OUTER JOINS

```
select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   6 | kshitij  |     24 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
(5 rows)

select * from department;
 deptid | deptname
--------+----------
    101 | java
    202 | python
    303 | php
    404 | Fresher
    505 | Trainee
(5 rows)
```

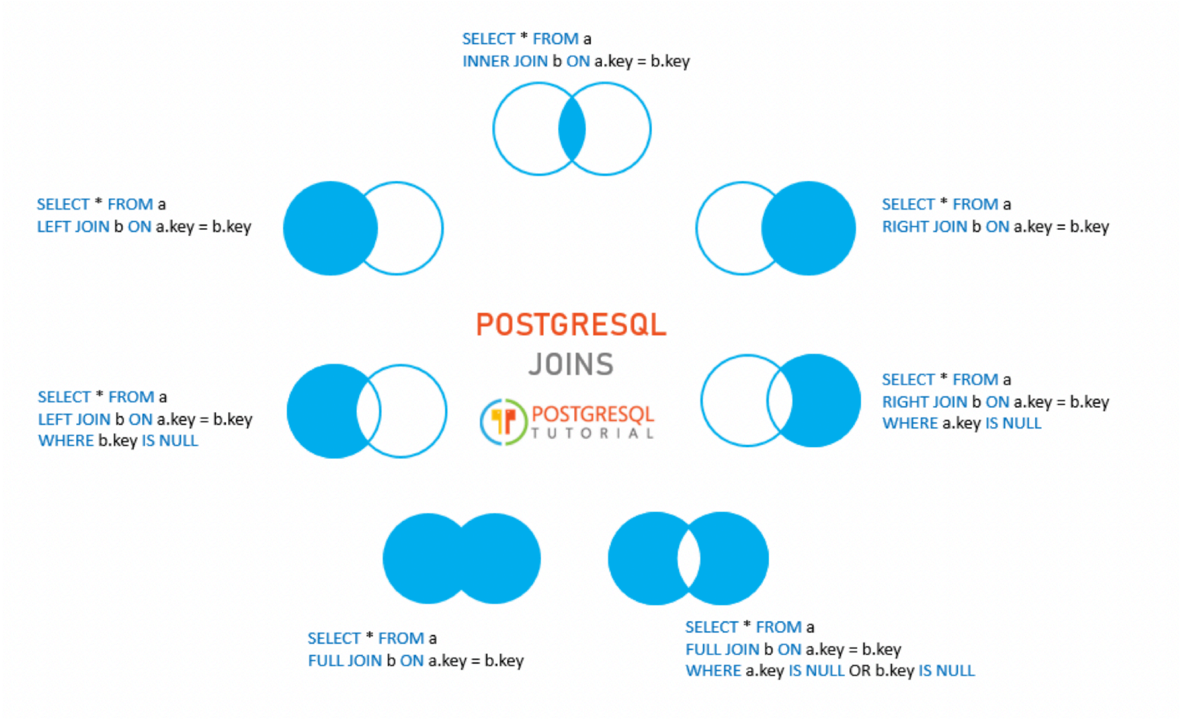
Right Join

:- all values in right table and column values from left table are retrieved.

```
select eid, ename, a.deptid, deptname from employee a right join
department b on a.deptid = b.deptid;
 eid |  ename   | deptid | deptname
-----+----------+--------+----------
   2 | hara     |    101 | java
   6 | kshitij  |    101 | java
   1 | uttej    |    202 | python
   3 | data     |    202 | python
   4 | vishnu   |    303 | php
     |          |        | Trainee
     |          |        | Fresher
```

Left Outer join :- All values from left table and common values from right table are retrieved;

```
dbfinserv=# select eid, ename, a.deptid, deptname from employee a
left  join department b on a.deptid = b.deptid;
 eid |  ename   | deptid | deptname
-----+----------+--------+----------
   2 | hara     |    101 | java
   6 | kshitij  |    101 | java
   1 | uttej    |    202 | python
   3 | data     |    202 | python
   4 | vishnu   |    303 | php
```


SELECT * FROM a
INNER JOIN b ON a.key = b.key
SELECT * FROM a
LEFT JOIN b ON a.key = b.key
SELECT * FROM a
RIGHT JOIN b ON a.key = b.key
POSTGRESQL
JOINS
POSTGRESQL TUTORIAL
SELECT * FROM a
LEFT JOIN b ON a.key = b.key
WHERE b.key IS NULL
SELECT * FROM a
RIGHT JOIN b ON a.key = b.key
WHERE a.key IS NULL
SELECT * FROM a
FULL JOIN b ON a.key = b.key
SELECT * FROM a
FULL JOIN b ON a.key = b.key
WHERE a.key IS NULL OR b.key IS NULL

Correlated sub query :- first outer query is executed then inner query is executed

```
dbfinserv=# select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   6 | kshitij  |     24 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
(5 rows)

dbfinserv=# select eid,ename, salary from employee e1 where
salary > (select avg(salary) from employee where deptid =
e1.deptid);
 eid |  ename   | salary
-----+----------+--------
   6 | kshitij  |     24
   3 | data     |     13
(2 rows)
```

Retrieval of salary which Is greater than avg salary within the same department;

```
dbfinserv=# select eid,ename, salary from employee e1 where
```

```
salary = (select avg(salary) from employee where eid =
e1.eid);
 eid |  ename   | salary
-----+----------+--------
   2 | hara     |     19
   6 | kshitij |     24
   1 | uttej    |     10
   3 | data     |     13
   4 | vishnu   |     18
(5 rows)
```

Here eid is primary so it is unique, different eid has a particular salary thats why all records are retrieved.

TRANSACTION MANAGEMENT

```
dbfinserv=# select * from employee orderby(eid);
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(5 rows)

dbfinserv=# begin;
BEGIN
dbfinserv=*# delete from employee where eid = 2;
DELETE 1
dbfinserv=*# delete from employee where eid = 1;
DELETE 1
dbfinserv=*# select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(3 rows)

dbfinserv=*# rollback;
ROLLBACK
dbfinserv=# select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(5 rows)
```

Using savepoints.

```
select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(5 rows)

dbfinserv=# rollback;
WARNING:  there is no transaction in progress
ROLLBACK
dbfinserv=# BEGIN;
BEGIN
dbfinserv=*# select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(5 rows)

dbfinserv=*# savepoint s1;
SAVEPOINT
dbfinserv=*# insert into employee values(6, 'samer',9,202);
INSERT 0 1
dbfinserv=*# select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
   6 | samer  |      9 |    202
(6 rows)

dbfinserv=*# rollback to s1;
ROLLBACK
dbfinserv=*# select * from employee;
 eid | ename  | salary | deptid
```

```
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
(5 rows)

dbfinserv=*# commit;
COMMIT


rollback to s1;
ERROR:  ROLLBACK TO SAVEPOINT can only be used in transaction
blocks #begin is must to use rollback;
```

EXAMPLE

```
dbfinserv=# begin;
BEGIN
dbfinserv=*# select * from employee;
 eid | ename  | salary | deptid
-----+--------+--------+--------
   2 | hara   |     19 |    101
   1 | uttej  |     10 |    202
   3 | data   |     13 |    202
   4 | vishnu |     18 |    303
   5 | rahul  |     16 |    505
   7 | shaik  |      9 |    303
(6 rows)

dbfinserv=*# insert into employee values
(9,'sankarsh',7,101);
INSERT 0 1
dbfinserv=*# savepoint s1;
SAVEPOINT
dbfinserv=*# insert into employee values (8,'sai',8,505);
INSERT 0 1
dbfinserv=*# savepoint s2;
SAVEPOINT
dbfinserv=*# select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
   5 | rahul    |     16 |    505
   7 | shaik    |      9 |    303
   9 | sankarsh |      7 |    101
   8 | sai      |      8 |    505
(8 rows)

dbfinserv=*# rollback to s2;
ROLLBACK
dbfinserv=*# select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
   5 | rahul    |     16 |    505
   7 | shaik    |      9 |    303
   9 | sankarsh |      7 |    101
   8 | sai      |      8 |    505
(8 rows)

dbfinserv=*# rollback to s1;
ROLLBACK
dbfinserv=*# select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
   5 | rahul    |     16 |    505
   7 | shaik    |      9 |    303
   9 | sankarsh |      7 |    101
(7 rows)

dbfinserv=*# commit;
COMMIT
dbfinserv=# select * from employee;
 eid |  ename   | salary | deptid
-----+----------+--------+--------
   2 | hara     |     19 |    101
   1 | uttej    |     10 |    202
   3 | data     |     13 |    202
   4 | vishnu   |     18 |    303
   5 | rahul    |     16 |    505
   7 | shaik    |      9 |    303
   9 | sankarsh |       7 |     101
(7 rows)

dbfinserv=# rollback;
WARNING:  there is no transaction in progress
ROLLBACK
```

INDEXING IMPORTANCE

```
dbfinserv=# EXPLAIN select * from employee where ename =
'uttej';
                        QUERY PLAN
---------------------------------------------------------
-
 Seq Scan on employee  (cost=0.00..24.12 rows=6 width=44)
   Filter: (ename = 'uttej'::text)
(2 rows)
```

# indexing command

```
dbfinserv=# create index enameindex on employee(ename);
CREATE INDEX


dbfinserv=# explain select * from employee where ename =
'sankarsh';
                       QUERY PLAN
---------------------------------------------------------
 Seq Scan on employee  (cost=0.00..1.09 rows=1 width=44)
   Filter: (ename = 'sankarsh'::text)
(2 rows)
```

DROPING INDEX

```
dbfinserv=# drop index enameindex;
DROP INDEX
dbfinserv=# explain select * from employee where ename =
'rahul';
                       QUERY PLAN
---------------------------------------------------------
 Seq Scan on employee  (cost=0.00..1.09 rows=1 width=44)
   Filter: (ename = 'rahul'::text)
(2 rows)

dbfinserv=# explain select * from employee where ename =
'hara';
```

```
                              QUERY PLAN
---------------------------------------------------------
 Seq Scan on employee  (cost=0.00..1.09 rows=1 width=44)
   Filter: (ename = 'hara'::text)
(2 rows)
```

VIEWS

```
dbfinserv=# CREATE view empsalaryview as select ename,
salary from employee;
CREATE VIEW
dbfinserv=# select * from empsalaryview;
  ename   | salary
----------+--------
 hara     |     19
 uttej    |     10
 data     |     13
 vishnu   |     18
 rahul    |     16
 shaik    |      9
 sankarsh |      7
(7 rows)

dbfinserv=# insert into empsalaryview
values('yashaswi',20);
ERROR:  null value in column "eid" of relation "employee"
violates not-null constraint
DETAIL:  Failing row contains (null, yashaswi, 20, null).
dbfinserv=# select * from empsalaryview;
  ename   | salary
----------+--------
 hara     |     19
 uttej    |     10
 data     |     13
 vishnu   |     18
 rahul    |     16
 shaik    |      9
 sankarsh |      7
(7 rows)
```

Unless there is a primary key in view we can insert values. If primary key is not available in view we can only update or delete in the view and it can reflect in the main table

```
dbfinserv=# select * from employee;
 eid |   ename    | salary | deptid
-----+------------+--------+--------
   3 | data       |     13 |    202
   4 | vishnu     |     18 |    303
   7 | shaik      |      9 |    303
   9 | sankarsh   |      7 |    101
   1 | sai uttej  |     10 |    202
   2 | hara teja  |     19 |    101
   5 | data       |     16 |    505
(7 rows)

dbfinserv=# update empsalaryview set ename = 'data
aditya' where ename = 'data';
UPDATE 2
dbfinserv=# select * from employee;
 eid |    ename    | salary | deptid
-----+-------------+--------+--------
   4 | vishnu      |     18 |    303
   7 | shaik       |      9 |    303
   9 | sankarsh    |      7 |    101
   1 | sai uttej   |     10 |    202
   2 | hara teja   |     19 |    101
   3 | data aditya |     13 |    202
   5 | data aditya |     16 |    505
(7 rows)
```

If there are duplicates in the view, updation duplication will be occurred in original table refer to above example.