Principles of Software Engineering and Data Bases

Davide Yi Xian Hu
Email: davideyi.hu@polimi.it
Date: 29 October 2024
Exercise Lecture: 01 - Relational Algebra

POLITECNICO
MILANO 1863

Davide Yi Xian Hu

- 👉 Ph.D. student @ PoliMi
- 👉 Teaching Assistant
- 👉 Building 22, third floor, office 001

- 👉 **Mail: davideyi.hu@polimi.it**

Exercise 1
Exams

Exercise 1 - Exams

Database Schema:

- 👉 **STUDENT** (Student-ID, Name, City, Degree Program)
- 👉 **EXAM** (Student-ID, Course-Code, Date, Grade)
- 👉 **COURSE** (Course-Code, Title, Professor)

Exercise 1 - Exams

Database Schema:

- **STUDENT** (Student-ID, Name, City, Degree Program)
- **EXAM** (Student-ID, Course-Code, Date, Grade)
- **COURSE** (Course-Code, Title, Professor)

PRIMARY KEY

A unique identifier for each record in the table

TABLE NAME

Name of the entity the table is representing

ATTRIBUTE

Column in the table that describe the properties of the entity

Exercise 1 - Exams

👉 **STUDENT** (Student-ID, Name, City, Degree Program)

STUDENT			
Student-ID	Name	City	Degree Program
000001	Davide	Milan	Informatics
000002	Mario	Milan	Health IT
000003	Giulia	Rome	Economics
...	...	...	...

Exercise 1 - Exams

👉 **COURSE** (Course-Code, Title, Professor)

COURSE		
Course-Code	Title	Professor
PSEDB01	Principles of Software Engineering and Data Bases	Prof. Luciano Baresi
IT01	Informatics	Prof. Mario Rossi
IT02	Informatics	Prof. Luigi Verdi
...	...	...

Exercise 1 - Exams

👉 **EXAM** (Student-ID, Course-Code, Date, Grade)

EXAM			
Student-ID	Course-Code	Date	Grade
000001	PSEDB01	27/10/2024	30
000002	PSEDB01	27/10/2024	18
000003	PSEDB01	27/10/2024	24
...	...	...	...

Exercise 1 - Exams - Query 1

❔ Find the names of students enrolled in the Health IT degree program in Milan.

Exercise 1 - Exams - Query 1

❔ Find the names of students enrolled in the Health IT degree program in Milan.

Solution

👉 $\pi_{\text{Name}} (\sigma_{\text{Degree Program}=\text{“Health IT”} \ \wedge \ \text{City}=\text{“Milan”}} \ \text{STUDENT})$

Wrong Solution

❌ $\sigma_{\text{Degree Program}=\text{“Health IT”} \ \wedge \ \text{City}=\text{“Milan”}} (\pi_{\text{Name}} \ \text{STUDENT})$

Exercise 1 - Exams - Query 1

👉 π_{Name} ($\sigma_{\text{Degree Program}=\text{“Health IT”} \wedge \text{City}=\text{“Milan”}}$ STUDENT)

STUDENT			
Student-ID	Name	City	Degree Program
000001	Davide	Milan	Informatics
000002	Mario	Milan	Health IT
000003	Giulia	Rome	Economics
...	...	...	...

Exercise 1 - Exams - Query 1

❌ $\sigma_{\text{Degree Program}=\text{“Health IT”} \wedge \text{City}=\text{“Milan”}} (\pi_{\text{Name}} \text{STUDENT})$

STUDENT			
Student-ID	Name	City	Degree Program
000001	Davide	Milan	Informatics
000002	Mario	Milan	Health IT
000003	Giulia	Rome	Economics
...	...	...	...

?

Exercise 1 - Exams - Query 2

❔ Find the names of students who have at least one grade of 30.

Exercise 1 - Exams - Query 2

❔ Find the names of students who have at least one grade of 30.

Solution

👉 π_{Name} ($\sigma_{Grade=30}$ (STUDENT ⋈ EXAM))

👉 π_{Name} (STUDENT ⋈ ($\sigma_{Grade=30}$ EXAM))

Wrong Solution:

👉 π_{Name} STUDENT ⋈ $\sigma_{Grade=30}$ EXAM

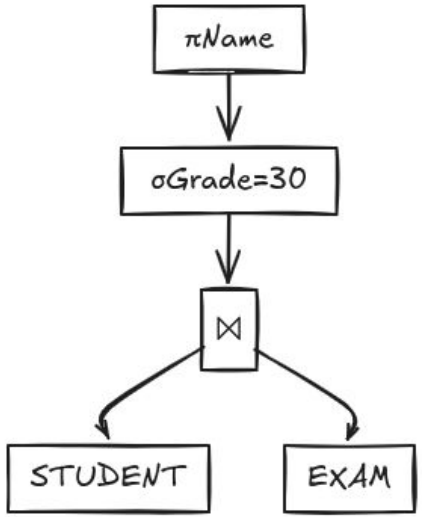
Exercise 1 - Exams - Query 2

Precedence Rule

👉 1. [**σ**, **π**, **ρ**] (highest)
👉 2. [**X**, ⋈]
👉 3. [∩]
👉 4. [∪ , —]

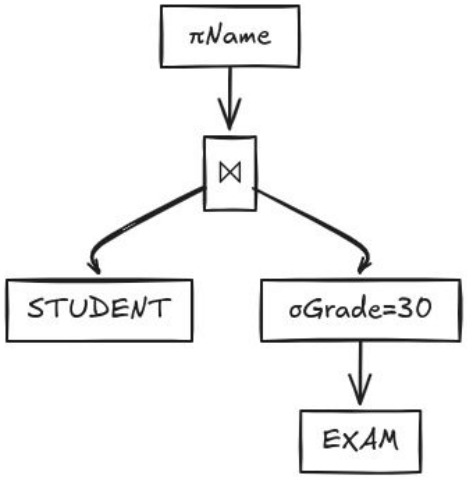
Exercise 1 - Exams - Query 2

👉 π_{Name} ($\sigma_{Grade=30}$ (STUDENT ⋈ EXAM))



Exercise 1 - Exams - Query 2

👉 π_{Name} (STUDENT ⋈ ($\sigma_{Grade=30}$ EXAM))



Exercise 1 - Exams - Query 3

❔ Find the “*Informatics*” courses in which at least one student not enrolled in the Management degree program has scored 30.

Exercise 1 - Exams - Query 3

❔ Find the "*Informatics*" courses in which at least one student not enrolled in the Management degree program has scored 30.

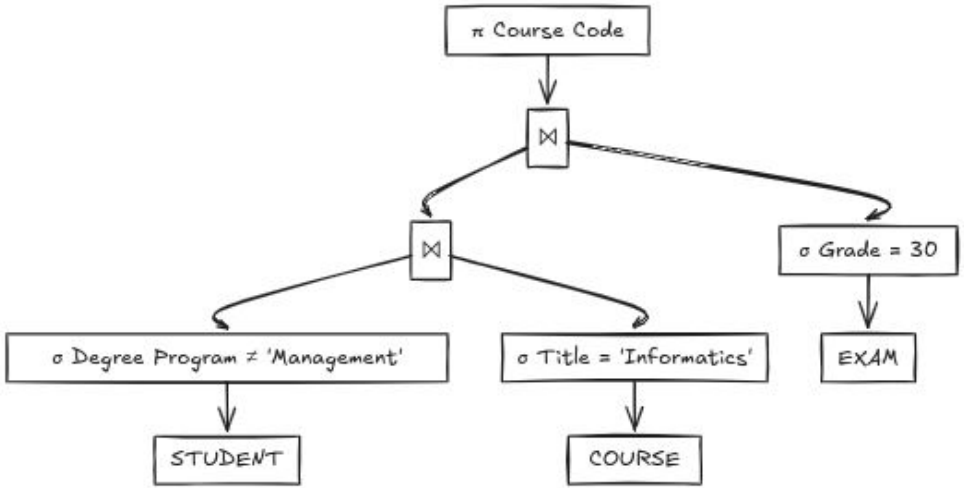
Solution

👉 $\pi_{\text{Course Code}} (\sigma_{\text{Title="Informatics"} \wedge \text{Grade=30} \wedge \text{Degree Program} \neq \text{"Management"}} (\text{STUDENT} \bowtie \text{EXAM} \bowtie \text{COURSE}))$

👉 $\pi_{\text{Course Code}} ((\sigma_{\text{Degree Program} \neq \text{"Management"}} \text{STUDENT}) \bowtie (\sigma_{\text{Grade=30}} \text{EXAM}) \bowtie (\sigma_{\text{Title="Informatics"}} \text{COURSE}))$

Exercise 1 - Exams - Query 3

👉 $\pi_{\text{Course Code}}$ (($\sigma_{\text{Degree Program}\neq\text{“Management”}}$ STUDENT) ⋈
($\sigma_{\text{Grade=30}}$ EXAM) ⋈
($\sigma_{\text{Title=“Informatics”}}$ COURSE))



Exercise 1 - Exams - Query 4

❔ Find the professors of Mathematics courses in which no student has scored 30.

Exercise 1 - Exams - Query 4

❔ Find the professors of Mathematics courses in which no student has scored 30.

Solution

👉 $\pi_{\text{Professor}} (\sigma_{\text{Title="Mathematics"}} \text{COURSE}) -$
$\pi_{\text{Professor}} ((\sigma_{\text{Grade=30}} \text{EXAM}) \bowtie (\sigma_{\text{Title="Mathematics"}} \text{COURSE}))$

👉 $\pi_{\text{Professor}} ((\sigma_{\text{Title="Mathematics"}} \text{COURSE}) -$
$(\sigma_{\text{Grade=30}} \text{EXAM}) \bowtie (\sigma_{\text{Title="Mathematics"}} \text{COURSE}))$

Exercise 1 - Exams - Query 5

❓ Find the title of courses where
no student has scored 18 and
no student has scored 30.

Exercise 1 - Exams - Query 5

❔ Find the title of courses where
no student has scored 18 and
no student has scored 30.

Solution

👉 π_{Title} ((COURSE - (COURSE $\bowtie$ ($\sigma_{\text{Grade=18}}$ EXAM))) $\cap$
(COURSE - (COURSE $\bowtie$ ($\sigma_{\text{Grade=30}}$ EXAM))))

Exercise 1 - Exams - Query 6

❔ Find the names of students
who have received at least one grade of 30 and
have never received a grade lower than 24.

Exercise 1 - Exams - Query 6

❔ Find the names of students
who have received at least one grade of 30 and
have never received a grade lower than 24.

Solution

👉 π_{Name} (STUDENT ⋈ ($\pi_{\text{Student-ID}}$ ($\sigma_{\text{Grade=30}}$ EXAM) -
($\pi_{\text{Student-ID}}$ ($\sigma_{\text{Grade<24}}$ EXAM))))

👉 π_{Name} ((STUDENT ⋈ ($\sigma_{\text{Grade=30}}$ EXAM)) -
(STUDENT ⋈ ($\sigma_{\text{Grade<24}}$ EXAM)))

Exercise 2
Flights

Exercise 2 - Flights

Database Schema:

👉 **FLIGHTS** (Departure, Arrival)

FLIGHTS	
Departure	Arrival
Milano Linate	London Heathrow
Milano Linate	London Gatwick
Milano Malpensa	London Heathrow
London Heathrow	New York JFK
London Gatwick	New York JFK
...	...

Exercise 2 - Flights - Query 1

❔ Determine all possible connections
between two airports where at most two flights are required.

Exercise 2 - Flights - Query 1

❔ Determine all possible connections
between two airports where at most **two** flights are required.

Solution

👉 Flights-1 = Flights, Flights-2 = Flights

Flights-1 ∪

$$\pi_{\text{Flights-1.Departure, Flights-2.Arrivals}} (\text{Flights-1} \bowtie_{\text{Flights-1.Arrivals = Flights-2.Departure}} \text{Flights-2})$$

Exercise 2 - Flights - Query 2

❔ Determine all possible connections
between two airports where at most three flights are required.

Exercise 2 - Flights - Query 2

❔ Determine all possible connections
between two airports where at most **three** flights are required.

Solution

👉 Flights-1 = Flights, Flights-2 = Flights, Flights-3 = Flights

$$\text{Flights-1} \cup \pi_{\text{Flights-1.Departure, Flights-2.Arrivals}} (\text{Flights-1} \bowtie_{\text{Flights-1.Arrivals, Flights-2.Departure}} \text{Flights-2}) \cup \pi_{\text{Flights-1.Departure, Flights-3.Arrivals}} (\text{Flights-1} \bowtie_{\text{Flights-1.Arrivals, Flights-2.Departure}} \text{Flights-2} \bowtie_{\text{Flights-2.Arrivals, Flights-3.Departure}} \text{Flights-3})$$

Exercise 2 - Flights - Query 3

❔ Determine all possible connections between two airports.

Exercise 2 - Flights - Query 3

❔ Determine all possible connections between two airports.

Solution

👉 **It’s not possible:**
Relational Algebra does not support recursive calls.

Exercise 3
Corporate Hierarchy

Exercise 3 - Hierarchy

Database Schema:

- 👉 **EMPLOYEE** (Name, Age, Salary)
- 👉 **SUPERVISION** (Manager, Subordinate)

Exercise 3 - Hierarchy

👉 **EMPLOYEE** (Name, Age, Salary)

👉 **SUPERVISION** (Manager, Subordinate)

EMPLOYEE		
Name	Age	Salary
Davide	28	1250
Mario	20	5000
Giulia	35	2400
...	...	...

SUPERVISION	
Manager	Subordinate
Giulia	Davide
Mario	Davide
Giulia	Mario
...	...

Exercise 3 - Hierarchy - Query 1

❔ Find the employees who report directly to Mario.

Exercise 3 - Hierarchy - Query 1

❔ Find the employees who report directly to Mario.

Solution

👉 $\pi_{\text{Name}} ((\sigma_{\text{Manager=“Mario”}} \text{SUPERVISION}) \bowtie_{\text{Subordinate=Name}} \text{EMPLOYEE})$

👉 $\pi_{\text{Subordinate}} (\sigma_{\text{Manager=“Mario”}} \text{SUPERVISION})$

Exercise 3 - Hierarchy - Query 2

❔ Find the pairs of employees who collaborate under Giulia (i.e., employees who share the same superior are considered to be collaborating).

Exercise 3 - Hierarchy - Query 2

❔ Find the pairs of employees who collaborate under Giulia (i.e., employees who share the same superior are considered to be collaborating).

Solution

👉 S1 = SUPERVISION, S2 = SUPERVISION

$$\pi_{S1.Subordinate,\ S2.Subordinate}\left(\left(\sigma_{Manager=\text{“Giulia”}}\ S1\right) \bowtie_{S1.Subordinate \neq S2.Subordinate} \left(\sigma_{Manager=\text{“Giulia”}}\ S2\right)\right)$$

Exercise 3 - Hierarchy - Query 2

❔ Find the pairs of employees who collaborate under Giulia (i.e., employees who share the same superior are considered to be collaborating).

Solution

👉 $$\pi_{S1,\,S2}\Big(\big(\sigma_{Manager=\text{“Giulia”}}(\rho_{S1\leftarrow Subordinate}\,SUPERVISION)\bowtie_{S1\neq S2}(\sigma_{Manager=\text{“Giulia”}}(\rho_{S2\leftarrow Subordinate}\,SUPERVISION))\big)\Big)$$

Exercise 3 - Hierarchy - Query 3

❔ Find the employees who do not have a manager.

Exercise 3 - Hierarchy - Query 3

❔ Find the employees who do not have a manager.

Solution

👉 π_{Name} (EMPLOYEE - $\pi_{Name, Age, Salary}$ ((EMPLOYEE $\bowtie_{Name=Subordinate}$ SUPERVISION)))

👉 (π_{Name} EMPLOYEE) - ($\pi_{Subordinate}$ SUPERVISION)

Exercise 3 - Hierarchy - Query 4

❔ Find the managers who have at least one subordinate who earns more than the manager.

Exercise 3 - Hierarchy - Query 4

❔ Find the managers who have at least one subordinate who earns more than the manager.

Solution

👉 $\pi_{mgr.Name}\left(\left(\left(\rho_{sub.Name \leftarrow Name,\ sub.Salary \leftarrow Salary}\ EMPLOYEE\right)\right.\right.$

$\bowtie_{sub.Salary > mgr.Salary}$

$\left(\rho_{mgr.Name \leftarrow Name,\ mgr.Salary \leftarrow Salary}\ EMPLOYEE\right)\left.\right)$

$\bowtie_{mgr.Name=Manager \wedge sub.Name=Subordinate}\ SUBORDINATE\left.\right)$