**SQL Window Function**

# Window Function

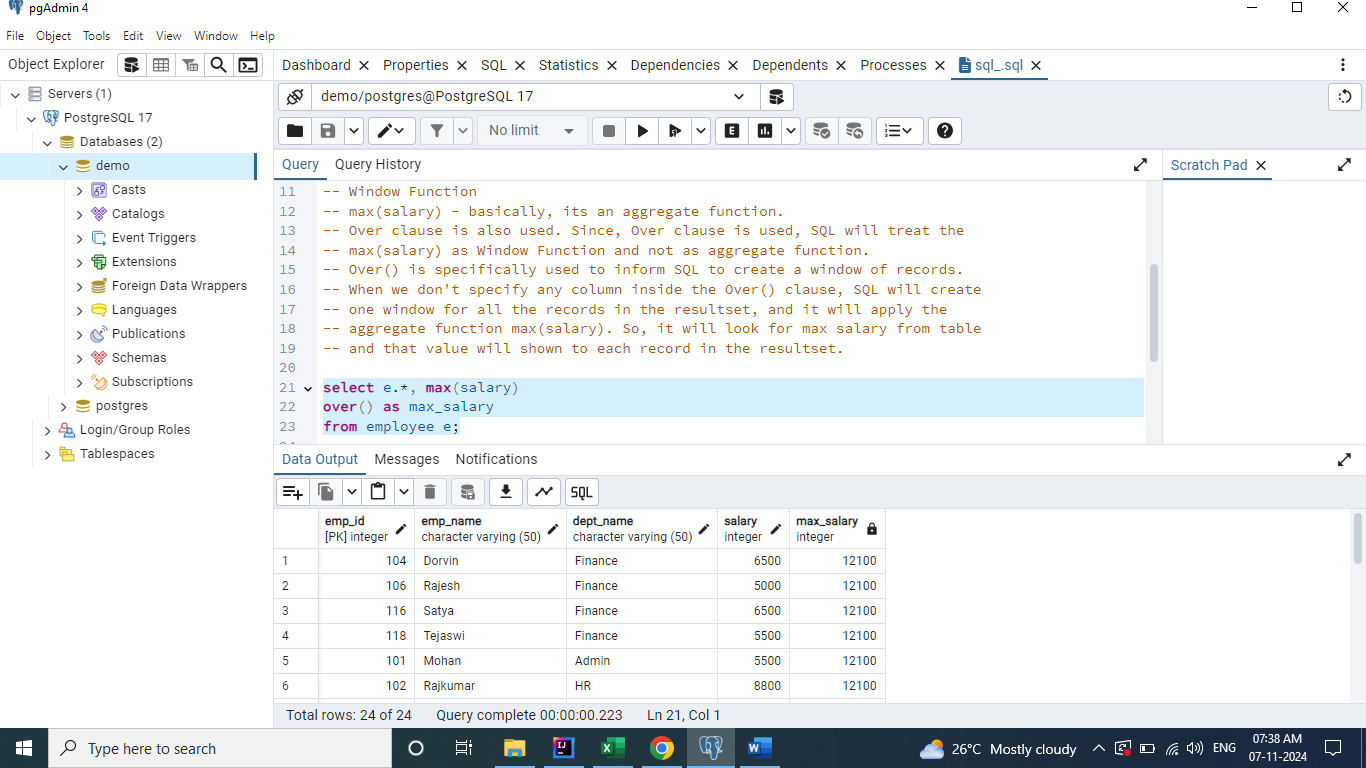
max(salary) - basically, it’s an aggregate function. Over clause is also used. Since, over clause is used, SQL will treat the max(salary) as Window Function and not as aggregate function.

Over() is specifically used to inform SQL to create a window of records. When we don't specify any column inside the Over() clause, SQL will create one window for all the records in the resultset, and it will apply the aggregate function max(salary). So, it will look for max salary from table and that value will show to each record in the resultset.

**select e.\*, max(salary)**

**over() as max\_salary**

**from employee e;**

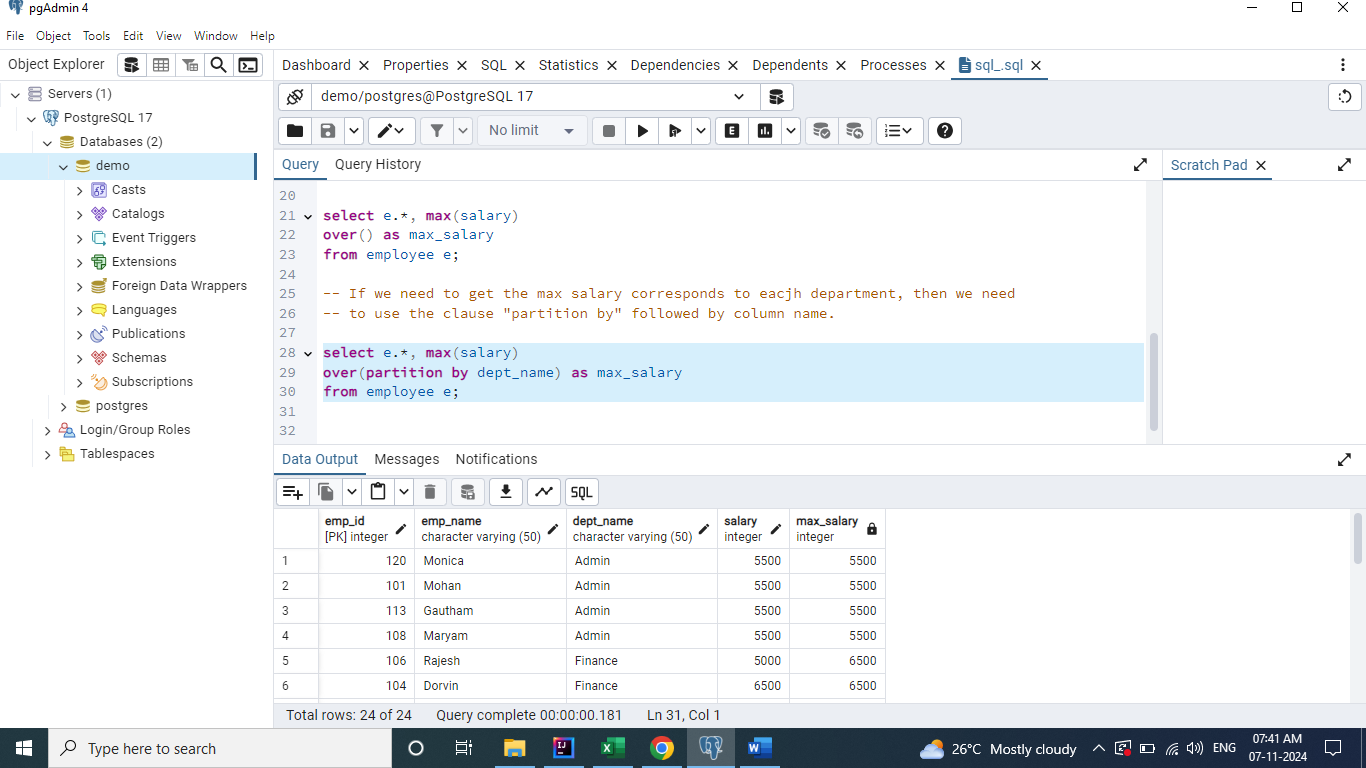


If we need to get the max salary corresponds to each department, then we need to use the clause "partition by" followed by column name.

**select e.\*, max(salary)**

**over(partition by dept\_name) as max\_salary**

**from employee e;**



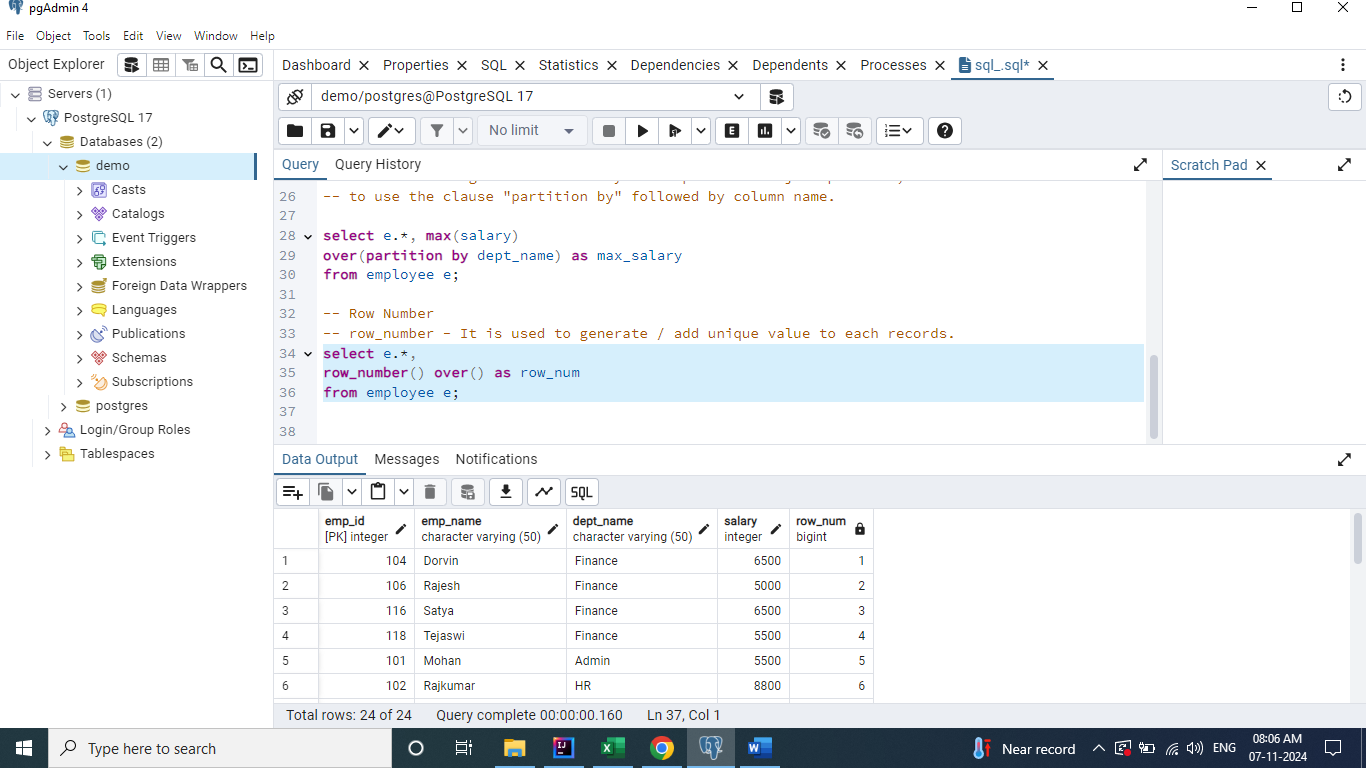
# Row Number

**It is used to generate / add unique value to each record.**

**select e.\*,**

**row\_number() over() as row\_num**

**from employee e;**

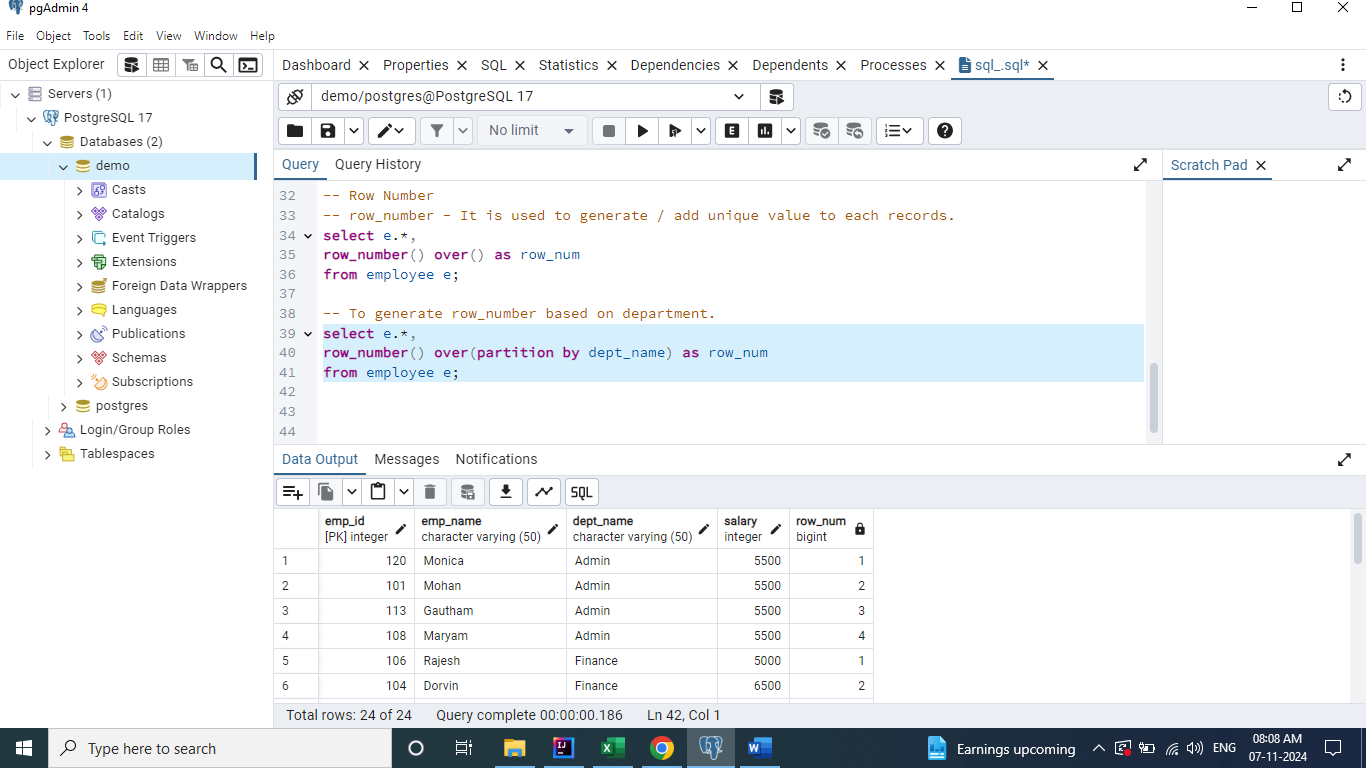


**To generate row\_number based on department.**

**select e.\*,**

**row\_number() over(partition by dept\_name) as row\_num**

**from employee e;**



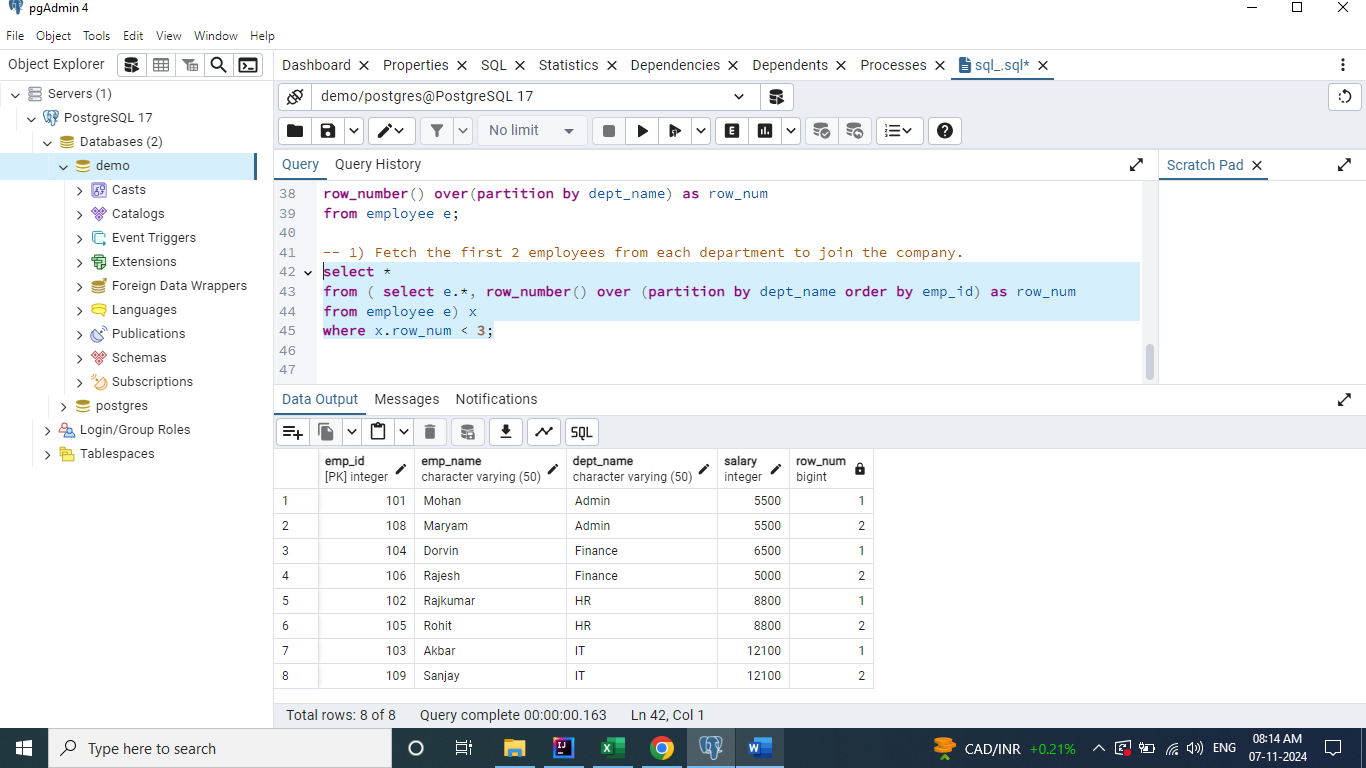
## 1) Fetch the first 2 employees from each department to join the company.

**select \***

**from ( select e.\*, row\_number() over (partition by dept\_name order by emp\_id) as row\_num**

**from employee e) x**

**where x.row\_num < 3;**



# Rank

## 2) Fetch the top 3 employees in each department earning the max salary.

**select \***

**from (select e.\*, rank() over (partition by dept\_name order by salary desc) as rank\_order**

**from employee e) x**

**where x.rank\_order < 4;**

