**Version: PostgreSQL13**

1. *create table PERSON (id serial, first\_name varchar (50), last\_name varchar (50), birth\_date date, gender varchar (10), salary numeric (8,2), primary key(id));*

2. *insert into PERSON*

*(first\_name, last\_name, birth\_date, gender, salary)*

*values*

*('John','Doe','01-01-2020','Male',10000.50),*

*('Mary','Jane','29-02-2020','Female',5000.12);*

3. -- Comment: Function to insert random values from the provided array list

*CREATE FUNCTION randomValueFromList (valueList IN TEXT[])*

*RETURNS TEXT AS*

*$$*

*WITH base AS (*

*SELECT val*

*FROM UNNEST(valueList) val*

*)*

*SELECT val*

*FROM base*

*ORDER BY RANDOM()*

*LIMIT 1*

*$$*

*LANGUAGE 'sql'*

*VOLATILE;*

-- Comment: Scripts to insert 1 million records in PERSON table

*insert into PERSON*

*(id, first\_name, last\_name, gender, birth\_date, salary)*

*select*

*generate\_series (3,1000000) as id,*

*randomValueFromList(Array['Aiden','Anika','Ariya','Ashanti','Avery','Cameron','Ceri','Che','Danica','Darcy''Dion','Eman','Eren','Esme','Frankie','Gurdeep','Haiden','Indi','Isa','Jaskaran','Jaya','Jo','Jodie','Kacey','Kameron','Kayden','Keeley','Kenzie','Lucca','Macauley','Manraj','Nur','Oluwatobiloba','Reiss','Riley','Rima','Ronnie','Ryley','Sam','Sana','Shola','Sierra','Tamika','Taran','Teagan','Tia','Tiegan','Virginia','Zhane','Zion']) as first\_name,*

*randomValueFromList(Array['Ahmad','Andersen','Arias','Barlow','Beck','Bloggs','Bowes','Buck','Burris','Cano','Chaney','Coombes','Correa','Coulson','Craig','Frye','Hackett','Hale','Huber','Hyde','Irving','Joyce','Kelley','Kim','Larson','Lynn','Markham','Mejia','Miranda','Neal','Newton','Novak','Ochoa','Pate','Paterson','Pennington','Rubio','Santana','Schaefer','Schofield','Shaffer','Sweeney','Talley','Trevino','Tucker','Velazquez','Vu','Wagner','Walton','Woodward']) as last\_name,*

*randomValueFromList(Array['Male', 'Female'])as gender,*

*now() +random()\*(timestamp without time zone '1970-01-01'- timestamp without time zone'2070-12-31') as birth\_date,*

*random() \*100000-random()\*100 as salary;*

*commit;*

4. *(a) select \* from person where first\_name='John' and last\_name='Doe';*

*(b) select \* from person where gender='Female' and salary>5000.50*

*and birth\_date between to\_timestamp (946684800)::date and to\_timestamp(1609372800)::date;*

*(c) select count (\*), round(salary ,-3) as salary ,gender from person group by gender, round(salary,-3);*

5,6. Performance Optimization of data retrieval:

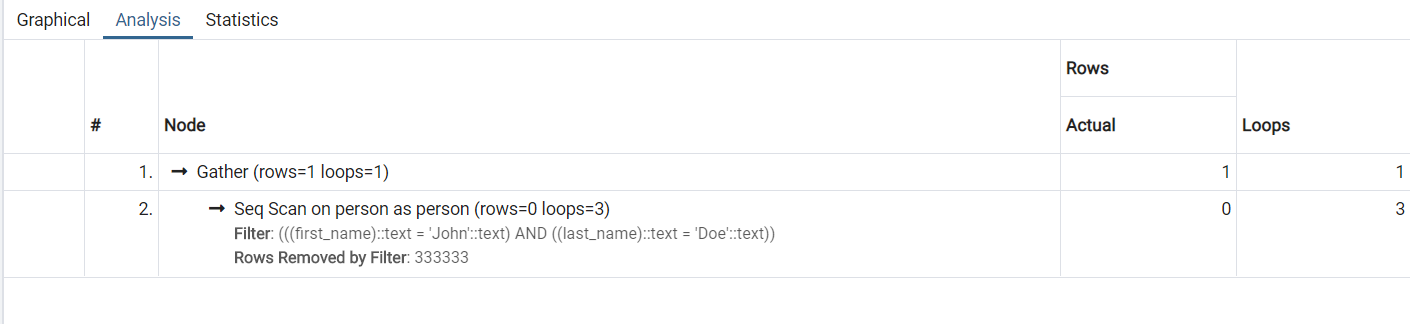
1. Creating Index:

*create index person\_name on person (last\_name, first\_name);*

Explanation: Creating an index on first\_name and last\_name column reduced the query execution time by reducing the number of loops and number of rows filtered.

Before creating index:

Query runtime: 1sec



After Index creation:

Query runtime: 89msec

