

Practical 6

Name: Shivam Tawari

Roll no: A-58

Aim: Exercise 2: Data retrieval from the designed database. e.g. JOINS.

Theory:

Join: A join clause is used to combine rows from two or more tables, based on a related column between them.

Types of joins:

1. Inner join:

```
SELECT column_name(s)
  FROM table1
  INNER JOIN table2
  ON table1.column_name = table2.column_name;
```
2. Left join:

```
SELECT column_name(s)
  FROM table1
  LEFT JOIN table2
  ON table1.column_name = table2.column_name;
```
3. Right join:

```
SELECT column_name(s)
  FROM table1
  RIGHT JOIN table2
  ON table1.column_name = table2.column_name;
```
4. Full join:

```
SELECT column_name(s)
  FROM table1
  FULL OUTER JOIN table2
  ON table1.column_name = table2.column_name
  WHERE condition;
```
5. Natural join:

```
SELECT *
  FROM table1
  NATURAL JOIN table2;
```

Program/Queries:

SQL Code followed by Output Screenshot for each table

-- Operation 1: INNER JOIN

SQL Script:

SELECT *

FROM Faculties

INNER JOIN Departments

ON Faculties.DeptID = Departments.DeptID;

Output:

The screenshot shows a PostgreSQL SQL Editor window titled "Query - Practical 6 on postgres@localhost:5432 *". The SQL Editor contains the following script:

```
ALTER TABLE Students RENAME COLUMN ContactDetails TO ContactNumber;
ALTER TABLE Students ADD AGE INT;
ALTER TABLE Students ADD CHECK (Age > 10);
ALTER TABLE Students ADD TGID INT;

INSERT INTO Students VALUES
('2019AAIE1111007', 'Shivam', 'Tawari', '2001-10-07', 4, 1, 7020282332, 19),
('2019AAIE1111010', 'Ajay', 'Sandhu', '2000-05-15', 4, 2, 9452195559, 21),
('2020AAIE1111111', 'Abhishek', 'Ingle', '2001-04-12', 2, 3, 8693738291, 20),
('2019AAIE1111234', 'Rajat', 'Sharma', '2001-09-18', 4, 1, 9523654715, 19),
('2019AAIE111123', 'Rohit', 'Kumar', '2001-01-05', 6, 3, 9686940294, 20),
('2019AAIE1111445', 'Elon', 'Morgan', '2001-11-09', 4, 1, 7685949322, 19),
('2020AAIE1111001', 'Rumar', 'Sangameshwar', '2001-10-21', 2, 2, 6909029310, 19);

SELECT *
FROM Faculties
INNER JOIN Departments
ON Faculties.DeptID = Departments.DeptID;
```

The Output pane shows the results of the query, displaying a table with 10 columns: techid, deptid, firstname, lastname, contactno, dateofjoin, deptid, deptname, deptest, and headofdept. The table contains 5 rows of data.

techid	deptid	firstname	lastname	contactno	dateofjoin	deptid	deptname	deptest	headofdept
1	101	Durgesh	Sharma	852718349	2021-01-0	1	Artificial Intelli	2019-01	A. Thomas
2	60	Achamma	Thomas	837363525	2010-03-0	1	Artificial Intelli	2019-01	A. Thomas
3	65	Dipti	Theng	837829191	2013-01-0	2	Computer Science	2010-01	Snehlata Dongre
4	200	Praveen	Ghatode	765342791	2012-01-0	3	Information Techno	2009-01	Mahendra Gaikwad
5	201	Gopal	Sakarkar	87772901	2014-01-0	4	Data Science	2021-01	A. Thomas

-- Operation 2: LEFT JOIN or LEFT OUTER JOIN

SQL Script:

SELECT *

FROM Students

LEFT JOIN Faculties

ON Faculties.TechID = Students.TGID;

Output:

The screenshot shows a PostgreSQL SQL Editor window titled "Query - Practical 6 on postgres@localhost:5432". The SQL Editor contains the following query:

```
SELECT *  
FROM Faculties  
INNER JOIN Departments  
ON Faculties.DeptID = Departments.DeptID;  
  
SELECT *  
FROM Students  
LEFT JOIN Faculties  
ON Faculties.TechID = Students.TGID;
```

The Output pane displays the results of the query, showing a table with 12 columns: regid, character varying(20), firstname, character varying(100), lastname, character varying(100), dateofbirth, date, semester, integer, departmentid, integer, contactnumber, bigint, age, integer, tgid, integer, techid, integer, deptid, integer, and firstn, chara. The table contains 7 rows of data.

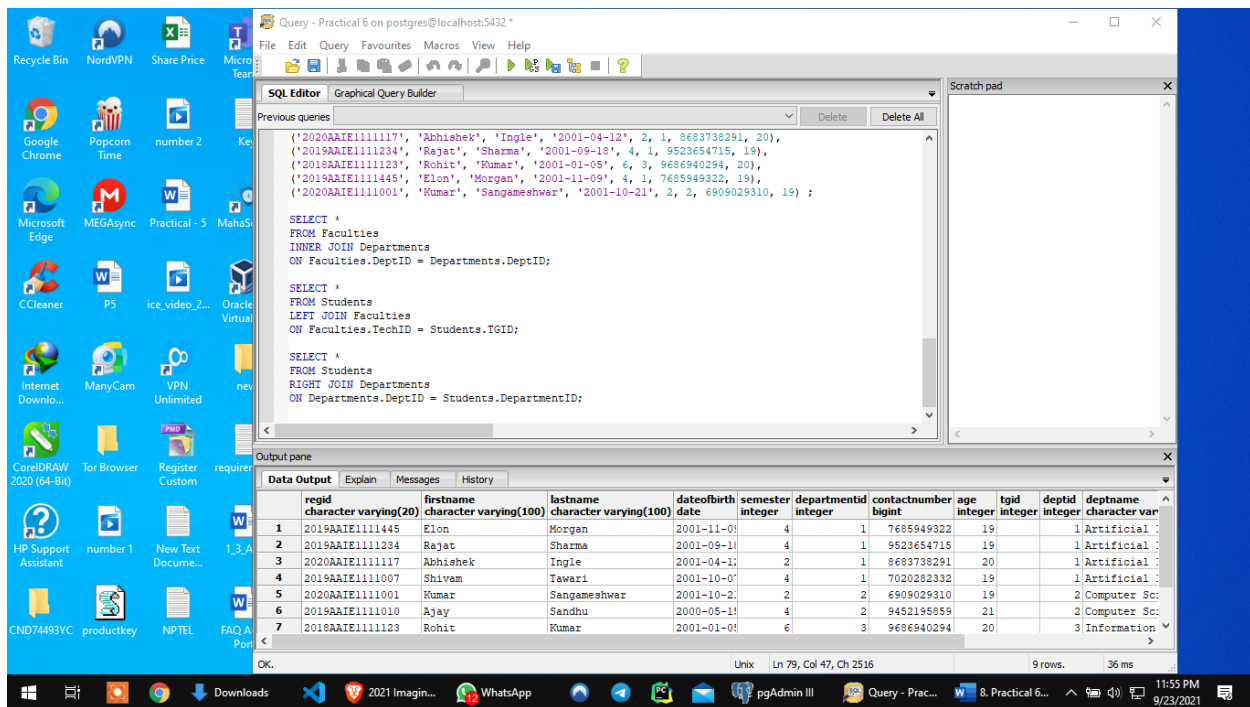
regid	character varying(20)	firstname	character varying(100)	lastname	character varying(100)	dateofbirth	date	semester	integer	departmentid	integer	contactnumber	bigint	age	integer	tgid	integer	techid	integer	deptid	integer	firstn	chara
1	2020AAIE1111001	Kumar	Sangameshwar			2001-10-21		2		2		6909029310		19									
2	2019AAIE1111445	Elon	Morgan			2001-11-06		4		1		7685949322		19									
3	2018AAIE1111123	Rohit	Kumar			2001-01-06		6		3		9686940294		20									
4	2019AAIE1111234	Rajat	Sharma			2001-09-16		4		1		9523654715		19									
5	2020AAIE1111117	Abhishek	Ingle			2001-04-12		2		1		8683738291		20									
6	2020AAIE1111001	Kumar	Sangameshwar			2001-10-21		2		2		6909029310		19									

-- Operation 3: RIGHT JOIN or RIGHT OUTER JOIN

SQL Script:

```
SELECT *  
FROM Students  
RIGHT JOIN Departments  
ON Departments.DeptID = Students.DepartmentID;
```

Output:



-- Operation 4: FULL JOIN or FULL OUTER JOIN

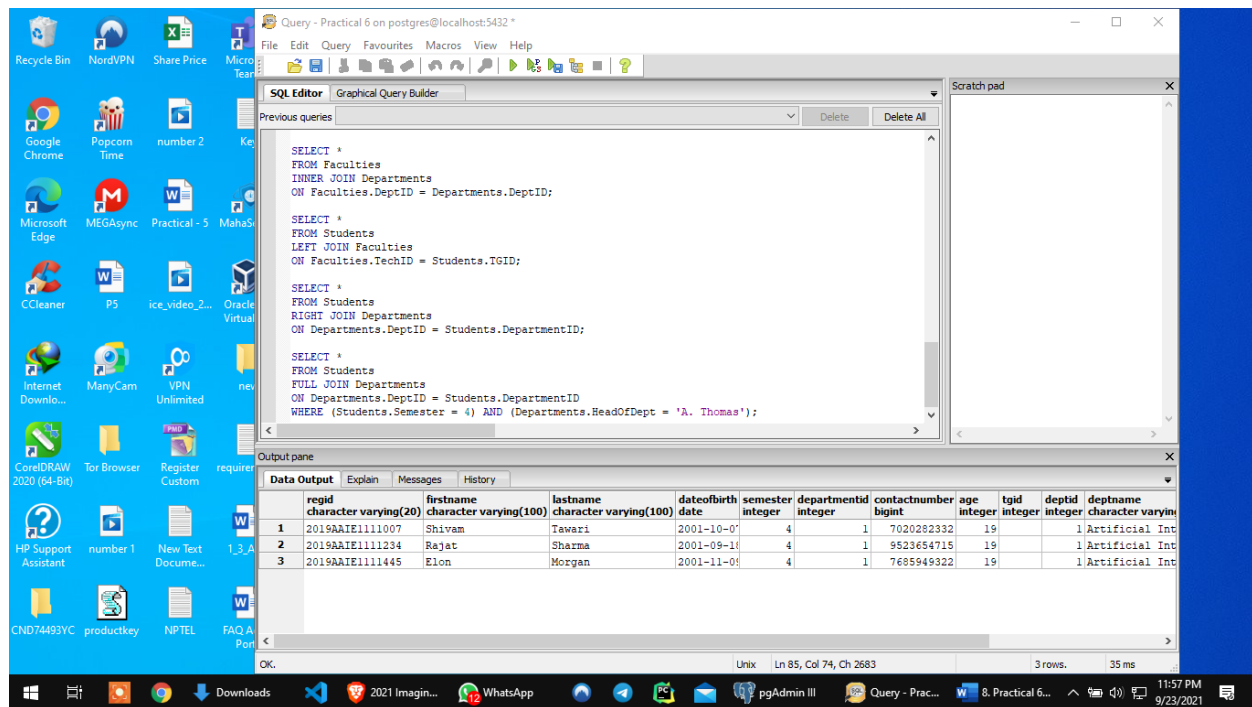
SQL Script:

```

SELECT *
FROM Students
FULL JOIN Departments
ON Departments.DeptID = Students.DepartmentID
WHERE (Students.Semester = 4) AND (Departments.HeadOfDept = 'A. Thomas');

```

Output:



-- Operation 5: NATURAL JOIN

SQL Script:

```
SELECT *
FROM Faculties
NATURAL JOIN Departments;
```

Output:

The screenshot shows a PostgreSQL SQL Editor window titled "Query - Practical 6 on postgres@localhost:5432". The SQL Editor contains the following query:

```

ON Faculties.DeptID = Departments.DeptID;

SELECT *
FROM Students
LEFT JOIN Faculties
ON Faculties.TechID = Students.TGID;

SELECT *
FROM Students
RIGHT JOIN Departments
ON Departments.DeptID = Students.DepartmentID;

SELECT *
FROM Students
FULL JOIN Departments
ON Departments.DeptID = Students.DepartmentID
WHERE (Students.Semester = 4) AND (Departments.HeadOfDept = 'A. Thomas');

SELECT *
FROM Faculties
NATURAL JOIN Departments;

```

The Output pane displays the results of the last query, showing a table with 10 columns: deptid, techid, firstname, lastname, contactno, dateofjoin, deptname, deptest, headofdept, and character varying(30). The table contains 5 rows of data.

deptid	techid	firstname	lastname	contactno	dateofjoin	deptname	deptest	headofdept	character varying(30)
1	1	101 Durgesh	Sharma	652718349	2021-01-01	(Artificial Intelli	2019-01-01	A. Thomas	
2	1	60 Achamma	Thomas	637363525	2010-03-03	(Artificial Intelli	2019-01-01	A. Thomas	
3	2	65 Dipti	Theng	637829191	2013-01-01	(Computer Science	2010-01-01	Soehlata Dongre	
4	3	200 Praveen	Ghatode	65342791	2012-01-01	(Information Techno	2009-01-01	Mahendra Gaikwad	
5	4	201 Gopal	Sakarker	687772901	2014-01-01	(Data Science	2021-01-01	A. Thomas	

Conclusion: Hence, we have learnt and performed data retrieval SQL query on multiple tables also learned use of JOINS Operation on various tables.