University
Mohammed VI
Polytechnic

College of Computing

Deliverable # 3: Relational Algebra, SQL and Functional Dependencies

Data Management Course

UM6P College of Computing

Professor: Karima Echihabi **Program:** Computer Engineering

Session: Fall 2025

Team Information

Team Name	Leo fl Bernabéu
Member 1	Yassine SQUALLI-HOUSSAINI
Member 2	Omar TSOULI
Member 3	Yahya TALIB
Member 4	Adam RHYA
Member 5	Hamza TAALOUCHT
Member 6	Diaa Eddine ZAINI
Repository Link	https://github.com/therealzaini/DMG_LAB2_LEO_FL_BERNABEU

1 Relational Algebra Expressions for the Queries and SQL Implementation:

Query 1:

Relational algebra expressions:

$$\rho\,(R_1, \quad \text{Staff} \bowtie \text{ClinicalActivity})$$
$$\rho\,(R_2, \quad R_1 \bowtie \text{Patient})$$
$$\pi_{\text{FullName}}\,(\sigma_{\text{Status} = \text{“Active”}}\,(R_2))$$

SQL Implementation:

```sql
SELECT DISTINCT P.FullName
FROM Patient P
JOIN ClinicalActivity C ON P.IID = C.IID
JOIN Staff S ON C.STAFF_ID = S.STAFF_ID
WHERE S.Status = 'Active';
```

Query 2:

Relational algebra expressions:

$$\rho\,(R_1, \quad \pi_{\text{SID}}\,(\sigma_{\text{Status} = \text{“Active”}}\,(\text{Staff})))$$
$$\rho\,(R_2, \quad \pi_{\text{SID}}\,(\text{ClinicalActivity} \bowtie \text{Appointment}))$$
$$R_1 \cup R_2$$

SQL Implementation:

```sql
SELECT STAFF_ID
FROM Staff
WHERE Status = 'Active'
UNION
SELECT staff_ID
FROM clinicalactivity NATURAL JOIN prescription;
```

Query 3:

Relational algebra expressions:

$$\rho\,\left(R_1, \quad \pi_{\text{HID}}\,\sigma_{\text{city} = \text{“Benguerir”}}\,(\text{Hospital})\right)$$
$$\rho\,\left(R_2, \quad \pi_{\text{HID}}\,\left(\sigma_{\text{Speciality} = \text{“Cardiology”}}\,(\text{Dept})\right)\right)$$
$$R_1 \cup R_2$$

SQL Implementation:

```sql
SELECT HID
FROM Hospital
WHERE city = 'Benguerir'
UNION
SELECT HID
FROM Dept
WHERE speciality = 'Cardiology';
```

Query 4:

Relational algebra expressions:

$$T_1 = \pi_{\text{HID}}\left(\sigma_{\text{speciality} = \text{“cardiology”}}(\text{Dept})\right)$$
$$T_2 = \pi_{\text{HID}}\left(\sigma_{\text{speciality} = \text{“Pediatrics”}}(\text{Dept})\right)$$
$$T_1 \cap T_2$$

SQL Implementation:

```sql
SELECT HID
FROM Dep
WHERE specialty = 'cardiology'
INTERSECT
SELECT HID
FROM Dep
WHERE specialty = 'pediatrics';
```

Query 5:

Relational algebra expressions:

$$T_1 = \pi_{\text{DEP\_ID}}\left(\sigma_{\text{HID} = 1}(\text{Dept})\right)$$
$$T_2 = \text{work\_in}/T_1$$
$$\text{result} = \pi_{\text{name}}(T_2 \bowtie \text{Staff})$$

SQL Implementation:

```sql
SELECT S.name
FROM Staff S
NATURAL JOIN work_in W
NATURAL JOIN Dept D
WHERE D.hid = 1
/* Here we are going to group staff members by name
if they have worked in all departments that belong
to hospital of id 1, by using count keyword:*/
GROUP BY S.staff_id, S.name
HAVING COUNT(DISTINCT D.dep_id) = (
    SELECT COUNT(DISTINCT dep_id)
```

```
    FROM Dept
    WHERE hid = 1
);
```

Query 6:

Relational algebra expressions:

$$\rho\left(R_1, \quad \pi_{\text{staff\_id, CAID}}\left(\text{ClinicalActivity}\right)\right)$$
$$\rho\left(R_2, \quad \pi_{\text{CAID}}\sigma_{\text{DEPT\_ID}=2}\left(\text{ClinicalActivity}\right)\right)$$
$$\pi_{\text{name}}\left(R_1|R_2 \bowtie \text{staff}\right)$$

SQL Implementation:

```
SELECT S.name
FROM Staff S
NATURAL JOIN Clinical_activity C
WHERE C.dep_id = 2
GROUP BY S.staff_id, S.name
HAVING COUNT(DISTINCT C.CAID) = (
    SELECT COUNT(DISTINCT CAID)
    FROM Clinical_activity
    WHERE dep_id = 2
);
```

Query 7:

Relational algebra expressions:

$$\rho\left(R_1, \quad \gamma_{\text{Staff\_ID}}\left(\text{count}\left(\text{CAID}\right) \longrightarrow n, \quad \pi_{\text{CAID, Staff\_ID}}\left(\text{ClinicalActivity}\right)\right)\right)$$
$$\rho\left(R_2\left(1 \longrightarrow \text{Staff\_ID0}, \quad 2 \longrightarrow n_0\right), \quad R_1 \times R_2\right)$$
$$\pi_{\text{Staff\_ID0}\times\text{Staff\_ID}}\left(\sigma_{n_0>n}\left(R_2\right)\right)$$

SQL Implementation:

```
WITH R1 AS (
    SELECT staff_id, COUNT(CAID) AS n
    FROM ClinicalActivity
    GROUP BY staff_id
),
R2 AS (
    SELECT R1a.staff_id AS staff_id0,
           R1a.n AS n0,
           R1b.staff_id,
           R1b.n
    FROM R1 AS R1a
    CROSS JOIN R1 AS R1b
)
```

```sql
SELECT staff_id0, staff_id
FROM R2
WHERE n0>n;
```

Query 8:

Relational algebra expressions:

$$\rho\left(C\left(3 \longrightarrow \text{SID1}, \quad 9 \longrightarrow \text{SID2}\right), \quad \text{ClinicalActivity} \times \text{ClinicalActivity}\right)$$
$$\pi_{\text{IID}}\left(\sigma_{\text{SID1} \neq \text{SID2}}(C)\right)$$

SQL Implementation:

```sql
SELECT C.IID
FROM ClinicalActivity C
GROUP BY C.IID
HAVING COUNT(DISTINCT C.STAFF_ID) >= 2;
```

Query 9:

Relational algebra expressions:

$$\rho\left(R_1, \quad \sigma_{\text{city} = \text{“Benguerir”}}\left(\text{Hospital}\right) \bowtie \text{Dept}\right)$$
$$\rho\left(R_2, \quad \sigma_{\text{date} = \text{September 2025}}\left(\mathit{ClinicalActivity}\right) \bowtie R_1\right)$$
$$\pi_{\text{CAID}}\left(R_2\right)$$

SQL Implementation:

```sql
SELECT C.CAID
FROM ClinicalActivity C
JOIN Department D ON C.DEP_ID = D.DEP_ID
JOIN Hospital H ON D.HID = H.HID
WHERE H.City = 'Benguerir'
  AND YEAR(C.Date) = 2025
  AND MONTH(C.Date) = 9;
```

Query 10:

Relational algebra expressions:

$$\rho\left(R_1, \quad \pi_{\text{Staff\_ID, CAID, PID}}\left(\text{ClinicalActivity}\right)\right)$$
$$\rho\left(R_2, \quad \pi_{\text{Staff\_ID, CAID, PID}}\left(\text{ClinicalActivity}\right)\right)$$
$$\pi_{\text{Staff\_ID}}\left(\sigma_{R_1.\text{Staff\_ID} = R_2.\text{Staff\_ID} \wedge R_1.\text{PID} \neq R_2.\text{PID}}\left(R_1 \times R_2\right)\right)$$

SQL Implementation:

```sql
SELECT DISTINCT r1.Staff_id
FROM (
    SELECT Staff_id, PID, CAID
    FROM Clinical_activity
) AS r1
CROSS JOIN (
    SELECT Staff_id, PID, CAID
    FROM Clinical_activity
) AS r2
WHERE r1.Staff_id = r2.Staff_id
  AND r1.PID <> r2.PID;
```

Query 11:

Relational algebra expressions:

$$T_1 = \pi_{\text{IID, CID, Dep\_ID}}\left(\sigma_{\text{Status = ``Scheduled''}}(\text{ClinicalActivity} \bowtie \text{Appointments})\right)$$
$$T_2 = \pi_{\text{IID, CID, Dep\_ID}}\left(\sigma_{\text{Status = ``Scheduled''}}(\text{ClinicalActivity} \bowtie \text{Appointments})\right)$$
$$\text{result} = \pi_{T_1.\text{IID}}\left(\sigma_{T_1.\text{IID} = T_2.\text{IID} \wedge T_1.\text{Dep\_ID} \neq T_2.\text{Dep\_ID}}(T_1 \times T_2)\right)$$

SQL Implementation:

```sql
SELECT DISTINCT T1.IID
FROM
    (SELECT CA.IID, CA.Dep_id
     FROM Clinical_activity CA
     NATURAL JOIN Appointments A
     WHERE CA.Status = 'Scheduled') AS T1
CROSS JOIN
    (SELECT CA.IID, CA.Dep_id
     FROM Clinical_activity CA
     NATURAL JOIN Appointments A
     WHERE CA.Status = 'Scheduled') AS T2
WHERE T1.IID = T2.IID
  AND T1.Dep_id <> T2.Dep_id;
```

Query 12:

Relational algebra expressions:

$$\pi_{\text{Staff\_ID}}(\text{Staff}) - \pi_{\text{Staff\_ID}}\left(\sigma_{\substack{\text{Status=``Scheduled''} \wedge \\ \text{EXTRACT(MONTH FROM Date)=11} \wedge \\ \text{EXTRACT(DAY FROM Date)=6}}}(\text{Appointment} \bowtie \text{ClinicalActivity})\right)$$

SQL Implementation:

```sql
SELECT STAFF_ID
FROM Staff
WHERE STAFF_ID NOT IN (
SELECT CA.STAFF_ID
FROM ClinicalActivity CA
JOIN Appointment A ON CA.CAID = A.CAID
WHERE A.Status = 'Scheduled'
AND EXTRACT(MONTH FROM CA.Date) = 11
AND EXTRACT(DAY FROM CA.Date) = 6
);
```

Query 13:

Relational algebra expressions:

$$\pi_{\text{DEP\_ID}}\Big(\sigma_{\text{activity\_count}<\text{global\_avg}}\Big(\big(\gamma_{\text{DEP\_ID};\text{activity\_count}\leftarrow\text{COUNT(CAID)}}(\text{ClinicalActivity})\big)\times \big(\gamma_{;\text{global\_avg}\leftarrow\text{AVG(activity\_count)}}\big(\gamma_{\text{DEP\_ID};\text{activity\_count}\leftarrow\text{COUNT(CAID)}}(\text{ClinicalActivity})\big)\big)\Big)\Big)$$

SQL Implementation:

```sql
WITH DeptActivity AS (
SELECT DEP_ID, COUNT(CAID) AS activity_count
FROM ClinicalActivity
GROUP BY DEP_ID
),
GlobalAverage AS (
SELECT AVG(activity_count) AS global_avg
FROM DeptActivity
)
SELECT DA.DEP_ID
FROM DeptActivity DA, GlobalAverage GA
WHERE DA.activity_count < GA.global_avg;
```

Query 14:

Relational algebra expressions:

$$\rho\left(R_1,\quad \pi_{\text{Staff\_ID, Status}}\left(\text{ClinicalActivity} \bowtie \text{Appointment}\right)\right)$$
$$\gamma_{\text{Staff\_ID}}\left(\max\left(\text{count}\left(\text{CAID}\right)\longrightarrow n\right)\left(R_1\right)\right)$$

SQL Implementation:

```sql
WITH R1 AS (
    select CAID,Staff_id,status
```

```
    from clinicalactivity natural join appointment
),
Counts AS (
    SELECT staff_ID, COUNT(CAID) AS n
    FROM R1
    GROUP BY staff_ID
)
SELECT staff_ID, MAX(n) AS max_count
FROM Counts
GROUP BY staff_ID;
```

Query 15:

Relational algebra expressions:

$$\pi_{\text{IID}}\left(\sigma_{\text{Count}\geq 3}\left(\gamma_{\text{IID; COUNT}(*)\rightarrow\text{Count}}\left(\sigma_{\substack{\text{ActivityType='Emergency'}\\ \wedge\\ \text{Date}\geq\text{'2024-01-01'}\\ \wedge\\ \text{Date}\leq\text{'2024-12-31'}}}(\text{ClinicalActivity})\right)\right)\right)$$

SQL Implementation:

```
SELECT IID
FROM ClinicalActivity
WHERE ActivityType = 'Emergency'
  AND Date BETWEEN '2024-01-01' AND '2024-12-31'
GROUP BY IID
HAVING COUNT(*) >= 3;
```

2 Functional Dependencies:

Every primary key determines the attributes of of its table or relationship.

Hospital city determines region and contact location city determines province.

Phone location determines city/province/street in contact location.