3. Union

- Any relation is a set.
- Similar to union operation in set theory
- It is a binary operation
- It is a set of all objects that are a member of A, or B or both.
- Like project, the duplicate rows are eliminated
- It is denoted by $\cup$
- Syntax: $\Pi_{Column}(Relation\text{-}1) \cup \Pi_{column}(Relation\text{-}2)$

Exple: Consider $R = \{A, B, C, D, E, F\}$ and
$S = \{A, B, \$, E, F, I, \#\}$

$R \cup S = \{A, B, C, D, E, F, \$, I, \#\}$

Example 1: List all customer names associated with the bank either as an account holder or a loan borrower

Solution

$$\Pi_{Cu\text{-}Name}(Depositor) \cup \Pi_{Cu\text{-}Name}(Borrower)$$

Depositor	Cu-Name	Acc-No	Borrower	Cu-Name	Loan-No
↑	John	A-101	↑	Smith	L-201
R	Rose	A-304	S	John	L-658
	Mercy	A-502		Rose	L-254
	Paul	A-689		James	L-547

Table 1

Cu-Name
John
Rose
Mercy
Paul
Smith
James

Example: University Database (Schema 1)

Instructor (ID, Name, Dept_Name, Salary)
Course (Course-ID, Title, Dept-Name, Credits)
Department (Dept_Name, Building, Budget)
Section (Course_ID, Sec-ID, Semester, Year, Building, Room_No, Time-slot_ID)
Teaches (ID, Course-ID, Sec-ID, Semester, Year)
Student (ID, Name, Dept_Name, Tot_Credit)
Advisor (S_ID, I_ID)
Takes (ID, Course-ID, Sec-ID, Semester, Year, Grade)
Classroom (Building, Room-Number, Capacity)
Time-Slot (Time-Slot_ID, Day, Start-Time, End-Time)

Example 2: Find the set of all courses taught in the fall 2022 semester, Spring 2023 Semester or both

$\Pi_{course\text{-}id}(\sigma_{semester=\text{"Fall"} \wedge year=2022}(Section))$

Set of all courses taught in the fall 2022 and do same for Spring 2023

$\Pi_{course\text{-}id}(\sigma_{semester=\text{"Spring"} \wedge year=2023}(Section))$

To answer this query, we need the union of these two sets:

$\Pi_{course\_id}(\sigma_{semester="fall" \wedge year=2022}(Section)) \cup$

$\Pi_{course\_id}(\sigma_{semester="Spring" \wedge year=2023}(Section))$

Two Important Conditions

For $R \cup S$ to be valid,

1. R and S must be of same arity. That is the number of columns in R and S must be the same

2. For all i,

Domain of the i^{th} attribute of R = Domain of i^{th} attribute of S

Rem: Attributes are descriptive properties possessed by each member of an entity set. For each attribute there is a set of permitted values, called domain or value set of that attribute.

Assignment

1) List all the course IDs which are taken in Summer 2022 or Fall 2021 semester

2) List all the instructors' ID who taught courses in Spring 2020 or Fall 2021

4. Set Difference (−)

- It is like the same set difference in set theory
- It is a binary operator
- To find tuples that are in one relation but are not in another
- R−S = tuples in R but not in S
- It is denoted by minus (−) symbol.

Exple: Consider R = {A, B, C, D, E, F} and
S = {A, B, $, L, Y, I, #}

R − S = {C, D, E, F}   S − R = ?

Example: Consider Table 1 above,

1) List all customer names, those who have a deposit account but no available loan

$\Pi_{cu\_name}(Depositor) - \Pi_{cu\_name}(Borrower)$

Output: R−S

Cu_Name
Mercy
Paul

Consider Schema 1 above,

2) Find all the courses taught in fall 2022 Semester but not in Spring 2023

We need to find the set of courses taught in the fall 2022 and 2023 and minus these two sets

$\Pi_{course\_id}(\sigma_{semester = \text{"Fall"} \wedge year = 2022}(Section)) - \Pi_{course\_id}(\sigma_{semester = \text{"Spring"} \wedge year = 2023}(Section))$

Note: Both sets must have same arity and domain

Assignment

1) Find all the instructors ID who are not advisors
2) List all the instructors ID who taught in Spring 2023 but not advisors

5. Cartesian Product (X)

- Cartesian product associates every tuple of R_1 with every tuple of R_2.
- It is a binary operation.
- It is denoted by cross (X) symbol
- $R_1 \times R_2$ = All possible pairing
- Same attribute may appear in R_1 and R_2
- R = Depositor X Borrower

Example: Perform Depositor X Borrower

DEPOSITOR

Cu-Name	Acc-No
John	A-101
Rose	A-304

BORROWER

Cu-Name	Loan-No
Smith	L-201
John	L-658
Rose	L-254
James	L-547

DEPOSITOR X BORROWER

Cu_Name	Acc-No	Cu-Name	Loan_No
John	A-101	Smith	L-201
John	A-101	John	L-658
John	A-101	Rose	L-254
John	A-101	James	L-547
Rose	A-304	Smith	L-201
Rose	A-304	John	L-658

Rose	A-304	Rose	L-254
Rose	A-304	James	L-547

Is there any ambiguity in this table?
Yes Cu-Name is common

R = Depositor X Borrower

R = (Depositor.Cu-Name, Depositor.Acc-No, Borrower.Cu-Name, Borrower.Loan-No)
Acc-No Loan-No

Consider the Schema 1 above,
1) Find the names of all instructors in the Computer Science department together with the course-id of all courses they taught

INSTRUCTOR

ID	Name	Dept-Name	Salary
10101	John	Software-Eng	650,000
78787	Shu	Computer Science	870,000

TEACHES

ID	Course-ID	Sec-ID	Semester	Year
10101	SE108	1	Spring	2021
20202	CS103	1	Spring	2022
78787	ICT101	2	Fall	2023
12345	MAT203	1	Spring	2021

Table 2

Quickly do the Cartesian Product
INSTRUCTOR X TEACHES
(You should have 9 columns and 8 rows)
Solve all ambiguity problems

$\sigma_{Dept\_Name = \text{"Computer Science"}}(INSTRUCTOR \times TEACHES)$

You should have the following output

INSTRUCTOR.ID	Name	Dept Name	Salary	TEACHES.ID	Course_ID	Sec-ID	Semester	Year
78787	Shu	Computer Science	870,000	10101	SE 108	1	Summer	2021
78787	Shu	Computer Science	870,000	20202	CS 103	1	Summer	2022
78787	Shu	Computer Science	870,000	78787	ICT 101	2	Fall	2023
78787	Shu	Computer Science	870,000	12345	MAT 203	1	Spring	2021

$\sigma_{Instructor\_ID = Teaches.ID}(\sigma_{Dept\_Name = \text{"Computer\_Science"}}(INSTRUCTOR \times TEACHES))$

Give the output

But we are asked to find the name and course_ids of the instructors so we have

$\Pi_{name, course\_id}(\sigma_{Instructor.ID = Teaches.ID}(\sigma_{Dept\_Name = \text{"Computer Science"}}(INSTRUCTOR \times TEACHES)))$

Output:

Name	Course_ID
Shu	ICT101

OR

$\Pi_{name, course\_id}(\sigma_{Instructor.ID = Teaches.ID}((\sigma_{Dept\_Name = \text{"Computer Science"}}(INSTRUCTOR) \times TEACHES))$

Exercise: Give the step by step output of this RA

6. Rename (ρ)

- Relations in the database have names
- The results of relational-algebra expressions do not have a name
- It is useful to be able to give them names
- It is a unary operation
- It is denoted by the lowercase Greek letter rho (ρ)

• Syntax : $\rho_x(E)$

Example: Find the names of all instructors in the Computer Science department together with the course id of all courses they taught and name the resultant relation as Ins_CS

$\rho_{Ins\_Phy}(\Pi_{name,\, course\_id}(\sigma_{Instructor.ID = Teaches.ID}((\sigma_{Dept\_Name = \text{"Computer\_Science"}}(INSTRUCTOR) \times TEACHES)))$