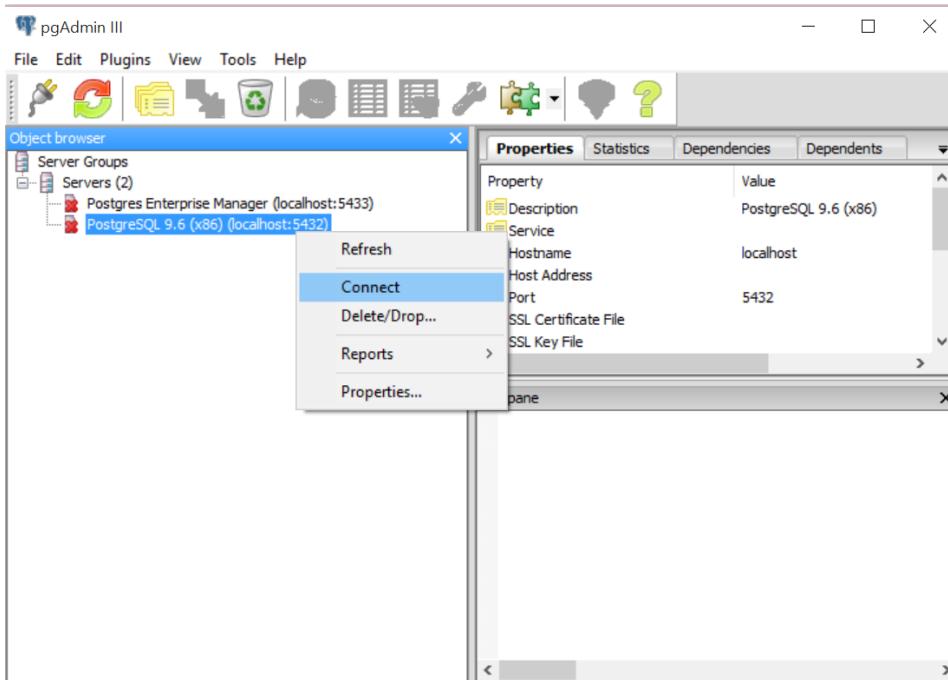


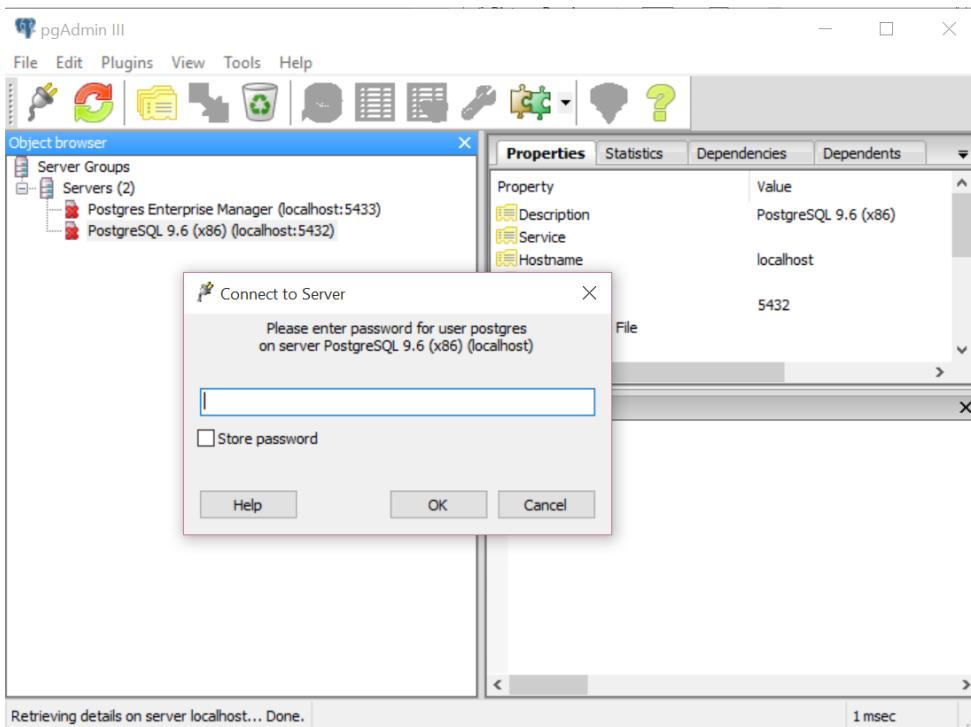
Tutorial 1

Part 1: Creating a table in a PostgreSQL database

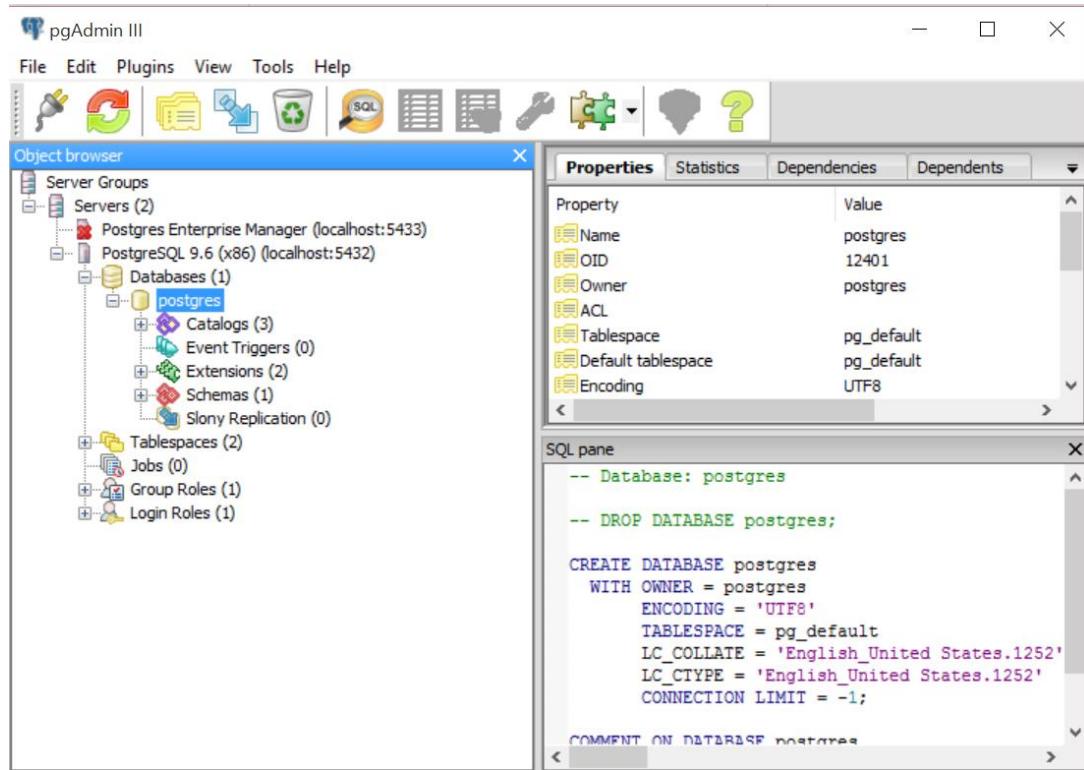
1. Right-click on PostgreSQL 9.6 (x86) (localhost: 5432)>>Connect



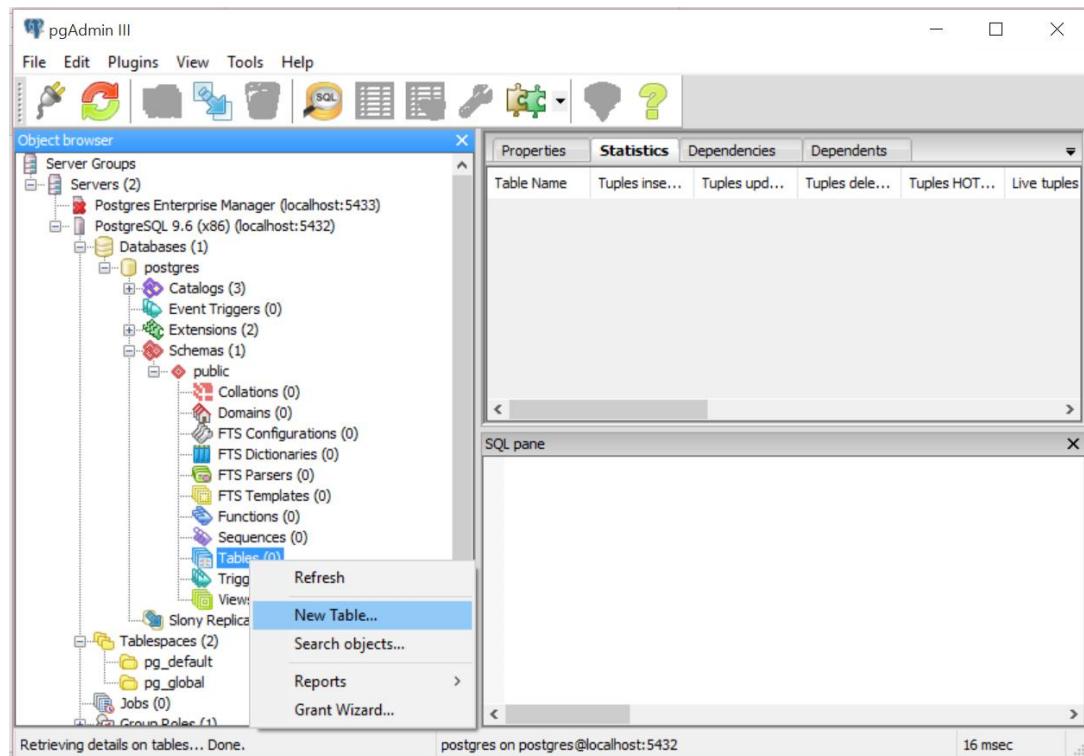
2. Enter the password and press "OK"



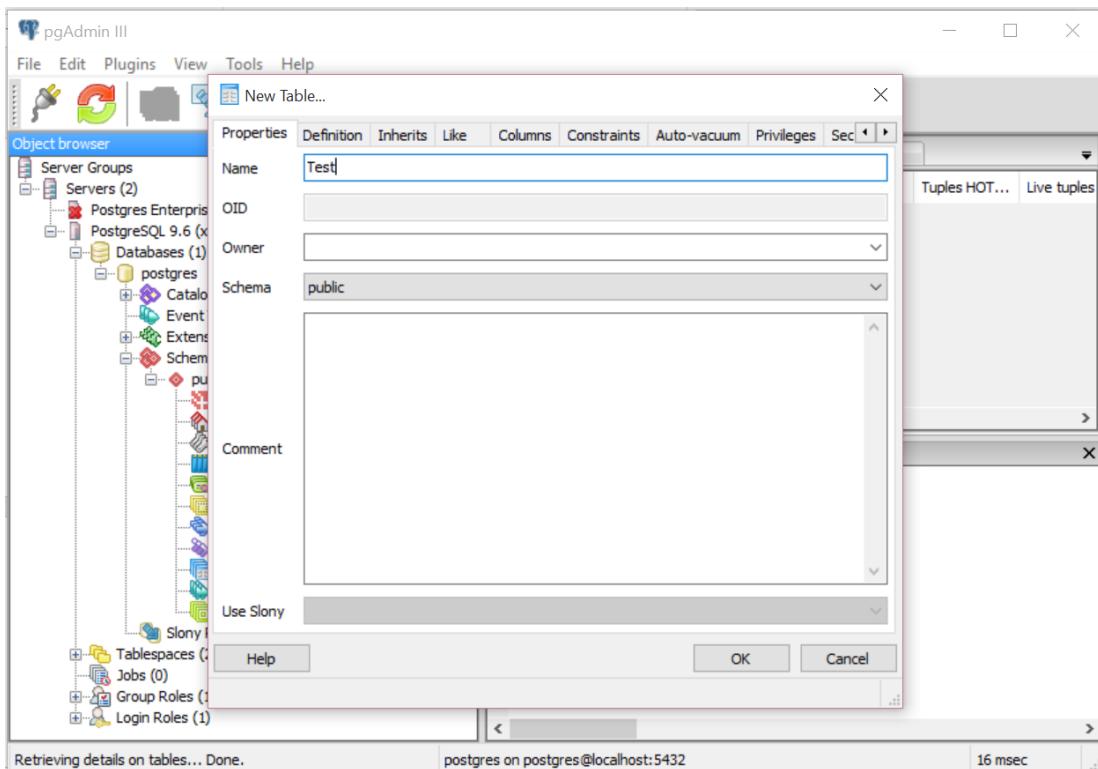
3. Now you are connected to PostgreSQL database server



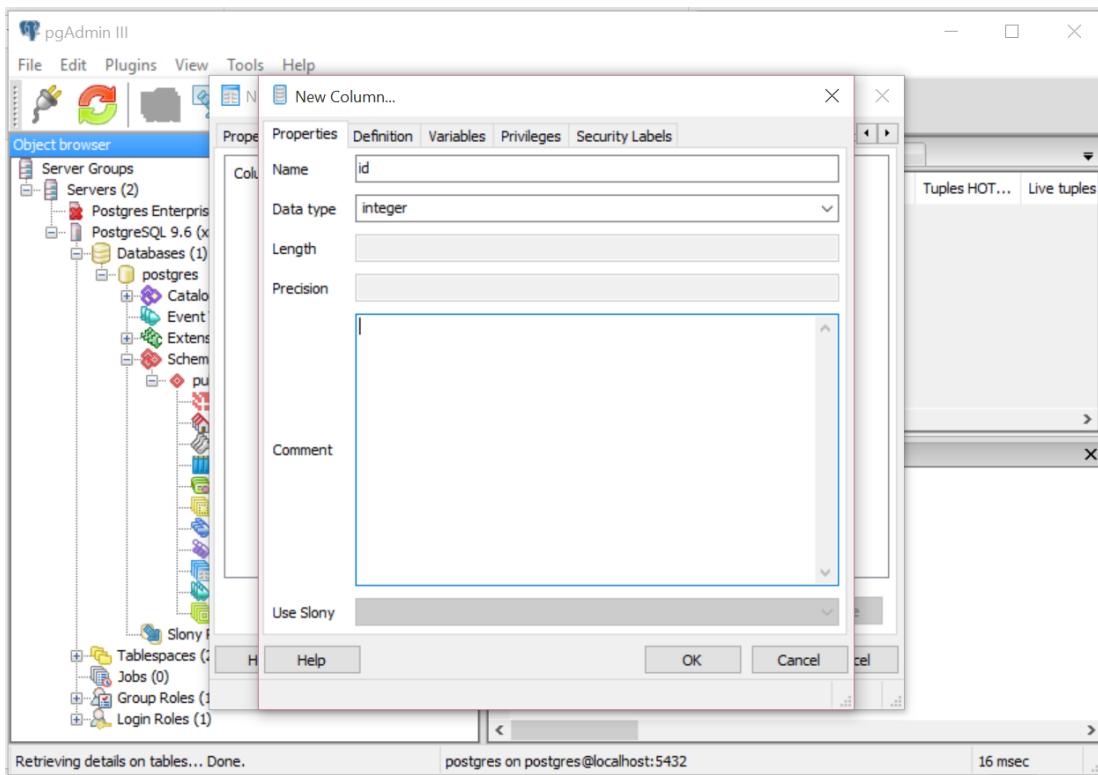
4. Create a new table under postgres database: right-click on “Tables”>>New Table OR select “Tables” node under Schemas>>public, then press this icon in the tools bar



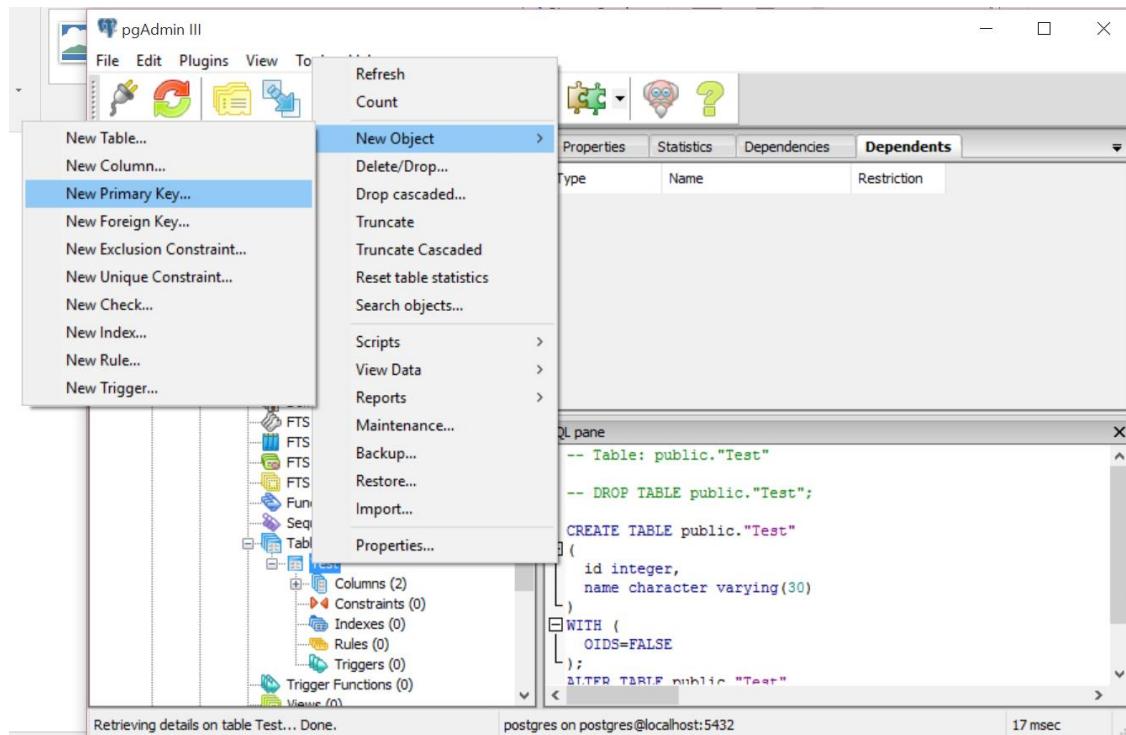
5. Give the table a name e.g., Test



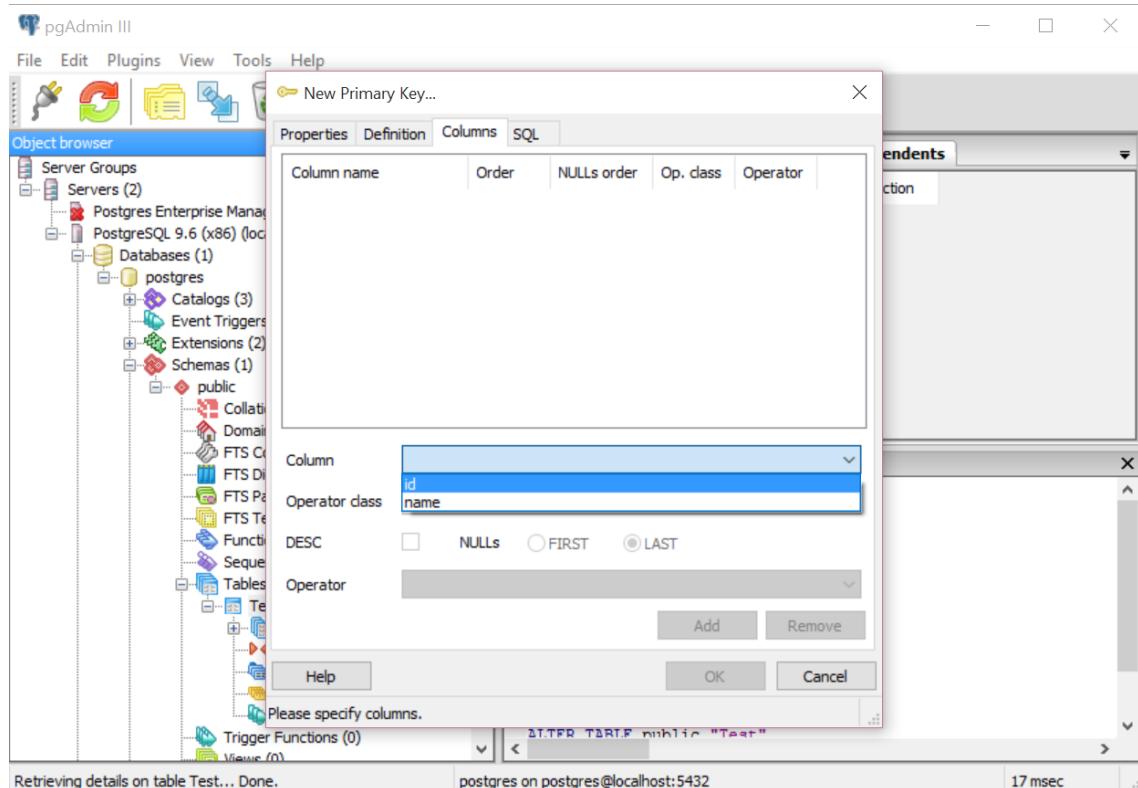
6. Go to Columns Tab and “Add” two columns “id” of type integer and “name” of type character varying (size 30)



7. Set constraint on column "id" to be a "primary key": right-click on table "Test"



8. At "Columns" Tab, select "id" and press "Add", then press "OK"



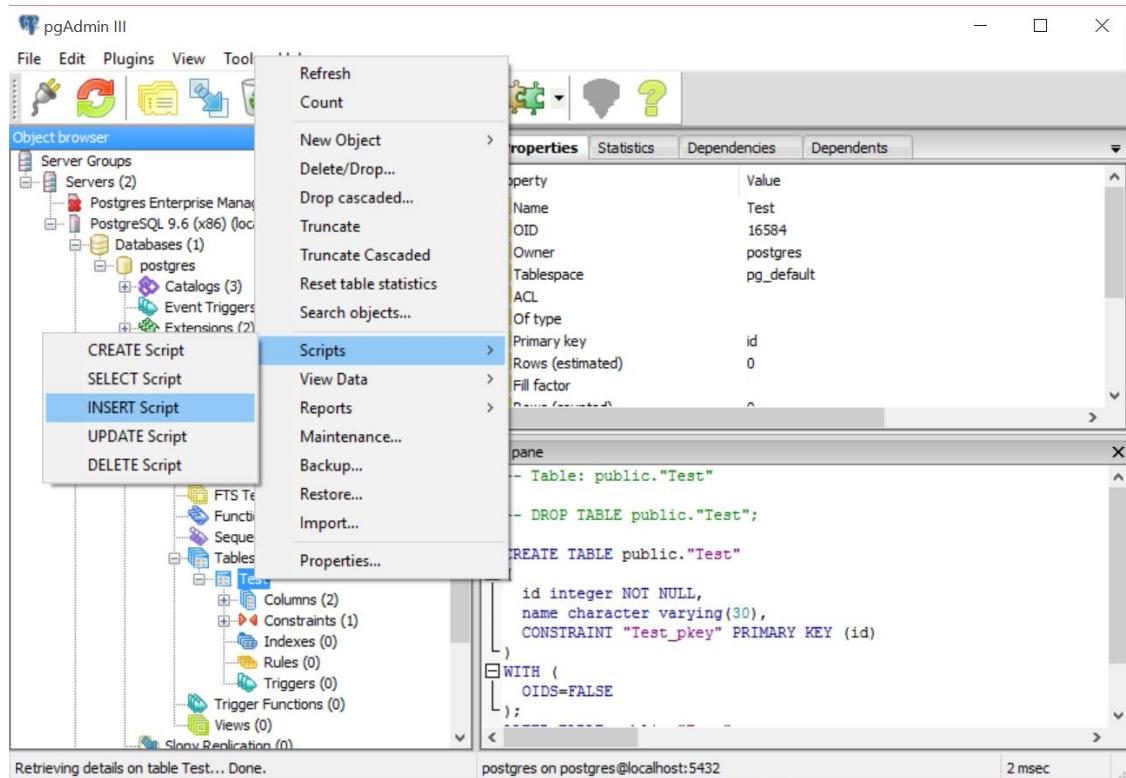
9. The result is shown below in the SQL Pane

```

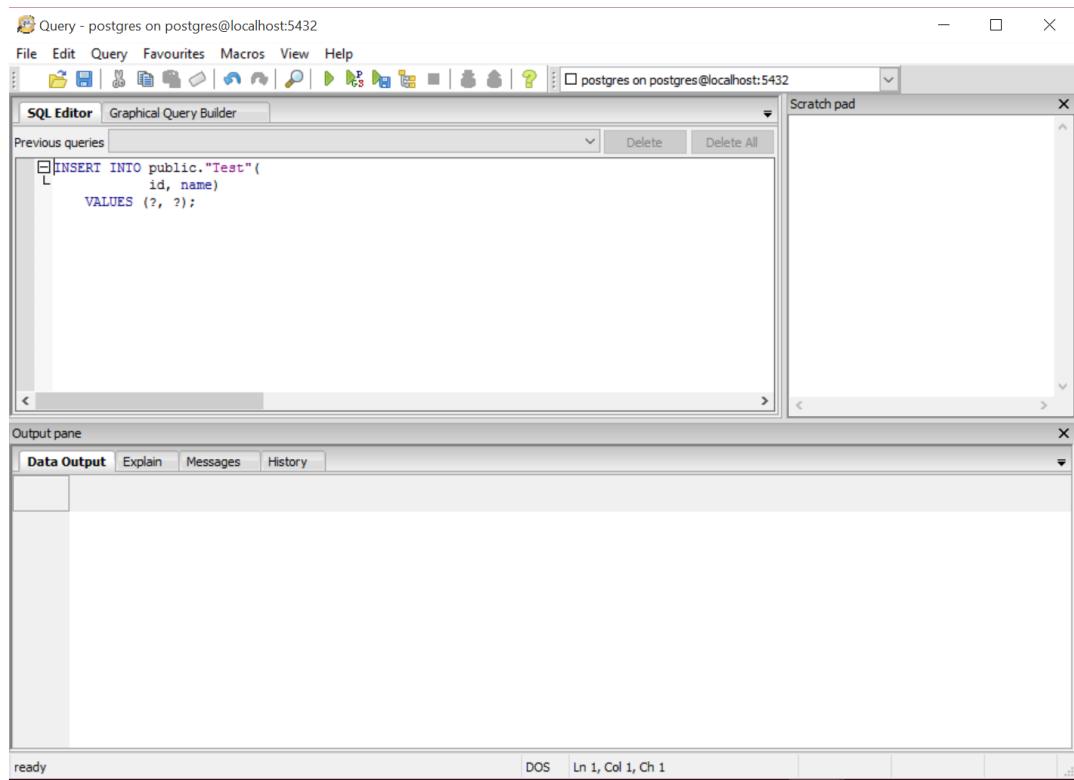
CREATE TABLE public."Test"
(
    id integer NOT NULL,
    name character varying(30),
    CONSTRAINT "Test_pkey" PRIMARY KEY (id)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE public."Test"
    OWNER TO postgres;

```

10. Insert a record into the table: right-click on “Test” table>>Scripts>>INSERT Script

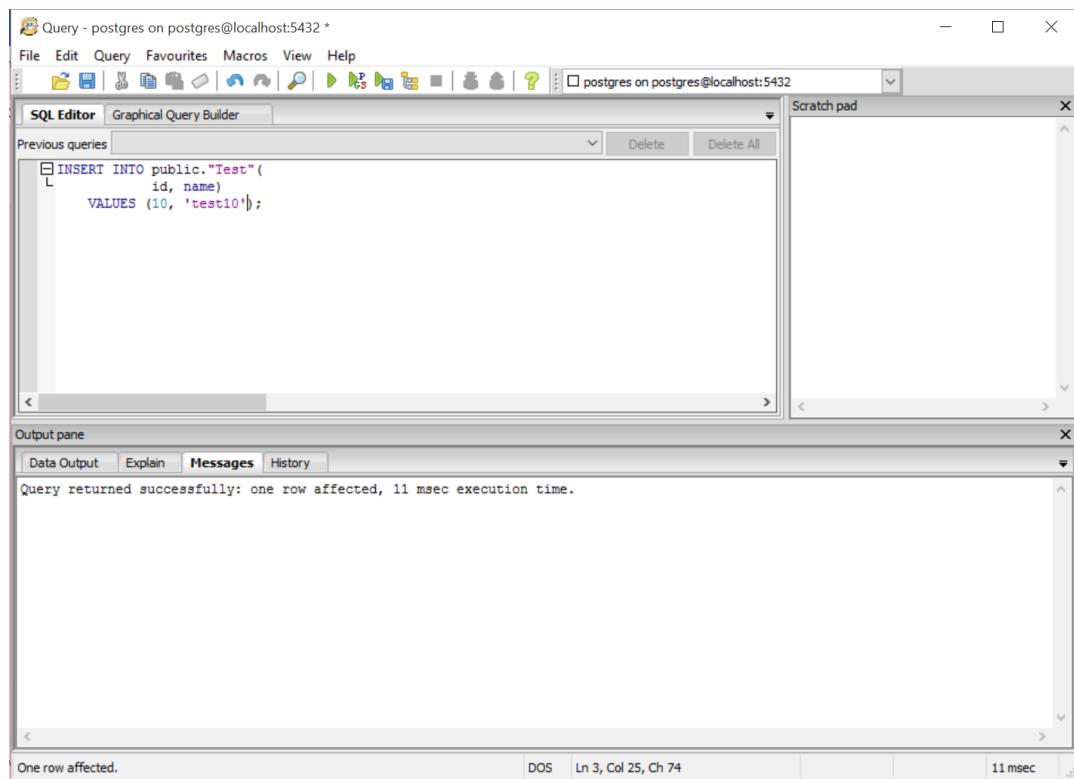


11. The SQL editor will open with the INSERT statement written ready for you to insert values



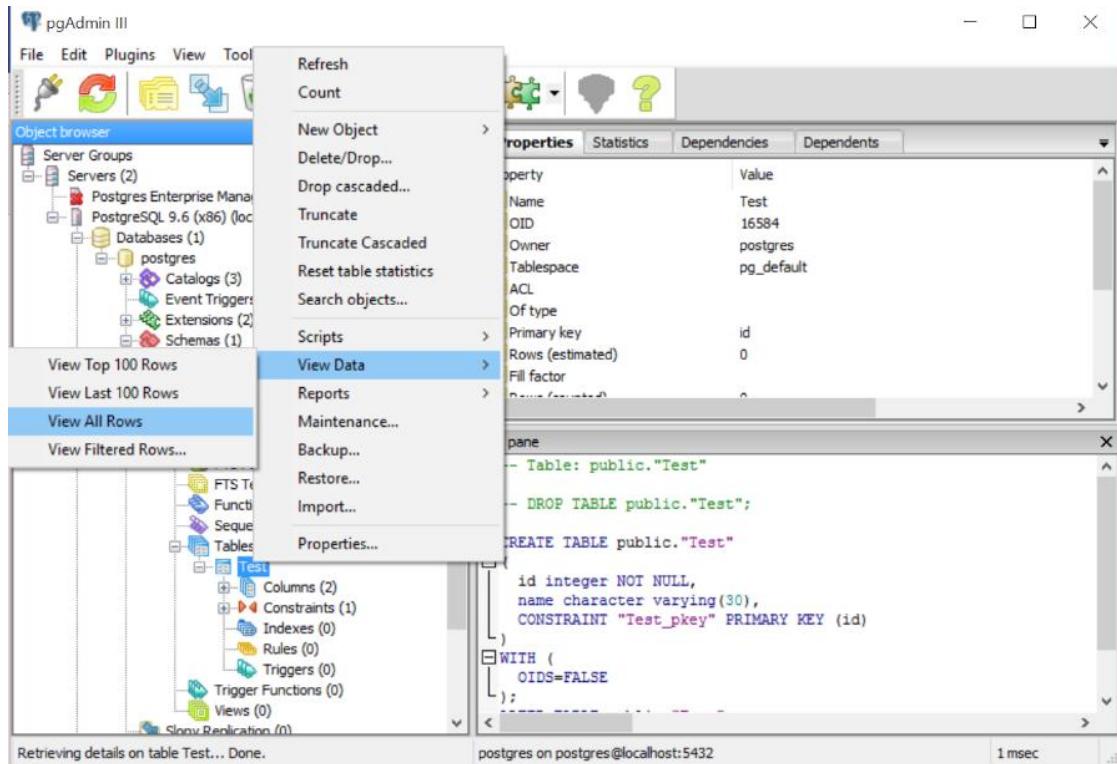
A screenshot of the SQL Editor interface. The title bar says "Query - postgres on postgres@localhost:5432". The menu bar includes File, Edit, Query, Favourites, Macros, View, and Help. The toolbar has various icons for file operations like Open, Save, Print, and Database management. The main window has tabs for "SQL Editor" and "Graphical Query Builder", with "SQL Editor" selected. A "Previous queries" dropdown shows the current query: "INSERT INTO public.\"Test\"(id, name) VALUES (?, ?);". To the right is a "Scratch pad" pane. Below the main editor is an "Output pane" with tabs for Data Output, Explain, Messages, and History. The status bar at the bottom shows "ready", "DOS", "Ln 1, Col 1, Ch 1", and a progress bar.

12. Insert the values as below and press this icon  to run the INSERT script

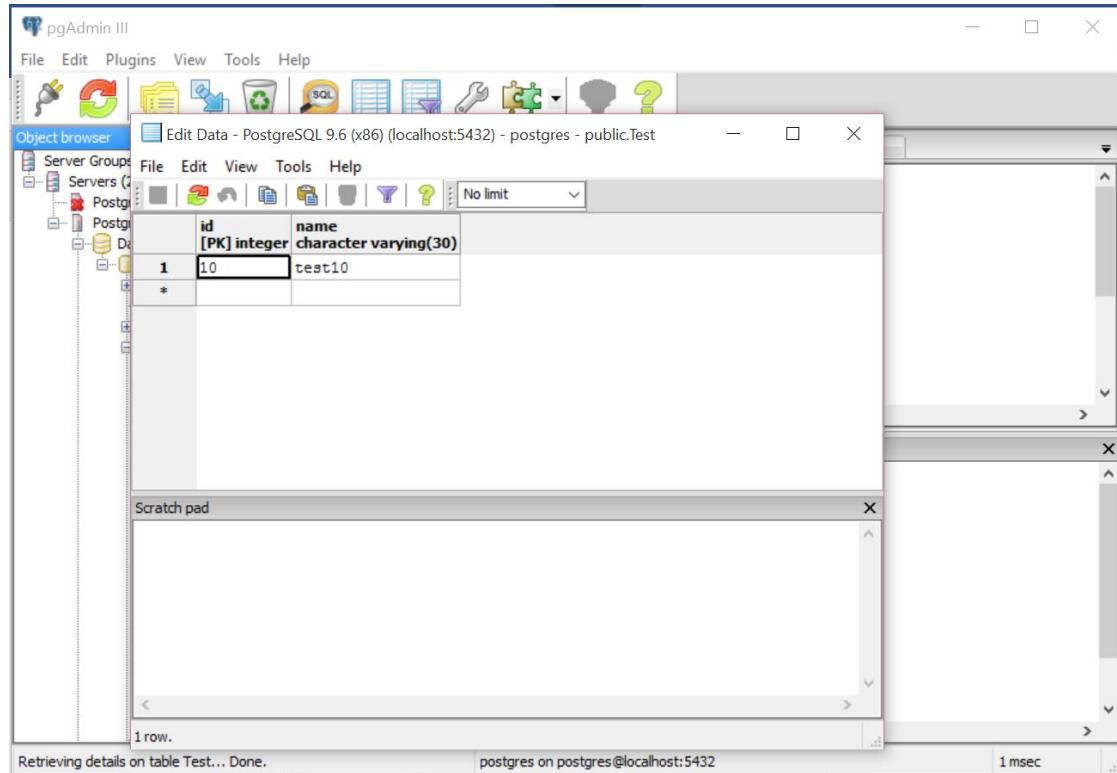


A screenshot of the SQL Editor interface, similar to the previous one but with a successful execution message. The title bar says "Query - postgres on postgres@localhost:5432 *". The main editor contains the same INSERT statement: "INSERT INTO public.\"Test\"(id, name) VALUES (10, 'test10');". The output pane shows the message "Query returned successfully: one row affected, 11 msec execution time." The status bar at the bottom shows "One row affected.", "DOS", "Ln 3, Col 25, Ch 74", and "11 msec".

13. View the record that you just inserted: right-click on “Test” table>>View Data>>View All Rows



14. The data view will look like below



15. Try to insert more values immediately in the table without script

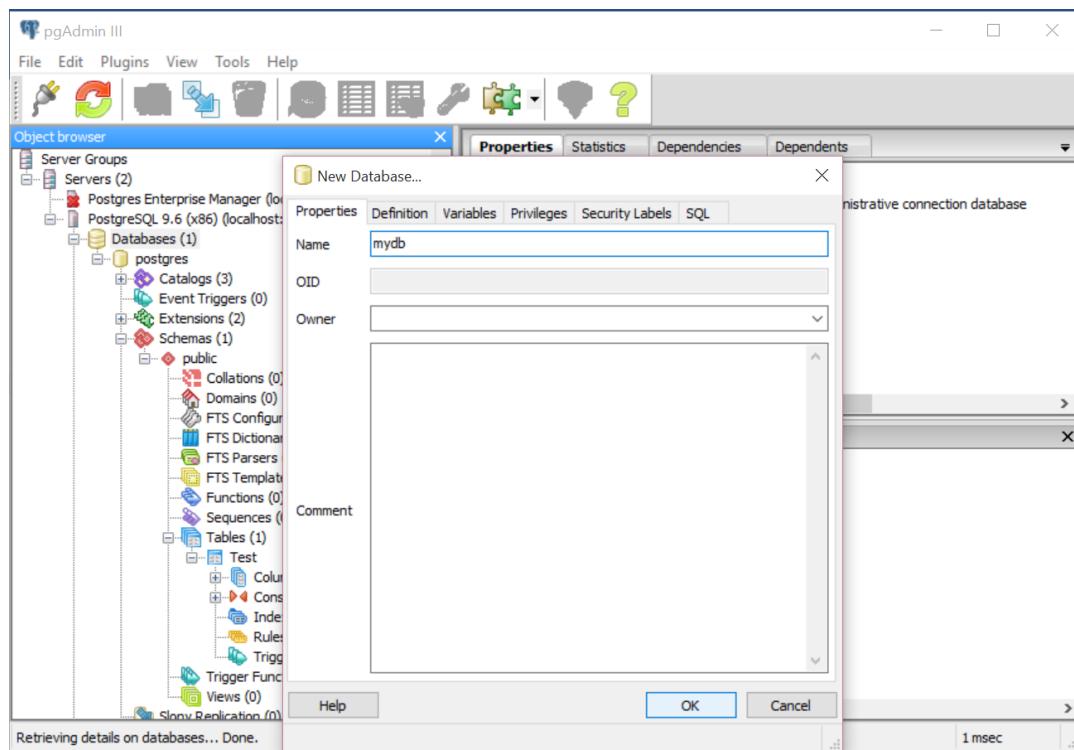
The screenshot shows the pgAdmin III interface. In the top left, the 'Object browser' pane displays a tree structure with 'Postgres' selected. A central window titled 'Edit Data - PostgreSQL 9.6 (x86) (localhost:5432) - postgres - public.Test' shows a table with two rows. The first row has 'id' as 10 and 'name' as 'test10'. The second row has 'id' as 11 and 'name' as 'test11'. The 'name' field for the second row is currently selected. Below the main window is a 'Scratch pad' pane. At the bottom, status bars show 'Retrieving details on table Test... Done.', 'postgres on postgres@localhost:5432', and '1 msec'.

Part 2: Creating a PostgreSQL database

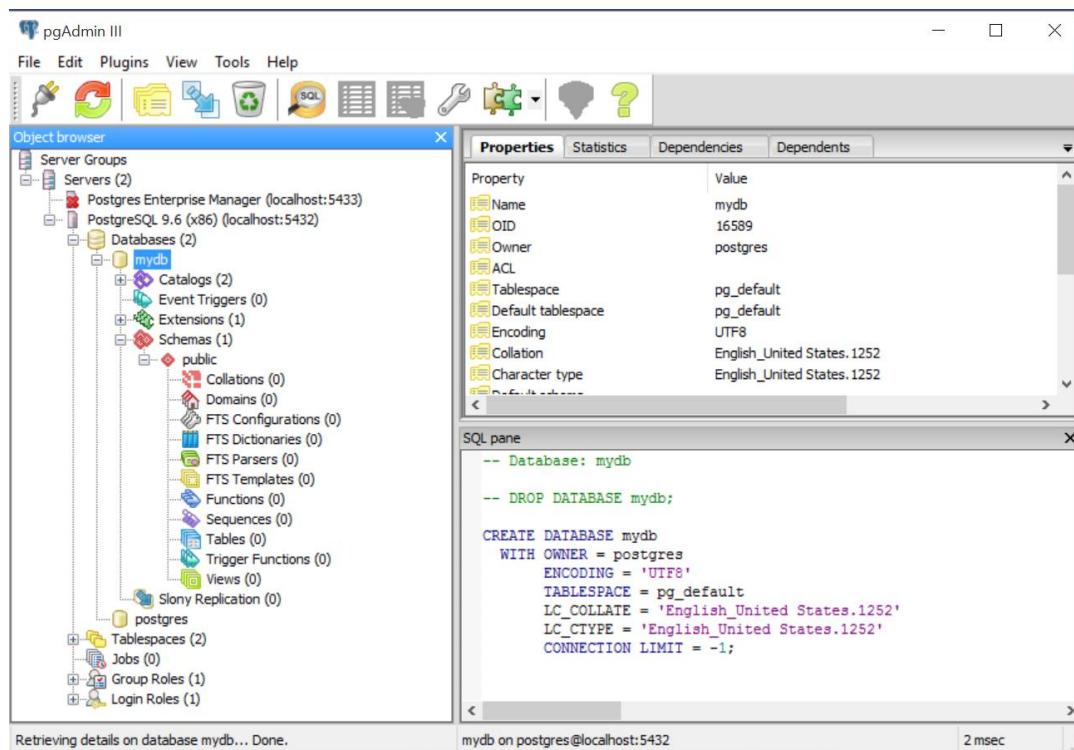
1. Select “Databases” node in the Object browser, then press

The screenshot shows the pgAdmin III interface. The 'Object browser' pane on the left shows 'Servers (2)' and 'Databases (1)'. The 'Properties' tab is selected in the top right, showing details for the 'postgres' database: Owner is 'postgres' and Comment is 'default administrative connection database'. Below the properties is a 'SQL pane' with an empty editor area. At the bottom, status bars show 'Retrieving details on databases... Done.', 'postgres on postgres@localhost:5432', and '1 msec'.

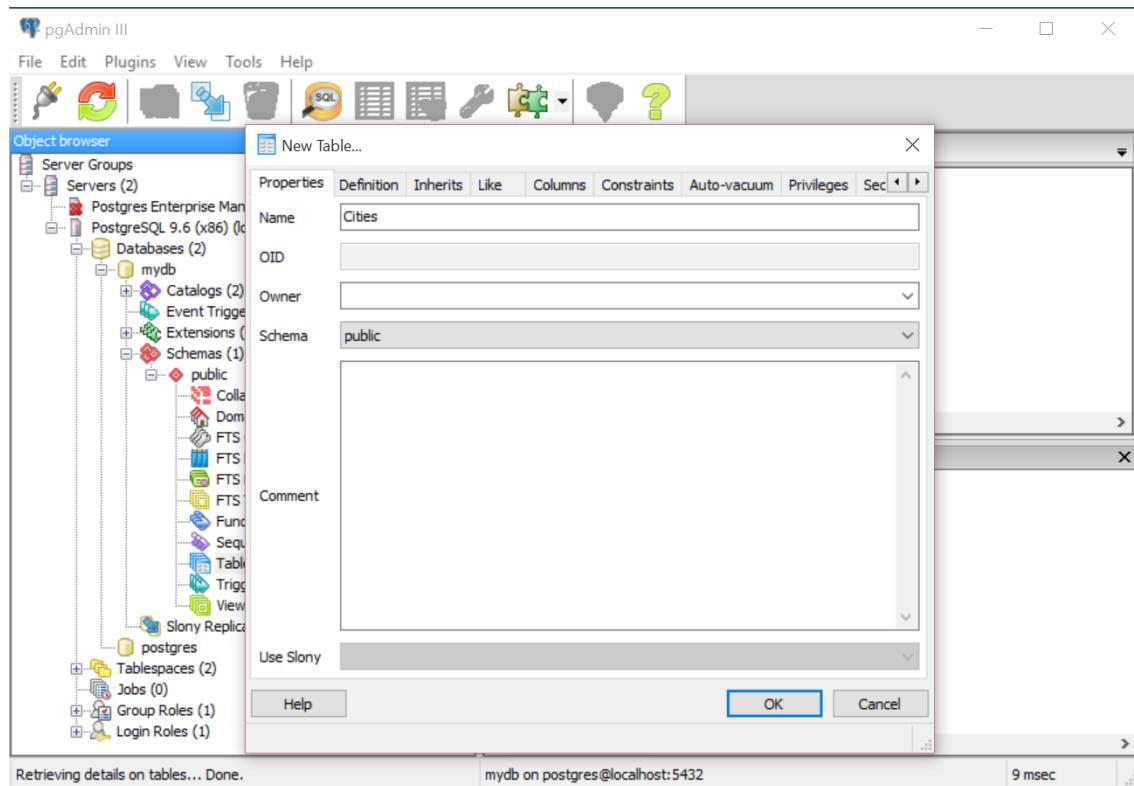
2. Name the database as “mydb” and press “OK”



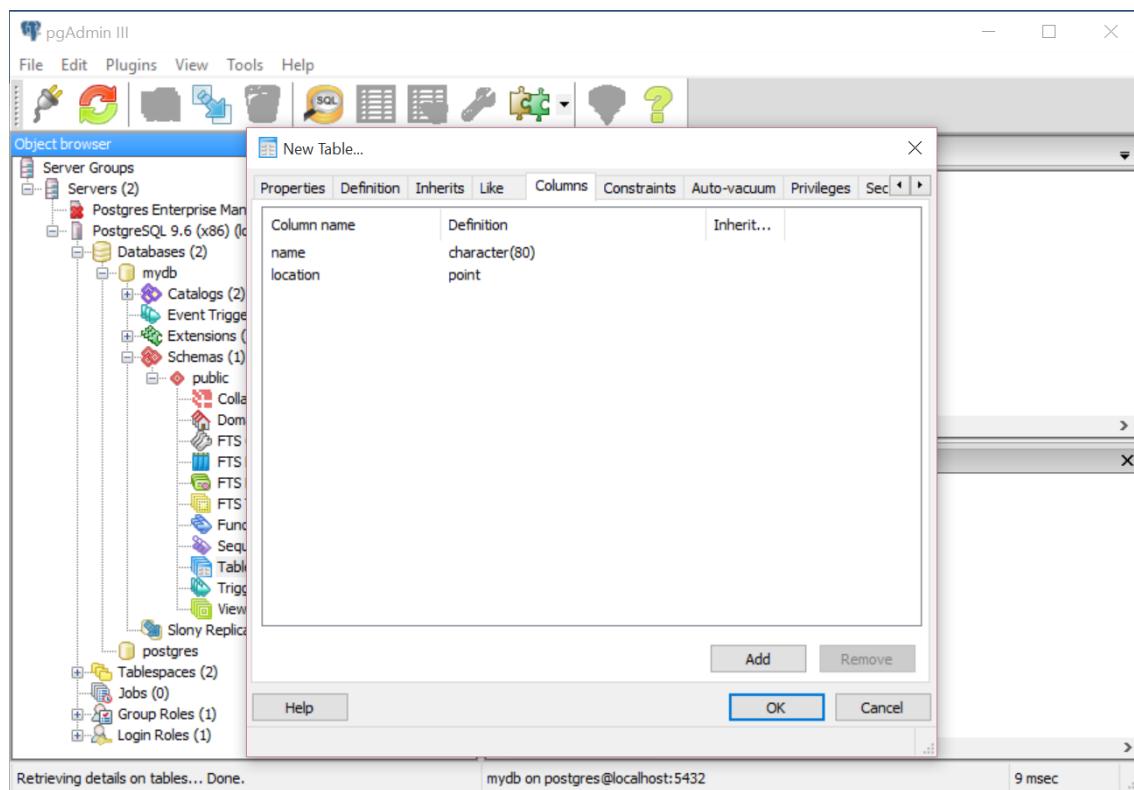
3. Now create tables to record weather data in various cities (refer to Part 1 for instructions to create a table)



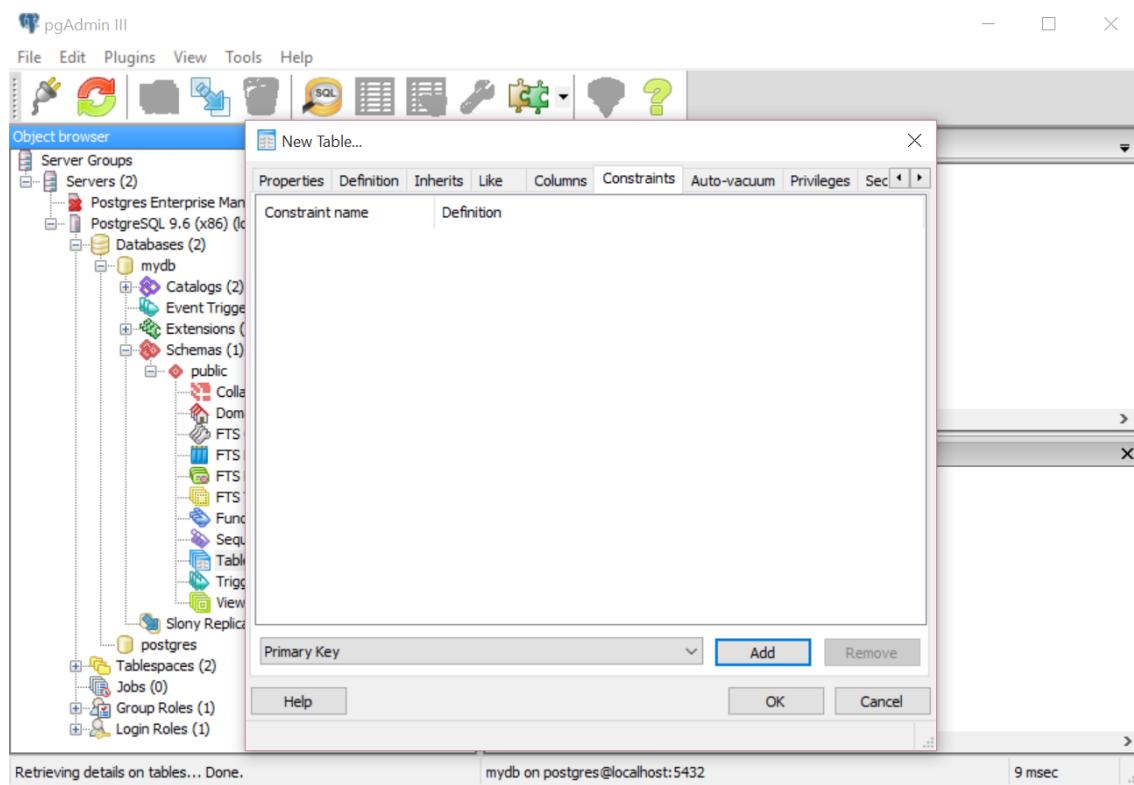
4. Create a table called “Cities”



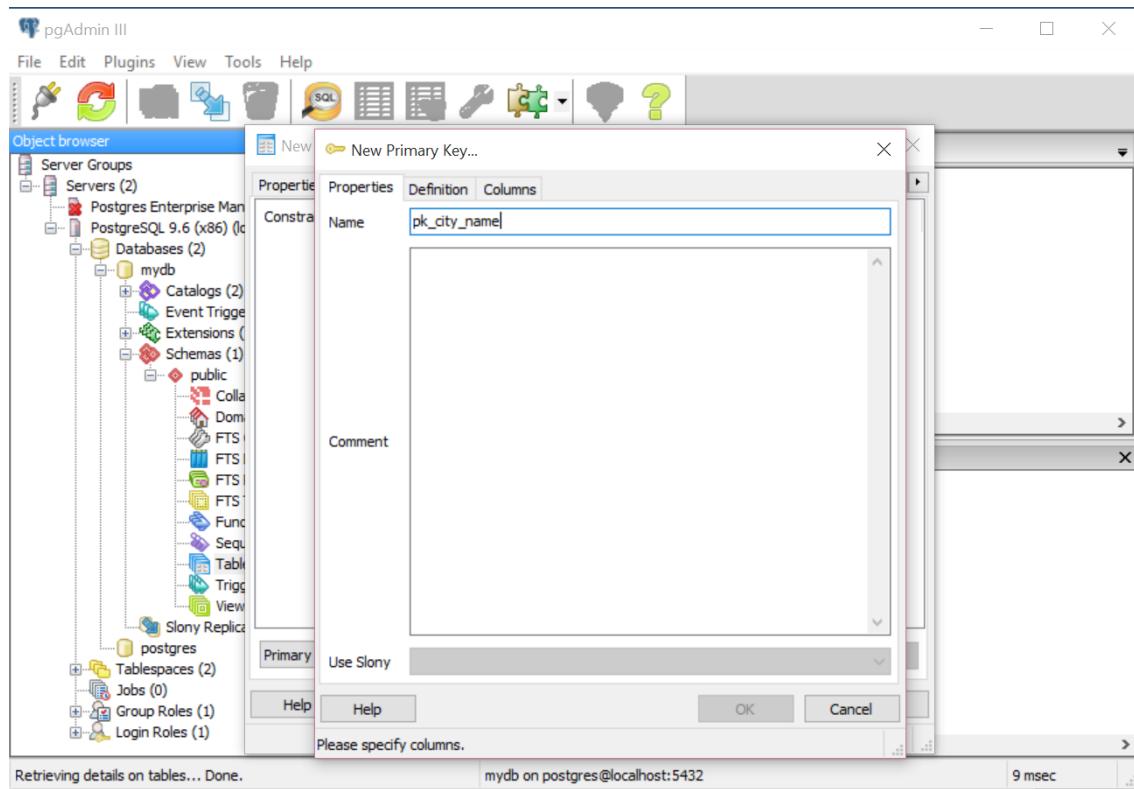
5. Add two columns to the table (name, location)



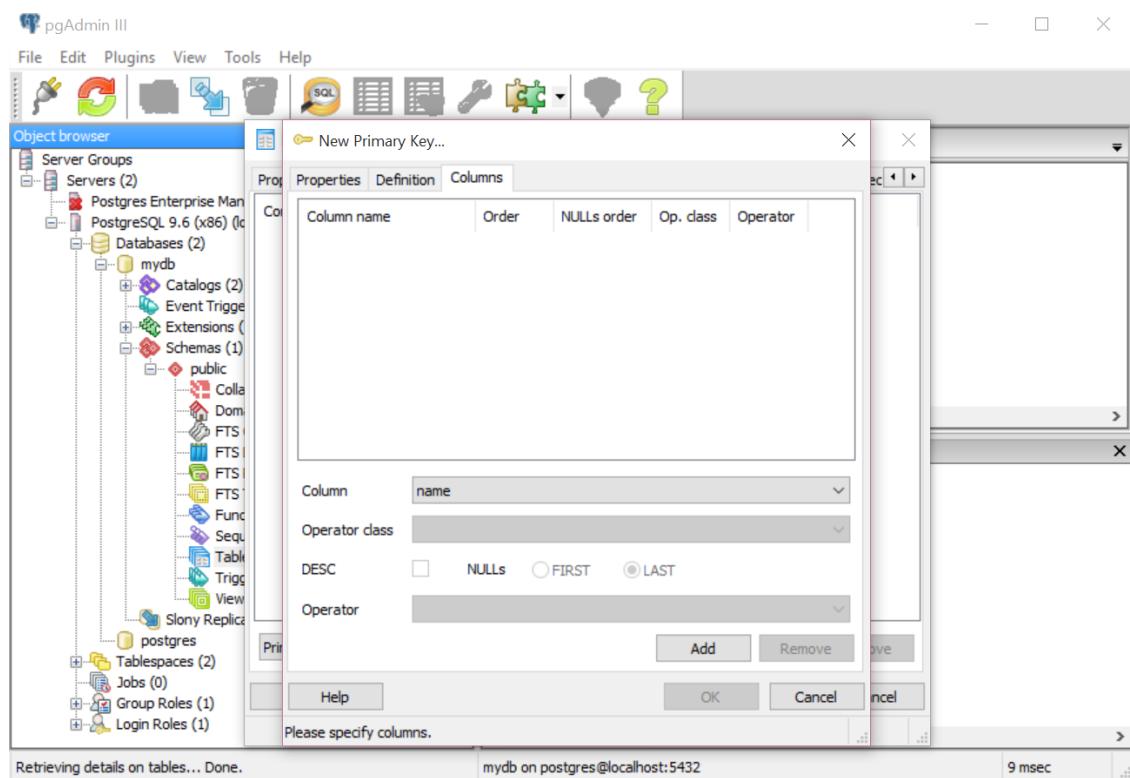
6. In the “New Table” window, go to “Constraints” Tab. Select Primary Key and press “Add”



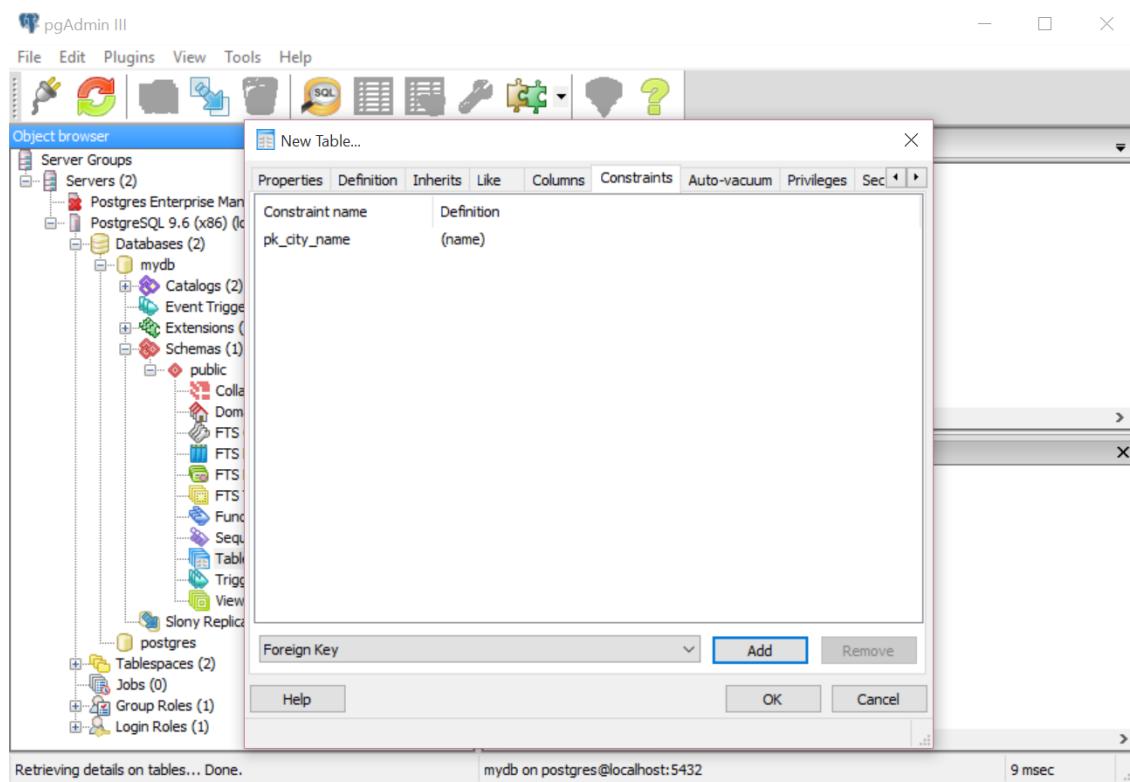
7. Make the city name as a primary key: in “Properties” Tab, type the following



8. Go to “Columns” Tab, select column “name”, and press “Add”, then press “OK”



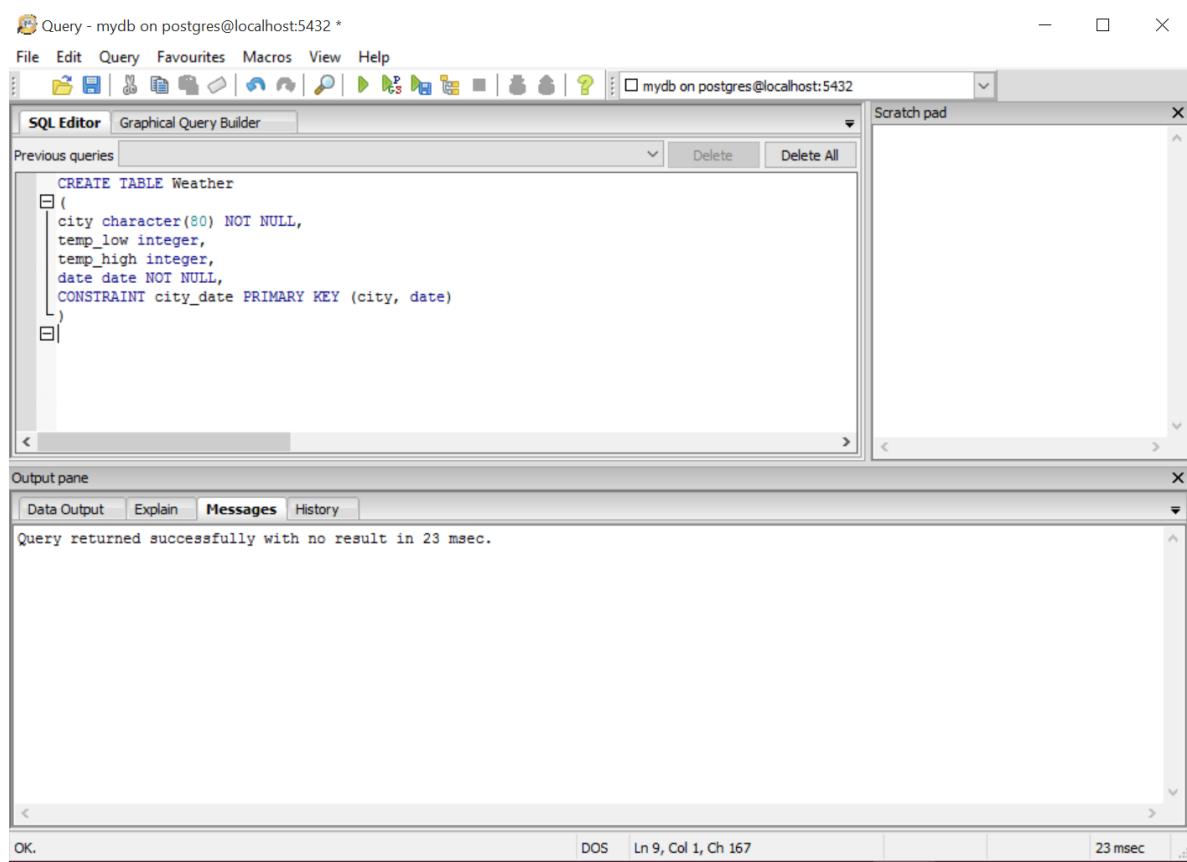
9. Now you created city name as a primary key, then press “OK”



10. Create another table called “Weather” using Query tool (Tools menu>>Query tool). Copy

and paste the following statement in the SQL editor and press :

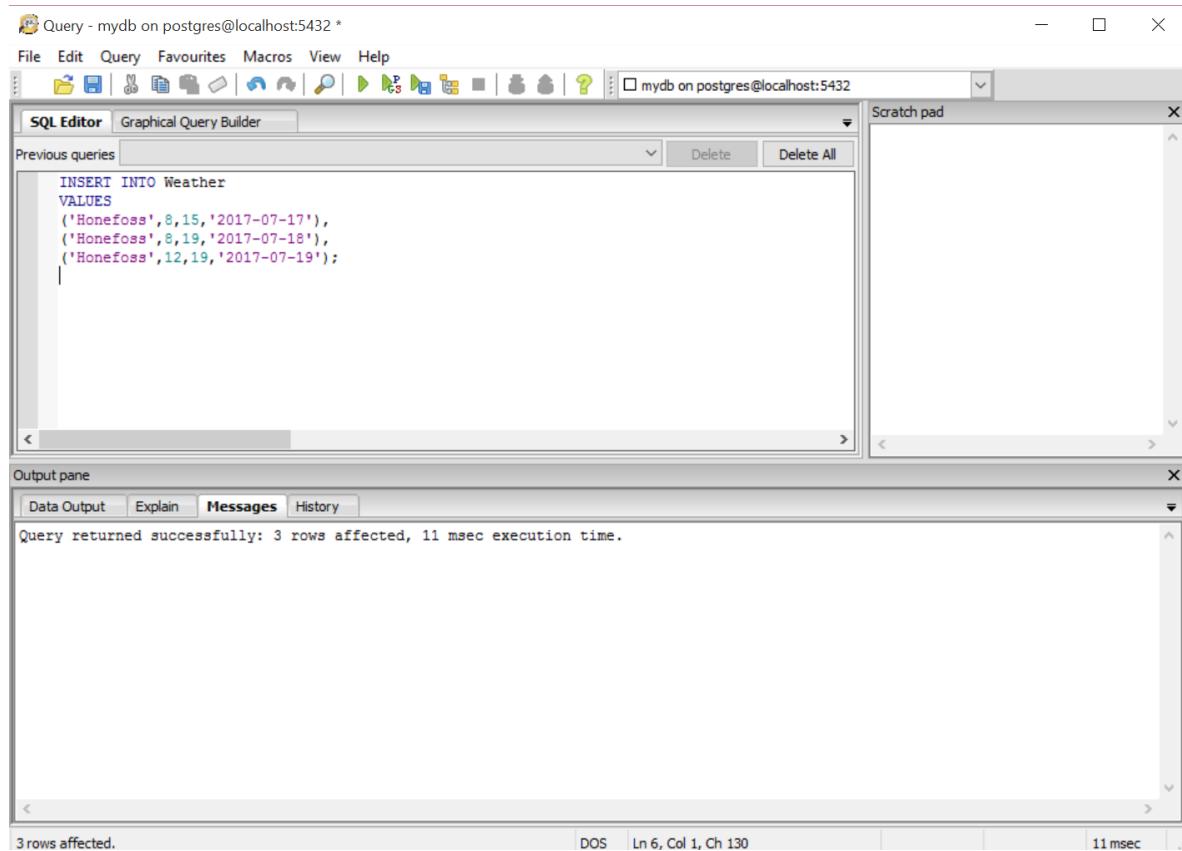
```
CREATE TABLE Weather
(
    city character(80) NOT NULL,
    temp_low integer,
    temp_high integer,
    date date NOT NULL,
    CONSTRAINT city_date PRIMARY KEY (city, date)
)
```



Right-click on the “Tables” node and click “Refresh” to see that “Weather” table that you just added.

11. Insert data into tables. Copy and paste the following statement in the SQL editor to fill in “Weather” table, then execute the INSERT INTO statement:

```
INSERT INTO Weather  
VALUES  
('Hønefoss',8,15,'2017-07-17'),  
('Hønefoss',8,19,'2017-07-18'),  
('Hønefoss',12,19,'2017-07-19');
```

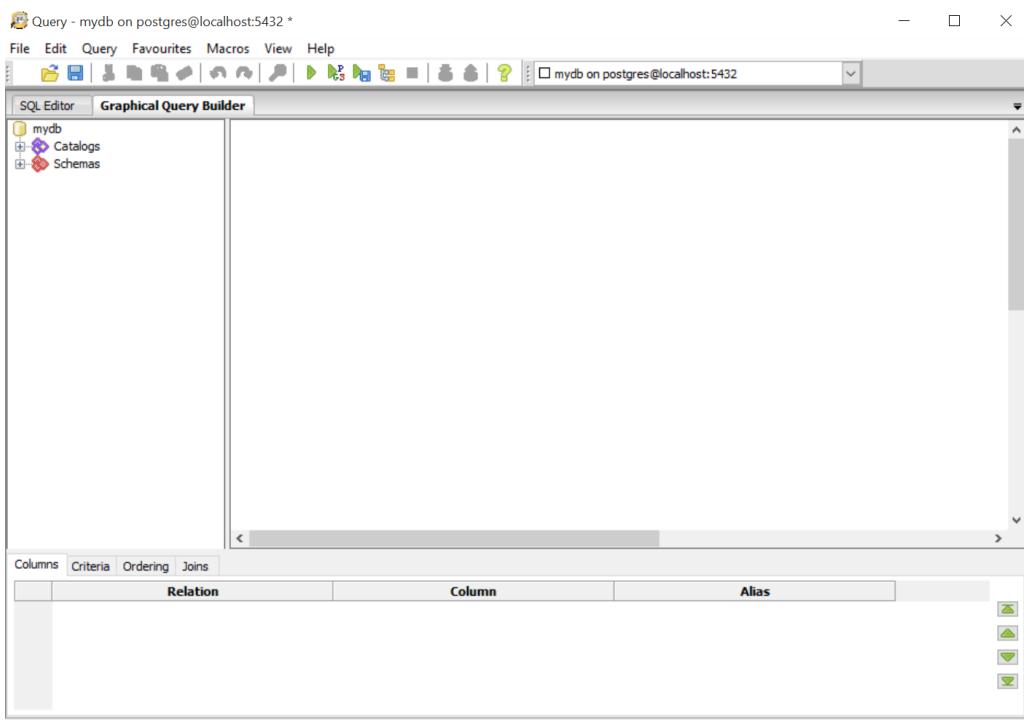


12. Insert the following two records into “Cities” table

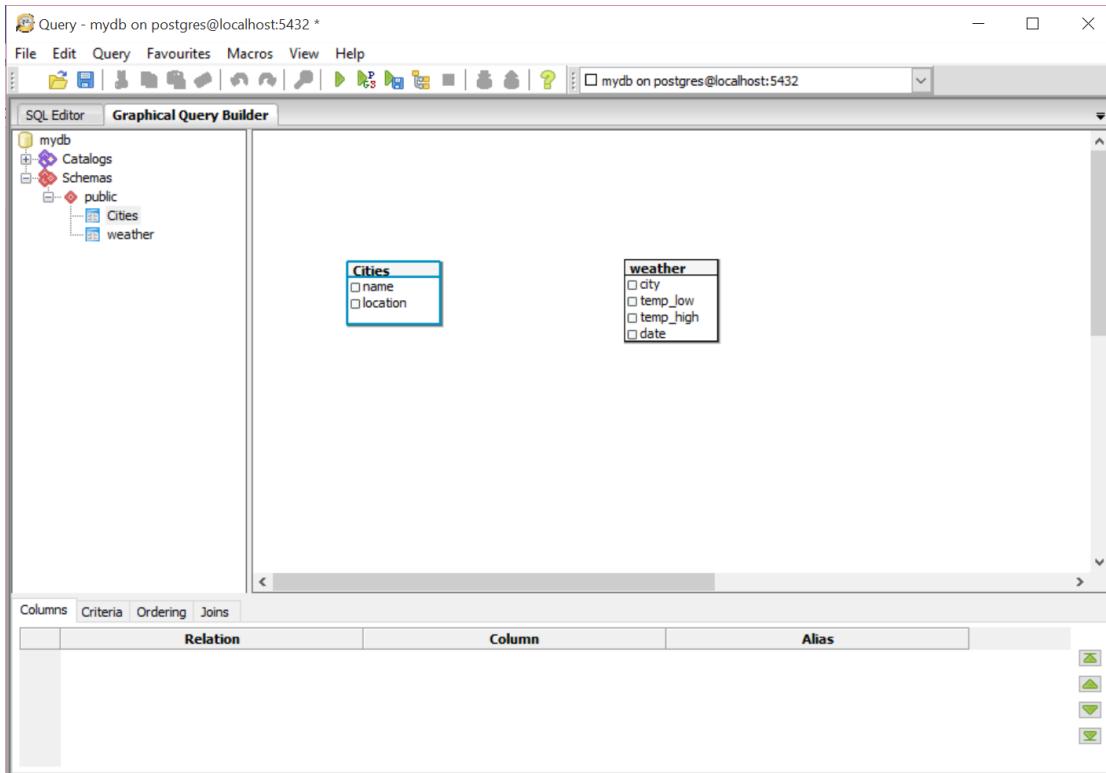
The screenshot shows the Edit Data tool for PostgreSQL 9.6 (x86) connected to the mydb database. The table being edited is public.Cities. The data is as follows:

	name [PK] character(80)	location point
1	Hønefoss	(60.166667,10.3)
2	Oslo	(59.95,10.75)
*		

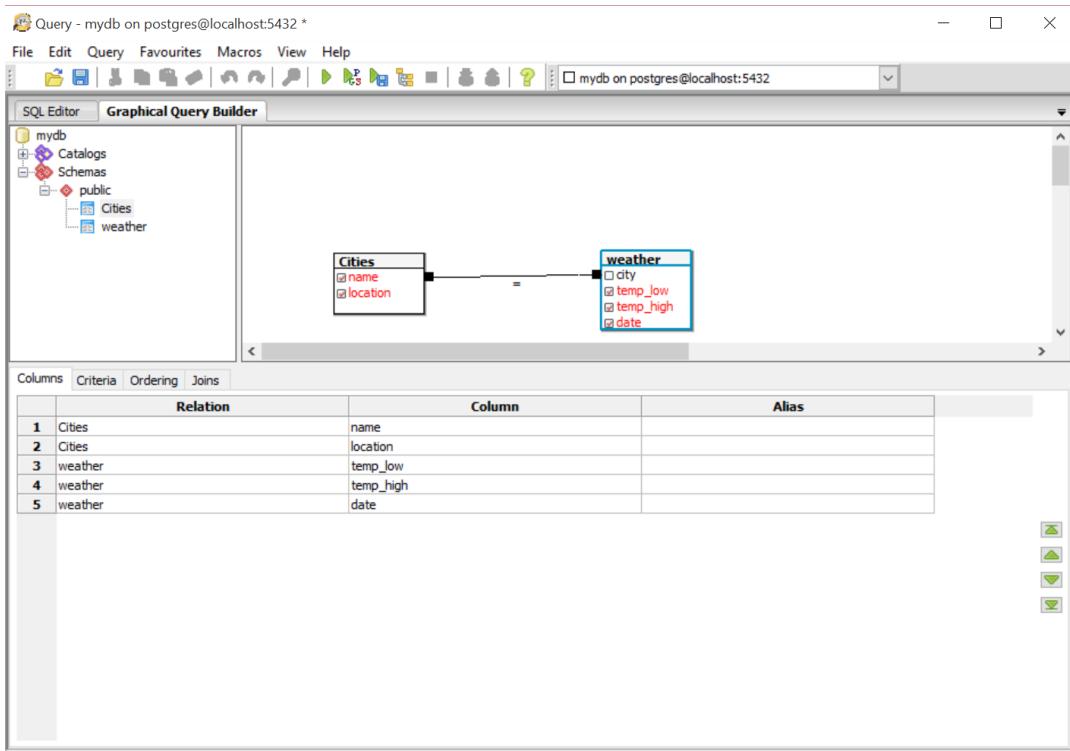
13. Now let's use the "Graphical Query Builder" to perform another query. While standing on "Tables" node, go to Tools>>Query tool and go to "Graphical Query Builder" Tab



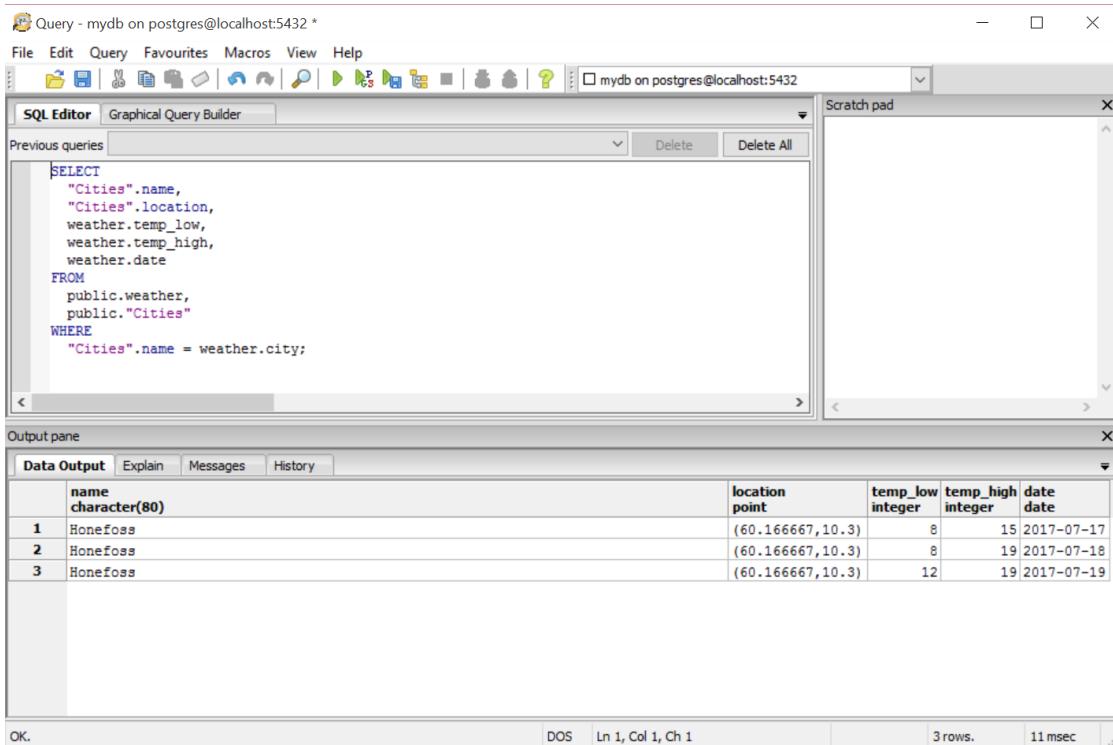
14. Expand the "Schemas" node, then drag and drop the two tables to the right-side area



15. Check the following columns from each table as shown below and make a join between the tables, then press :



16. The result would be as below:



The screenshot shows the SQL Editor with the following query entered in the SQL pane:

```

SELECT
    "Cities".name,
    "Cities".location,
    weather.temp_low,
    weather.temp_high,
    weather.date
FROM
    public.weather,
    public."Cities"
WHERE
    "Cities".name = weather.city;
  
```

The Output pane shows the results of the query:

	name character(80)	location point	temp_low integer	temp_high integer	date
1	Honefoss	(60.166667,10.3)	8	15	2017-07-17
2	Honefoss	(60.166667,10.3)	8	19	2017-07-18
3	Honefoss	(60.166667,10.3)	12	19	2017-07-19