



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

**Batch: B1      Roll No.: 1711072**

**Experiment / assignment / tutorial No. 5**

**Grade: AA / AB / BB / BC / CC / CD / DD**

**Title: Queries based on Joins, and Views**

**Objective:** To be able to use SQL JOIN clause to extract data from 2 (or more) tables, we need a relationship between certain columns in these tables.

**Expected Outcome of Experiment:**

CO 2: Convert entity-relationship diagrams into relational tables, populate a relational database and formulate SQL queries on the data Use SQL for creation and query the database.

CO 3: Define and apply integrity constraints and improve database design using normalization techniques.

**Books/ Journals/ Websites referred:**

1. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g.Black book, Dreamtech Press
2. www.db-book.com
3. Korth, Slberchatz, Sudarshan : “Database Systems Concept”, 5<sup>th</sup> Edition , McGraw Hill
4. Elmasri and Navathe,”Fundamentals of database Systems”, 4<sup>th</sup> Edition,PEARSON Education.

**Resources used:** Postgresql

**Theory**

Join is a combination of a Cartesian product followed by a selection process. A Join operation pairs two tuples from different relations, if and only if a given join condition is satisfied. Or JOINS are used to retrieve data from multiple tables. A JOIN is performed whenever two or more tables are joined in a SQL statement.



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

There are different types of Joins:

- The CROSS JOIN
- The INNER JOIN
- The LEFT OUTER JOIN
- The RIGHT OUTER JOIN
- The FULL OUTER JOIN

A **CROSS JOIN** matches every row of the first table with every row of the second table. If the input tables have x and y columns, respectively, the resulting table will have x+y columns. Because CROSS JOINS have the potential to generate extremely large tables, care must be taken to use them only when appropriate.

**Ex.** SELECT EMP\_ID, NAME, DEPT FROM COMPANY CROSS JOIN DEPARTMENT;

A **INNER JOIN** creates a new result table by combining column values of two tables (table1 and table2) based upon the join-predicate. The query compares each row of table1 with each row of table2 to find all pairs of rows, which satisfy the join-predicate. When the join-predicate is satisfied, column values for each matched pair of rows of table1 and table2 are combined into a result row.

**Ex.** SELECT EMP\_ID, NAME, DEPT FROM COMPANY INNER JOIN DEPARTMENT ON COMPANY.ID = DEPARTMENT.EMP\_ID;

The **OUTER JOIN** is an extension of the INNER JOIN. SQL standard defines three types of OUTER JOINS: LEFT, RIGHT, and FULL and PostgreSQL supports all of these.

In case of **LEFT OUTER JOIN**, an inner join is performed first. Then, for each row in table T1 that does not satisfy the join condition with any row in table T2, a joined row is added with null values in columns of T2. Thus, the joined table always has at least one row for each row in T1.

**Ex.** SELECT EMP\_ID, NAME, DEPT FROM COMPANY LEFT OUTER JOIN DEPARTMENT ON COMPANY.ID = DEPARTMENT.EMP\_ID;

### **The RIGHT OUTER JOIN**

First, an inner join is performed. Then, for each row in table T2 that does not satisfy the join condition with any row in table T1, a joined row is added with null values in



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

columns of T1. This is the converse of a left join; the result table will always have a row for each row in T2.

Ex. `SELECT EMP_ID, NAME, DEPT FROM COMPANY RIGHT OUTER JOIN DEPARTMENT ON COMPANY.ID = DEPARTMENT.EMP_ID;`

**The FULL OUTER JOIN**

First, an inner join is performed. Then, for each row in table T1 that does not satisfy the join condition with any row in table T2, a joined row is added with null values in columns of T2. In addition, for each row of T2 that does not satisfy the join condition with any row in T1, a joined row with null values in the columns of T1 is added.

`SELECT EMP_ID, NAME, DEPT FROM COMPANY FULL OUTER JOIN DEPARTMENT ON COMPANY.ID = DEPARTMENT.EMP_ID;`

**Views** are pseudo-tables. That is, they are not real tables; nevertheless appear as ordinary tables to `SELECT`. A view can represent a subset of a real table, selecting certain columns or certain rows from an ordinary table. A view can even represent joined tables. Because views are assigned separate permissions, you can use them to restrict table access so that the users see only specific rows or columns of a table.

A view can contain all rows of a table or selected rows from one or more tables. A view can be created from one or many tables, which depends on the written PostgreSQL query to create a view.

Views, which are kind of virtual tables, allow users to do the following –

- Structure data in a way that users or classes of users find natural or intuitive.
- Restrict access to the data such that a user can only see limited data instead of complete table.
- Summarize data from various tables, which can be used to generate reports.

Since views are not ordinary tables, you may not be able to execute a `DELETE`, `INSERT`, or `UPDATE` statement on a view. However, you can create a `RULE` to correct this problem of using `DELETE`, `INSERT` or `UPDATE` on a view.

Syntax

`CREATE [TEMP | TEMPORARY] VIEW view_name AS`

`SELECT column1, column2.....`

`FROM table_name`

`WHERE [condition];`



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

Ex

```
CREATE VIEW COMPANY_VIEW AS  
SELECT ID, NAME, AGE  
FROM COMPANY;
```

Dropping Views

Syntax: DROP VIEW view\_name;

**Implementation:**

```
select * from movie join review on movie_id = mov_id  
  
select * from movie inner join review on movie_id = mov_id  
  
select * from movie full outer join review on movie_id = mov_id  
  
select * from movie left outer join review on movie_id = mov_id  
  
select * from movie right outer join review on movie_id = mov_id  
  
select * from movie natural join review  
  
select * from movie cross join cast_movie  
  
select * from movie t, movie m where  
m.movie_name=t.movie_name ;self join  
  
create or replace view reviewer as select * from movie join review  
on movie_id = mov_id
```



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

**Screenshots:**

Previous queries

```
select * from movie join review on movie_id = mov_id
```

Output pane

Data Output Explain Messages History

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	1892	Amazing thriller experience. L
2	1892	2012	2008-04-02	3.8	Thriller	1892	Absolutely loved it.
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary

Previous queries

```
select * from movie inner join review on movie_id = mov_id
```

Output pane

Data Output Explain Messages History

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	1892	Amazing thriller experience. L
2	1892	2012	2008-04-02	3.8	Thriller	1892	Absolutely loved it.
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in t
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary

```
select * from movie full outer join review on movie_id = mov_id
```

Output pane

Data Output Explain Messages History

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	1892	Amazing thriller experie
2	1892	2012	2008-04-02	3.8	Thriller	1892	Absolutely loved it.
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary
8	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy		
9	5482	MSD-An Untold Story	2017-10-20	4.8	Biography		
10	1000	New York	2009-09-10	3.5	Adventure		
11	5017	The Man Who Knew Infinity	2017-09-06	4.8	Biography		
12	4013	Stephen Hawking	2013-08-06	4.4	Biography		



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

```
select * from movie natural join review
```

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	cust_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	Amazing thriller experience. Loved it.
2	1892	2012	2008-04-02	3.8	Thriller	Absolutely loved it.
3	3281	Welcome Back	2012-09-06	3.5	Comedy	John did a fantastic role in the movie
4	3281	Welcome Back	2012-09-06	3.5	Comedy	John did a fantastic role in the movie
5	3281	Welcome Back	2012-09-06	3.5	Comedy	John did a fantastic role in the movie
6	9882	Annabelle	2016-10-20	4.2	Horror	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	Very scary

```
select * from movie cross join cast_movie
```

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	cast_id integer	name character varying(30)
1	1892	2012	2008-04-02	3.8	Thriller	1	Shah Rukh Khan
2	1892	2012	2008-04-02	3.8	Thriller	4	Aamir Khan
3	1892	2012	2008-04-02	3.8	Thriller	2	Salmaan Khan
4	1892	2012	2008-04-02	3.8	Thriller	5	Emma Watson
5	1892	2012	2008-04-02	3.8	Thriller	6	Deepika Padukone
6	1892	2012	2008-04-02	3.8	Thriller	7	Anushka Sharma
7	1892	2012	2008-04-02	3.8	Thriller	8	Dwayne The Rock Johnson
8	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy	1	Shah Rukh Khan
9	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy	4	Aamir Khan
10	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy	2	Salmaan Khan
11	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy	5	Emma Watson

```
create or replace view reviewer as select * from movie join review on movie.movie_id=review.mov_id  
select * from reviewer
```

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)	analyst_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	1892	Amazing thriller experience. Loved it.	Excellent acting. Had me at the
2	1892	2012	2008-04-02	3.8	Thriller	1892	Absolutely loved it.	Excellent acting. The storyline
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in the movie.	Justice is not done to the prequ
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in the movie.	Justice is not done to the prequ
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in the movie.	Excellent acting. The storyline
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.	Perfect film to watch at 1 AM.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary	Good job. Expecting a sequel to

```
select * from movie right outer join review on movie_id = mov_id
```

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)
1	1892	2012	2008-04-02	3.8	Thriller	1892	Amazing thriller experience. Lo
2	1892	2012	2008-04-02	3.8	Thriller	1892	Absolutely loved it.
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in th
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in th
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic role in th
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary



**K. J. Somaiya College of Engineering, Mumbai-77**  
(Autonomous College Affiliated to University of Mumbai)

```
select * from movie t, movie m where m.movie_name=t.movie_name
```

Output pane

Data Output Explain Messages History

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	movie_id integer	movie_name character varying(60)
1	1892	2012	2008-04-02	3.8	Thriller	1892	2012
2	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy	2091	Mr. Bean-The Painting
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	Welcome Back
4	4011	Singham Returns	2012-09-06	4.4	Action	4011	Singham Returns
5	5017	The Man Who Knew Infinity	2017-09-06	4.8	Biography	5017	The Man Who Knew Infini
6	5482	MSD-An Untold Story	2017-10-20	4.8	Biography	5482	MSD-An Untold Story
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Annabelle
8	2122	Stree	2018-11-20	3.2	Horror	2122	Stree
9	1000	New York	2009-09-10	3.5	Adventure	1000	New York

```
select * from movie left outer join review on movie_id = mov_id
```

Output pane

Data Output Explain Messages History

	movie_id integer	movie_name character varying(60)	release_date date	rating double precision	genre character varying(15)	mov_id integer	cust_review character varying(200)
3	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
4	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
5	3281	Welcome Back	2012-09-06	3.5	Comedy	3281	John did a fantastic rol
6	9882	Annabelle	2016-10-20	4.2	Horror	9882	I almost freaked out.
7	9882	Annabelle	2016-10-20	4.2	Horror	9882	Very scary
8	2091	Mr. Bean-The Painting	2012-03-08	4.1	Comedy		
9	5482	MSD-An Untold Story	2017-10-20	4.8	Biography		
10	1000	New York	2009-09-10	3.5	Adventure		
11	5017	The Man Who Knew Infinity	2017-09-06	4.8	Biography		
12	4011	Singham Returns	2012-09-06	4.4	Action		
13	2122	Stree	2018-11-20	3.2	Horror		

**Conclusion:** Various kinds of join operations were performed and views were created.