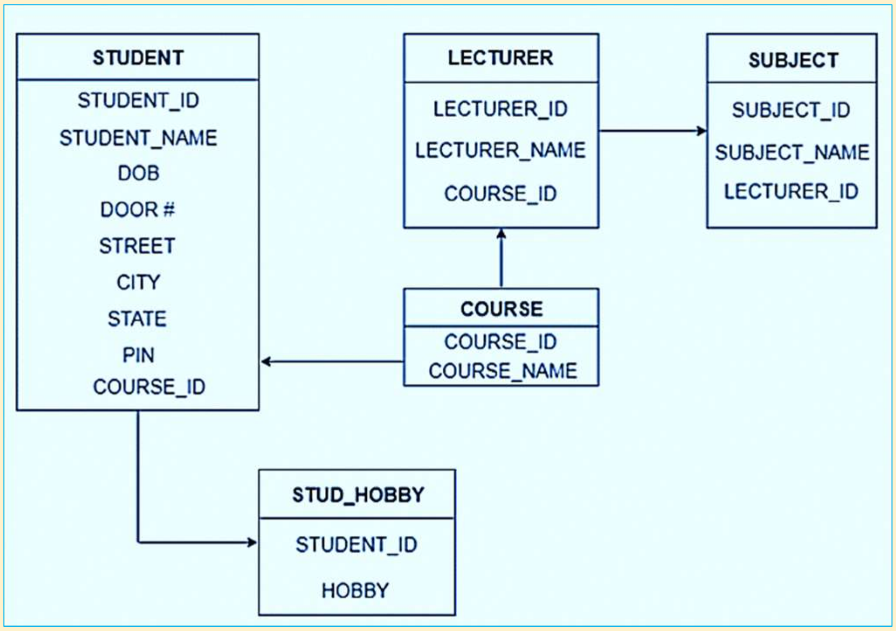
A (a1 , a2)

BR (a1 , b1 , b2)

❖**points for converting the ER diagram to the table:**

1. Entity type becomes a table.
2. All single-valued attribute becomes a column for the table.
3. A key attribute of the entity type represented by the primary key.
4. The multivalued attribute is represented by a separate table.
5. Composite attribute represented by components.
6. Derived attributes are not considered in the table.





RELATIONAL ALGEBRA

- Relational database systems are expected to be equipped with a query language that can assist its users to query the database instances.
- There are two kinds of query languages: relational algebra and relational calculus.

❖**Relational Algebra**

- Relational algebra is a procedural query language, which takes instances of relations as input and yields instances of relations as output.
- It uses operators to perform queries. An operator can be either unary or binary.

- They accept relations as their input and yield relations as their output.
- Relational algebra is performed recursively on a relation and intermediate results are also considered relations.

The fundamental operations of relational algebra are as follows:

- **Select**
- **Project**
- **Union**
- **Set different**
- **Cartesian product**
- **Rename**

1. Select Operation (σ)

- It selects tuples that satisfy the given predicate from a relation.

Notation: **σp(r)**

- Where σ stands for selection predicate and r stands for relation. p is prepositional logic formula which may use connectors like and, or, and not.
- These terms may use relational operators like: =, ≠, ≥, <, >, ≤

Example

loan_number	*branch_name*	*amount*
L-11	Round Hill	900
L-14	Downtown	1500
L-15	Perryridge	1500
L-16	Perryridge	1300
L-17	Downtown	1000
L-23	Redwood	2000
L-93	Mianus	500

Fig: Loan Relation

1. Select tuples of the loan relation where branch is “perryridge”?

   σ branch_name=“perryridge” (Loan)

2. Find all tuples in which the amount lent is more than $1200?

   σ amount > 1200 (Loan)

3. Find all tuples pertaining to loans of more than $1200 made by perryridge branch?

   σ branch_name=“perryridge” ^ amount > 1200 (Loan)

2. Project Operation (Π)

- It projects column(s) that satisfy a given predicate.

Notation: $\Pi_{A1, A2, An}(r)$

- Where A1, A2, An are attribute names of relation r. Duplicate rows are automatically eliminated as relation is a set.

For example:

1. Π loan_number, amount (Loan)

- The above query lists all loan number and amount of the loan relation. The output is given below:

Prepared By Mr. EBIN PM, AP EDULINE 37

loan_number	*amount*
L-11	900
L-14	1500
L-15	1500
L-16	1300
L-17	1000
L-23	2000
L-93	500

Prepared By Mr. EBIN PM, AP EDULINE 38

- Consider the following relation; find the customers who live in Harrison?

customer_name	customer_street	customer_city
Adams	Spring	Pittsfield
Brooks	Senator	Brooklyn
Curry	North	Rye
Glenn	Sand Hill	Woodside
Green	Walnut	Stamford
Hayes	Main	Harrison
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Smith	North	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

Fig: Customer relation

Π customer_name (σ customer_city = "Harrison" (Customer))

3. Union Operation (∪)

- It performs binary union between two given relations and is defined as:

  r ∪ s = { t | t ∈ r or t ∈ s}

  Notion: **r U s**

- Where r and s are either database relations or relation result set (temporary relation).
- For a union operation to be valid, the following conditions must hold:

➢ r and s must have the same number of attributes.

➢ Attribute domains must be compatible.

➢ Duplicate tuples are automatically eliminated.

Eg : Π author (Books) ∪ Π author (Articles)

Output: Projects the names of the authors who have either written a book or an article or both.

4. Set Difference (−)

- The result of set difference operation is tuples, which are present in one relation but are not in the second relation.

  Notation: **r − s**

Eg : Finds all the tuples that are present in r but not in s.

Πauthor(Books) − Πauthor(Articles)

Output: Provides the name of authors who have written books but not articles.

5. Cartesian Product (X)

- Combines information of two different relations into one.

  Notation: **r X s**

- Where r and s are relations and their output will be defined as:

$$r \text{ X } s = \{ q\,t \mid q \in r \text{ and } t \in s\}$$

Eg: Πauthor = ‘eduline'(Books X Articles)

Output: Yields a relation, which shows all the books and articles written by eduline.

6. Rename Operation (ρ)

- The results of relational algebra are also relations but without any name.
- The rename operation allows us to rename the output relation.
- rename operation is denoted with small Greek letter rho ρ.

  Notation: **ρ x (E)**
- Where the result of expression E is saved with name of x.

A

K	X	Y
1	A	2
2	B	4
3	C	6

B

K	X	Y
1	A	2
4	D	8
5	E	10

A UNION B

K	X	Y
1	A	2
2	B	4
3	C	6
4	D	8
5	E	10

A INTERSECT B

K	X	Y
1	A	2

A-B

K	X	Y
2	B	4
3	C	6

B-A

K	X	Y
4	D	8
5	E	10

A

K	X	Y
1	A	2
2	B	4
3	C	6

B

K	X	Y
1	A	2
4	D	8
5	E	10

A×B

AK	AX	AY	BK	BX	BY
1	A	2	1	A	2
1	A	2	4	D	8
1	A	2	5	E	10
2	B	4	1	A	2
2	B	4	4	D	8
2	B	4	5	E	10
3	C	6	1	A	2
3	C	6	4	D	8
3	C	6	5	E	10

6. Join operation ⋈

- If the join criterion is based on equality of column values, the result is called an **equijoin.**
- A **natural join** is an equijoin with redundant columns removed.
- The following tables D and E are joined based on the equality of K in both tables.

D

K	X	Y
1	A	2
2	B	4
3	C	6
4	D	8
5	E	10

E

K	Z
1	20
4	24
5	28
7	32
9	36

Equijoin

K	X	Y	K	Z
1	A	2	1	20
4	D	8	4	24
5	E	10	5	28

Natural join

K	X	Y	Z
1	A	2	20
4	D	8	24
5	E	10	28

7. Division operation (÷)

- The Division operator results in columns values in one table for which there are other matching column values corresponding to every row in another table.

A

K	X	Y
10	1101	A
10	1201	B
10	1301	C
20	1201	B
30	1101	A
30	1201	B
30	1301	C

B (divisor)

X	Y
1101	A
1201	B
1301	C

Result

K
10
30

8. OUTER JOINS

- Equijoin and Natural Join are called inner joins.
- An inner join includes only those tuples with matching attributes and the rest are discarded in the resulting relation.
- Therefore, we need to use outer joins to include all the tuples from the participating relations in the resulting relation.
- There are three kinds of outer joins:
  - left outer join
  - right outer join
  - full outer join.

Prepared By Mr. EBIN PM, AP EDULINE 49

❖Left Outer Join (R ⟕ S)

- All the tuples from the Left relation, R, are included in the resulting relation. If there are tuples in R without any matching tuple in the Right relation S, then the S-attributes of the resulting relation are made NULL.

Left	
A	B
100	Database
101	Mechanics
102	Electronics

[Table: Left Relation]

Right	
A	B
100	Alex
102	Maya
104	Mira

[Table: Right Relation]

Prepared By Mr. EBIN PM, AP EDULINE 50

Courses HoD

A	B	C	D
100	Database	100	Alex
101	Mechanics	---	---
102	Electronics	102	Maya

[Table: Left outer join output]

Prepared By Mr. EBIN PM, AP EDULINE 51

❖Right Outer Join (R ⟖ S)

- All the tuples from the Right relation, S, are included in the resulting relation. If there are tuples in S without any matching tuple in R, then the R-attributes of resulting relation are made NULL.

Courses HoD

A	B	C	D
100	Database	100	Alex
102	Electronics	102	Maya
---	---	104	Mira

[Table: Right outer join output]

Prepared By Mr. EBIN PM, AP EDULINE 52

❖Full Outer Join (R ⟗ S)

- All the tuples from both participating relations are included in the resulting relation. If there are no matching tuples for both relations, their respective unmatched attributes are made NULL.

Courses		HoD	
A	B	C	D
100	Database	100	Alex
101	Mechanics	---	---
102	Electronics	102	Maya
---	---	104	Mira

[Table: Full outer join output]