



IT214 Lab 4 Report
Keyur Govrani
202101498

Contents

1	Relational algebraic expressions for all queries	2
2	SQL statements for all queries	3

1 Relational algebraic expressions for all queries

Relational algebraic expressions of queries

Page No.	
Date	

- 1) $\Pi_{\text{instructorid}, \text{instructorname}} (\sigma_{\text{semester} = 'Autumn' \text{ and } \text{acadyear} = 2010} (\text{instructor} * \text{offers}))$
- 2) $\Pi_{\text{name}, \text{studentid}} (\sigma_{\text{batch} = 2008 \text{ and } \text{semester} = 'Autumn' \text{ and } \text{acadyear} = 2008 \text{ and } \text{courseid} = 'MT101'} (\text{student} * \text{registers}))$
- 3) $\Pi_{\text{studentid}, \text{name}} (\sigma_{\text{progid} = '01' \text{ and } \text{batch} = 2008 \text{ and } \text{semester} = 'Autumn' \text{ and } \text{acadyear} = 2008 \text{ and } \text{courseid} = 'MT101'} (\text{student} * \text{registers}))$
- 4) $\Pi_{\text{courseid}} * (\text{course} * (\Pi_{\text{courseid}}(\text{offers}) - \Pi_{\text{courseid}}(\text{registers})))$
- 5) $\Pi_{\text{studentid}, \text{courseid}, \text{name}, \text{grade}} (\sigma_{\text{grade} = 'F' \text{ and } \text{acadyear} = 2008 \text{ and } \text{semester} = 'Autumn'} (\text{student} * \text{registers}))$
- 6) $\Pi_{\text{courseid}} * (\sigma_{\text{instructorid} = 'PMJ' \text{ and } \text{acadyear} = 2010 \text{ and } \text{semester} = 'winter'} (\text{course} * \text{offers}))$
- 7) $\Pi_{\text{studentid}, \text{acadyear}, \text{courseid}} * (\sigma_{\text{studentid} = '200711002' \text{ and } \text{acadyear} = 2008} (\text{student} * \text{registers} * \text{course}))$
- 8) $\Pi_{\text{studentid}} (\text{student} * (\Pi_{\text{studentid}} (\sigma_{\text{courseid} = 'MT101'} (\text{registers})) \cap \Pi_{\text{studentid}} (\sigma_{\text{courseid} = 'MT104'} (\text{registers}))))$
- 9) $\Pi_{\text{studentid}} (\text{student} * (\Pi_{\text{studentid}} (\sigma_{\text{courseid} = 'MT101'} (\text{registers})) - \Pi_{\text{studentid}} (\sigma_{\text{courseid} = 'MT104'} (\text{registers}))))$

2 SQL statements for all queries

1. List Faculty (ID, Name) for all courses of Autumn'2010

```
SELECT instructorid, instructorname FROM
(instructor NATURAL JOIN offers)
WHERE semester = 'Autumn' AND acadyear = 2010;
```

The screenshot shows a PostgreSQL web interface with the following components:

- Top Bar:** Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes. The active connection is `public/202101498@PostgreSQL*`.
- Query Editor:** Contains the SQL query:

```
-- 1
SELECT instructorid, instructorname FROM (instructor NATURAL JOIN offers)
WHERE semester = 'Autumn' AND acadyear = 2010;

-- 2
SELECT studentid, name FROM (student NATURAL JOIN registers)
WHERE batch = 2008 AND semester = 'Autumn' AND acadyear = 2008 AND courseno = 'MT101';

-- 3
SELECT studentid, name FROM (student NATURAL JOIN registers)
WHERE progid = '01' AND batch = 2008 AND semester = 'Autumn'
AND acadyear = 2008 AND courseno = 'MT101';
```
- Data Output:** Shows the results of the first query in a table with two columns: `instructorid` and `instructorname`.

	instructorid [PK] character varying (5)	instructorname character varying (30)
1	MHR	Mehul Raval
2	PK	Pankaj Kumar
- Footer:** Total rows: 2 of 2 | Query complete 00:00:00.104 | Successfully run. Tot

2. List Students (ID, Name) from batch 2008 registered for course MT101 in Autumn'2008.

```
SELECT studentid, name FROM (student NATURAL JOIN registers)
WHERE batch = 2008 AND semester = 'Autumn'
AND acadyear = 2008 AND courseno = 'MT101';
```

Dashboard Properties SQL Statistics Dependencies Dependents Processes public/202101498@PostgreSQL*

public/202101498@PostgreSQL

Query Query History

```
15 -- 2
16 SELECT studentid, name FROM (student NATURAL JOIN registers)
17 WHERE batch = 2008 AND semester = 'Autumn' AND acadyear = 2008 AND courseno = 'MT101';
18
19 -- 3
20 SELECT studentid, name FROM (student NATURAL JOIN registers)
```

Data Output Messages Notifications

	studentid [PK] character varying (9)	name character varying (30)
1	200811001	Rama Kant
2	200811002	Akshya Gupta
3	200811003	Unnati Gupta
4	200811004	Mridula Singh
5	200811005	Amit Ajaad
6	200802001	Ram Kant
7	200802002	Akshya Singh
8	200802003	Shobha Gupta
9	200802004	Shikha Singh
10	200802005	Ashish Jain
11	200801001	Ram Kant
12	200801002	Akshya Singh
13	200801003	Shobha Gupta
14	200801004	Shikha Singh
15	200801005	Ashish Jain

Total rows: 15 of 15 Query complete 00:00:00.114

3. List Students (ID, Name) from B Tech (progid='01') batch 2008 registered for course MT101 in Autumn'2008.

```
SELECT studentid, name FROM (student NATURAL JOIN registers)
WHERE progid = '01' AND batch = 2008
AND semester = 'Autumn' AND acadyear = 2008
AND courseno = 'MT101';
```

The screenshot shows a PostgreSQL web interface with the following components:

- Navigation Bar:** Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes. The active user is `public/202101498@PostgreSQL*`.
- Query Editor:** Contains the SQL query:

```
18
19 -- 3
20 SELECT studentid, name FROM (student NATURAL JOIN registers)
21 WHERE progid = '01' AND batch = 2008 AND semester = 'Autumn'
22 AND acadyear = 2008 AND courseno = 'MT101';
23
24 -- 4
25 SELECT * FROM course
26 NATURAL JOIN
27 (
28     SELECT courseno FROM offers
29     EXCEPT
30     SELECT courseno FROM registers
```
- Data Output:** Displays the results of the query in a table with two columns: `studentid` and `name`.
- Summary:** Total rows: 5 of 5. Query complete 00:00:00.103.

	studentid [PK] character varying (9)	name character varying (30)
1	200801001	Ram Kant
2	200801002	Akshya Singh
3	200801003	Shobha Gupta
4	200801004	Shikha Singh
5	200801005	Ashish Jain

4. List courses that were offered but students were not registered in those offerings

```
SELECT * FROM course
NATURAL JOIN
(
    SELECT courseno FROM offers
    EXCEPT
    SELECT courseno FROM registers
) AS temp;
```

The screenshot shows a PostgreSQL query editor interface. The top navigation bar includes tabs for Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, and Processes. The current session is identified as 'public/202101498@PostgreSQL*'. Below the navigation bar is a toolbar with icons for file operations, query execution, and other database functions. The main area is divided into two tabs: 'Query' and 'Query History'. The 'Query' tab is active, displaying a SQL query that has been executed. The query is as follows:

```
23
24 -- 4
25 SELECT * FROM course
26 NATURAL JOIN
27 (
28     SELECT courseno FROM offers
29     EXCEPT
30     SELECT courseno FROM registers
31 )
32 ) AS temp;
33
34 -- 5
35 SELECT studentid, name, courseno, grade FROM student
36 NATURAL JOIN registers
```

Below the query editor, there is a 'Data Output' tab, which is currently empty. To the right of the 'Data Output' tab, there are three columns of data types: 'courseno' (character varying (5)), 'coursename' (character varying (35)), and 'credit' (numeric (3,1)). At the bottom of the interface, a status bar indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.103'.

5. List students (id, name, courseno, grade) who got 'F' grade in Autumn'2008

```
SELECT studentid, name, courseno, grade FROM student
NATURAL JOIN registers
WHERE grade = 'F' AND acadyear = 2008 AND semester = 'Autumn';
```

Dashboard Properties SQL Statistics Dependencies Dependents Processes public/202101498@PostgreSQL*

public/202101498@PostgreSQL

Query Query History

```
33
34 -- 5
35 SELECT studentid, name, courseno, grade FROM student
36 NATURAL JOIN registers
37 WHERE grade = 'F' AND acadyear = 2008 AND semester = 'Autumn';
38
39 -- 6
40 SELECT course.* FROM course NATURAL JOIN offers
41 WHERE instructorid = 'PMJ' AND acadyear = 2010 AND semester = 'Winter';
42
43 -- 7
44 SELECT studentid, acadyear, course.*, grade FROM student NATURAL JOIN registers NATURAL JOIN course
45 WHERE studentid = '200711002' AND acadyear = 2008;
```

Data Output Messages Notifications

	studentid character varying (9)	name character varying (30)	courseno character varying (5)	grade character varying (2)
1	200701005	Amit Tiwari	MT101	F

Total rows: 1 of 1 Query complete 00:00:00.072

✓ Successfully run. To

6. List Courses (Course No, Title, Credits) offered by Instructor (ID='PMJ') in semester Winter'2010.

```
SELECT course.* FROM course NATURAL JOIN offers
WHERE instructorid = 'PMJ' AND acadyear = 2010
AND semester = 'Winter';
```

Dashboard Properties SQL Statistics Dependencies Dependents Processes public/202101498@PostgreSQL*

public/202101498@PostgreSQL

Query Query History

```
33
34 -- 5
35 SELECT studentid, name, courseno, grade FROM student
36 NATURAL JOIN registers
37 WHERE grade = 'F' AND acadyear = 2008 AND semester = 'Autumn';
38
39 -- 6
40 SELECT course.* FROM course NATURAL JOIN offers
41 WHERE instructorid = 'PMJ' AND acadyear = 2010 AND semester = 'Winter';
42
43 -- 7
44 SELECT studentid, acadyear, course.*, grade FROM student NATURAL JOIN registers NATURAL JOIN course
45 WHERE studentid = '200711002' AND acadyear = 2008;
```

Data Output Messages Notifications

	courseno [PK] character varying (5)	coursename character varying (35)	credit numeric (3,1)
1	MT202	Adhoc Network	4.0
2	EC403	Settelite Communication	3.0

Total rows: 2 of 2 Query complete 00:00:00.085

✓ Successfully run. To

7. Produce transcript of student '200711002' for Academic year 2008-09 List Course No, Course Name, Course Credit, Grade

```
SELECT studentid, acadyear, course.*, grade FROM student
NATURAL JOIN registers NATURAL JOIN course
WHERE studentid = '200711002' AND acadyear = 2008;
```

Dashboard Properties SQL Statistics Dependencies Dependents Processes public/202101498@PostgreSQL*

public/202101498@PostgreSQL

Query Query History

```
38
39 -- 6
40 SELECT course.* FROM course NATURAL JOIN offers
41 WHERE instructorid = 'PMJ' AND acadyear = 2010 AND semester = 'Winter';
42
43 -- 7
44 SELECT studentid, acadyear, course.*, grade FROM student
45 NATURAL JOIN registers NATURAL JOIN course
46 WHERE studentid = '200711002' AND acadyear = 2008;
47
48 -- 8
49 SELECT studentid FROM student NATURAL JOIN
50 (
51 SELECT studentid FROM registers
```

Data Output Messages Notifications

	studentid character varying (9)	acadyear integer	courseno character varying (5)	coursename character varying (35)	credit numeric (3,1)	grade character varying (2)
1	200711002	2008	MT102	Computer Networks	4.0	AA
2	200711002	2008	MT104	Probability	3.0	AA
3	200711002	2008	MT105	VLSI	3.0	BB

Total rows: 3 of 3 Query complete 00:00:00.070

✓ Successfully run. T

8. List ID of students who have taken both courses - 'MT101' AND 'MT104'.

```
SELECT studentid FROM student NATURAL JOIN
(
    SELECT studentid FROM registers
    WHERE courseno = 'MT101'

    INTERSECT

    SELECT studentid FROM registers
    WHERE courseno = 'MT104'

) AS temp;
```

Dashboard Properties SQL Statistics Dependencies Dependents Processes public/202101498@PostgreSQL*

public/202101498@PostgreSQL

Query Query History

```
48 -- 8
49 SELECT studentid FROM student NATURAL JOIN
50 (
51     SELECT studentid FROM registers
52     WHERE courseno = 'MT101'
53     INTERSECT
54     SELECT studentid FROM registers
55     WHERE courseno = 'MT104'
56
57 ) AS temp;
58
59 -- 9
60 SELECT studentid FROM student NATURAL JOIN
```

Data Output Messages Notifications

	studentid [PK] character varying (9)
1	200811001
2	200811002
3	200811003
4	200811004
5	200811005

Total rows: 5 of 5 Query complete 00:00:00.093

✓ Successfully run. To

9. List ID of students who have taken 'MT101' but not 'MT104'

```
SELECT studentid FROM student NATURAL JOIN
(
    SELECT studentid FROM registers
    WHERE courseno = 'MT101'

    EXCEPT

    SELECT studentid FROM registers
    WHERE courseno = 'MT104'

) AS temp;
```

The screenshot shows a PostgreSQL query editor interface. The top navigation bar includes 'Dashboard', 'Properties', 'SQL', 'Statistics', 'Dependencies', 'Dependents', and 'Processes'. The current connection is 'public/202101498@PostgreSQL*'. The query editor shows the following SQL code:

```
56
57 ) AS temp;
58
59 -- 9
60 SELECT studentid FROM student NATURAL JOIN
61 (
62     SELECT studentid FROM registers
63     WHERE courseno = 'MT101'
64     EXCEPT
65     SELECT studentid FROM registers
66     WHERE courseno = 'MT104'
67
68 ) AS temp;
```

Below the query editor, the 'Data Output' tab is active, showing a table with the following data:

	studentid [PK] character varying (9)
1	200702001
2	200702002
3	200702003
4	200702004
5	200702005
6	200802001
7	200802002
8	200802003
9	200802004
10	200802005

The status bar at the bottom indicates 'Total rows: 20 of 20' and 'Query complete 00:00:00.106'.