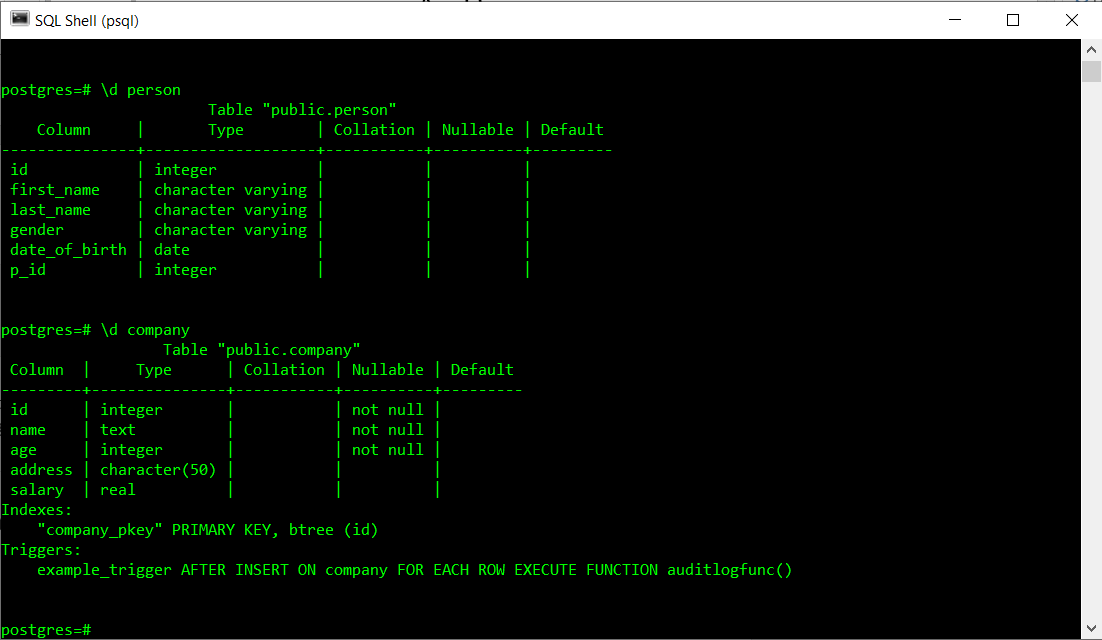
Kloudone Assignment 23-08-2020

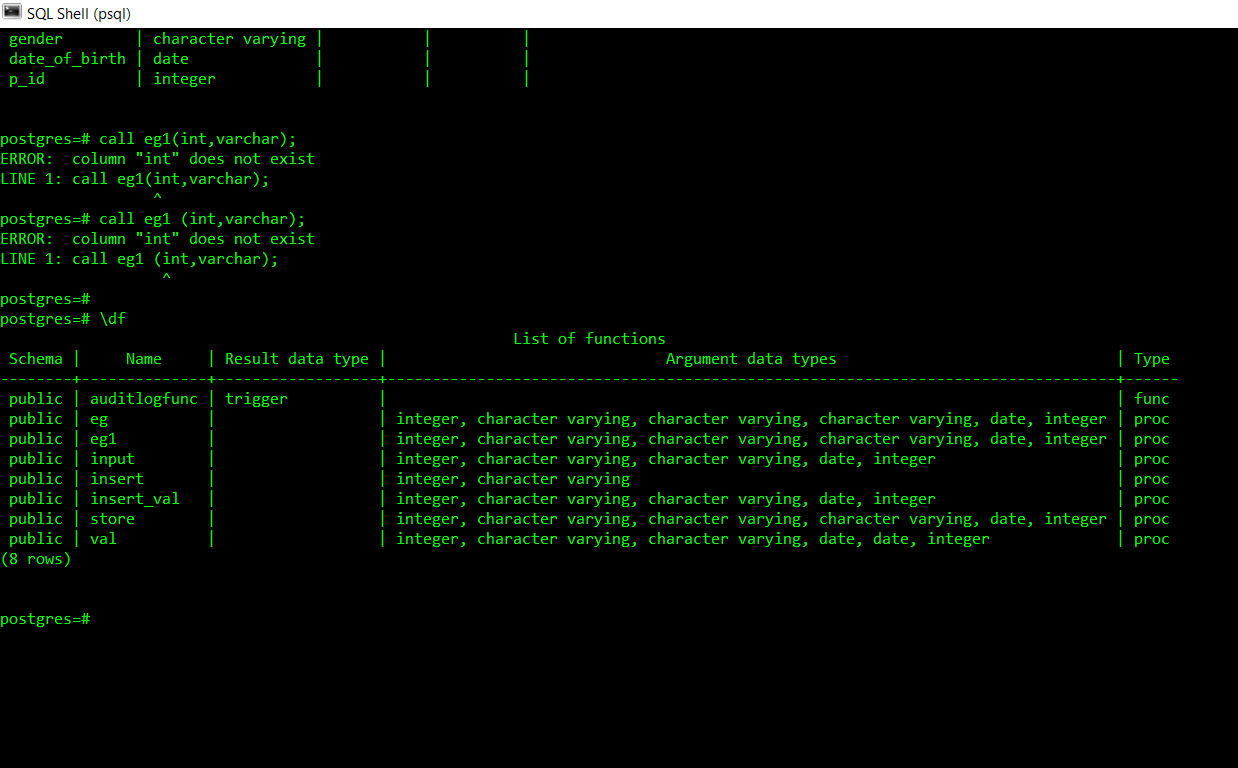
The differences between the stored procedure and a function with an example:

| **Sr. No.** | **Key** | Function | **Procedure** |
| --- | --- | --- | --- |
| 1 | Definition | A function is used to calculate result using given inputs. | A procedure is used to perform certain task in order. |
| 2 | Call | A function can be called by a procedure. | A procedure cannot be called by a function. |
| 3 | DML | DML statments cannot be executed within a function. | DML statements can be executed within a procedure. |
| 4 | SQL, Query | A function can be called within a query. | A procedure cannot be called within a query. |
| 5 | SQL, Call | Whenever a function is called, it is first compiled before being called. | A procedure is compiled once and can be called multiple times without being compiled. |
| 6 | SQL, Return | A function returns a value and control to calling function or code. | A procedure returns the control but not any value to calling function or code. |
| 7 | try-catch | A function has no support for try-catch | A procedure has support for try-catch blocks. |
| 8 | SELECT | A select statement can have a function call. | A select statemnt can't have a procedure call. |
| 9 | Explicit Transaction Handling | A function can not have explicit transaction handling. | A procedure can use explicit transaction handling. |

A table created with the constraint and example for trigger:



Functions in SQL:



SQL functions are simply sub-programs, which are commonly used and re-used throughout SQL database applications for processing or manipulating data. All SQL database systems have DDL (data definition language) and DML (data manipulation language) tools to support the creation and maintenance of databases.

SYNTAX for function:

CREATE [OR REPLACE] FUNCTION function\_name (arguments)

RETURNS return\_datatype AS $variable\_name$

DECLARE

declaration;

[...]

BEGIN

< function\_body >

[...]

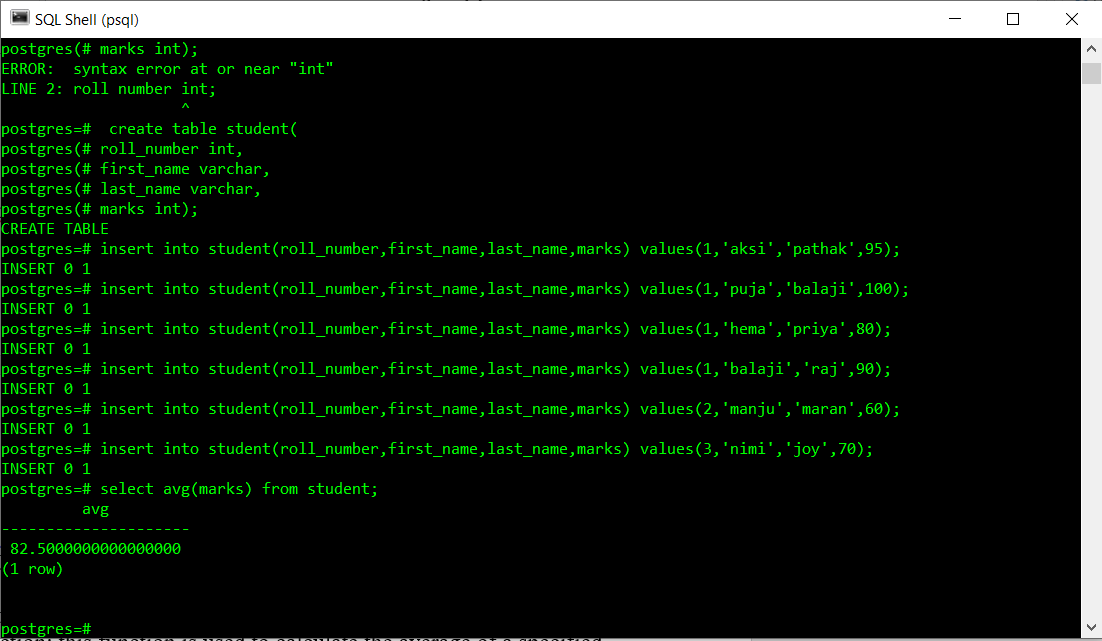
RETURN { variable\_name | value }

END; LANGUAGE plpgsql;

The group functions:

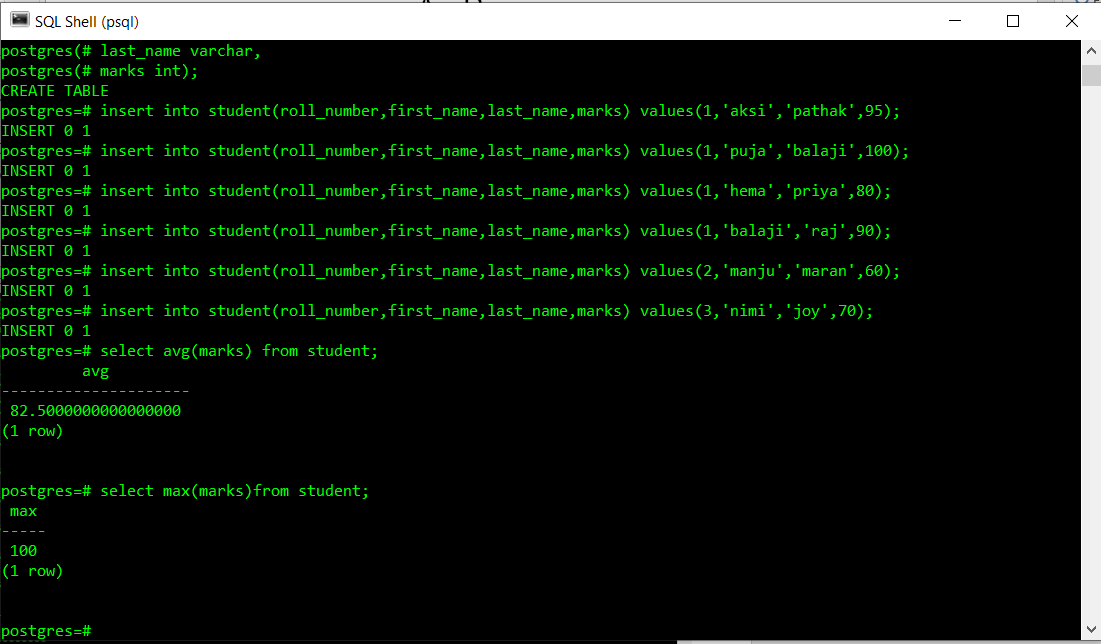
1. AVG function: this function is used to calculate the average of a specified coloumn.

SELECT AVG<attribute name> FROM<table name>;



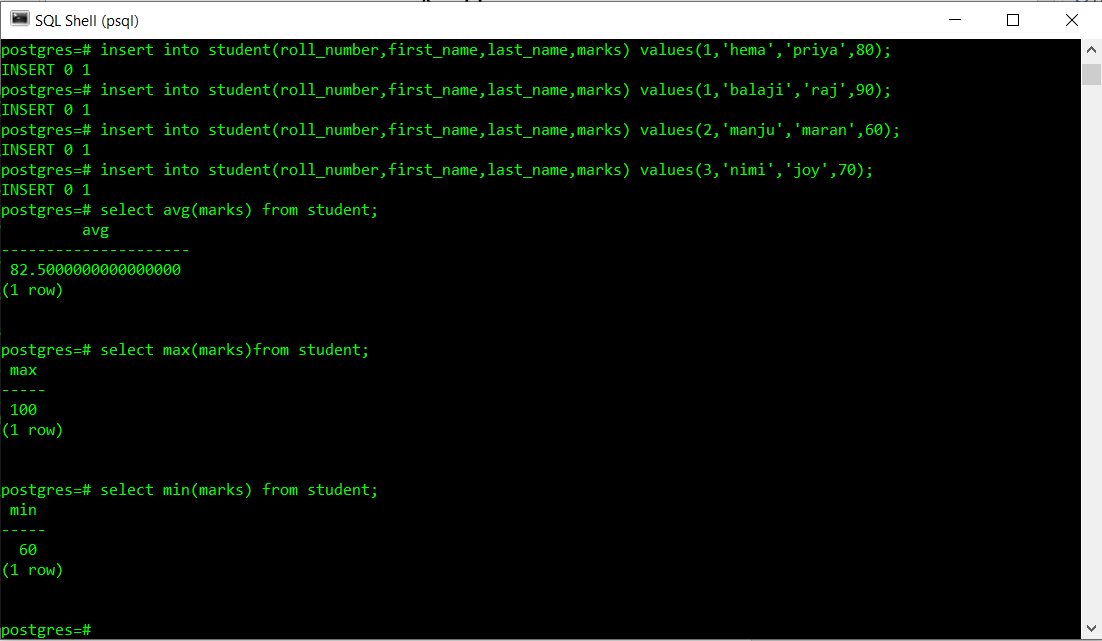
1. MAX function: This function is used to find the maximum value of the column.

SELECT MAX<attribute name> from<table name>;



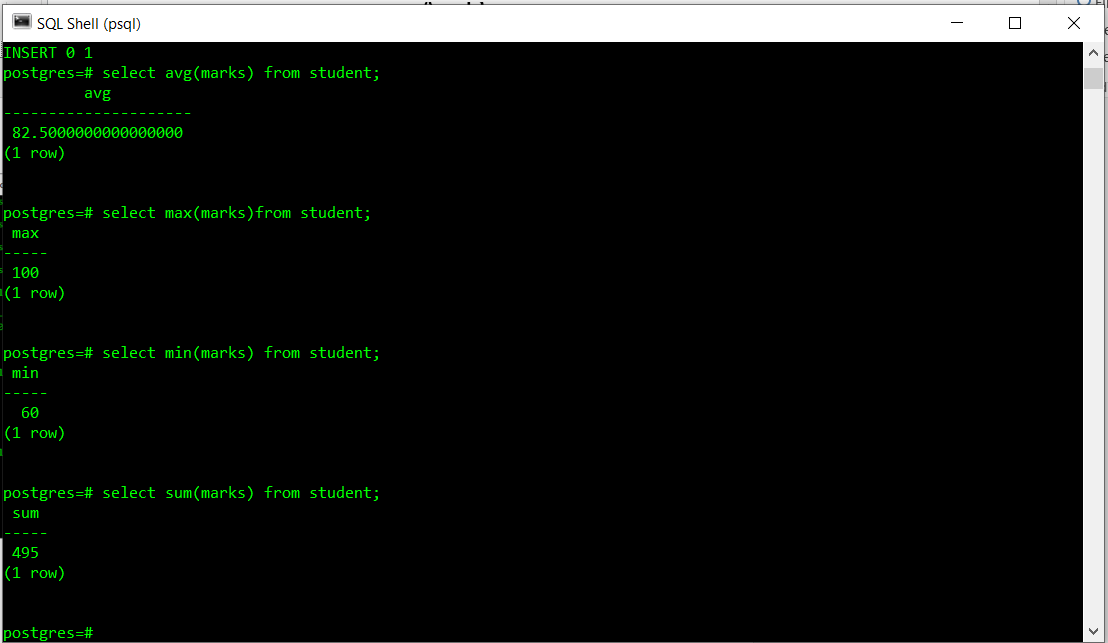
1. MIN function: This function is used to find the minimum value of the column.

SELECT MIN<attribute name> FROM<table name>;



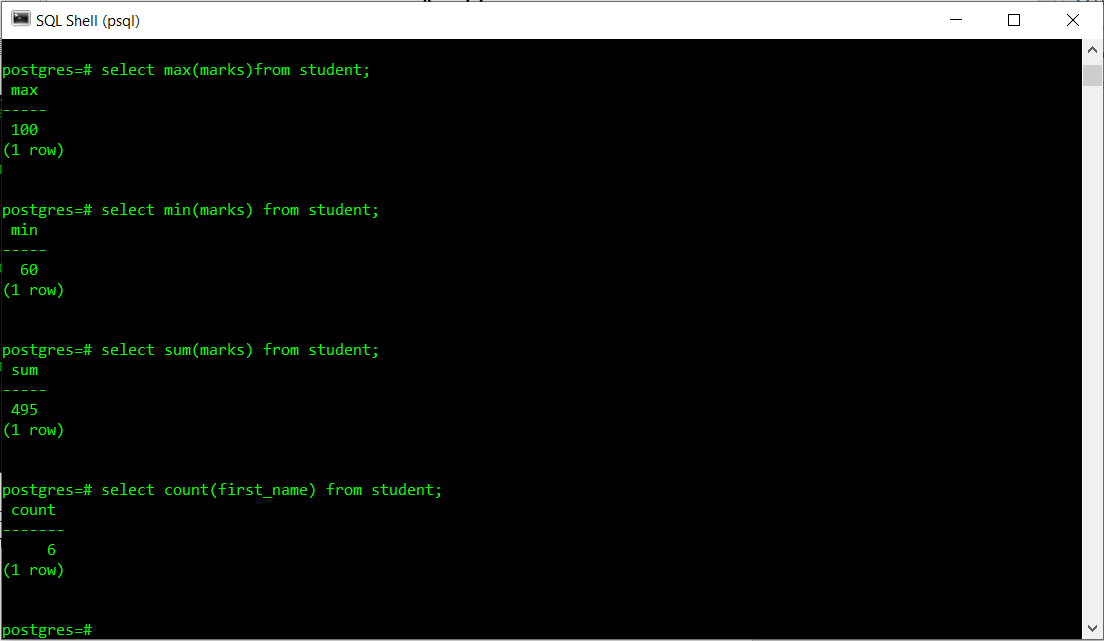
1. SUM function: This function is used to calculate SUM of the column.

SELECT SUM<attribute name> FROM<table name>;



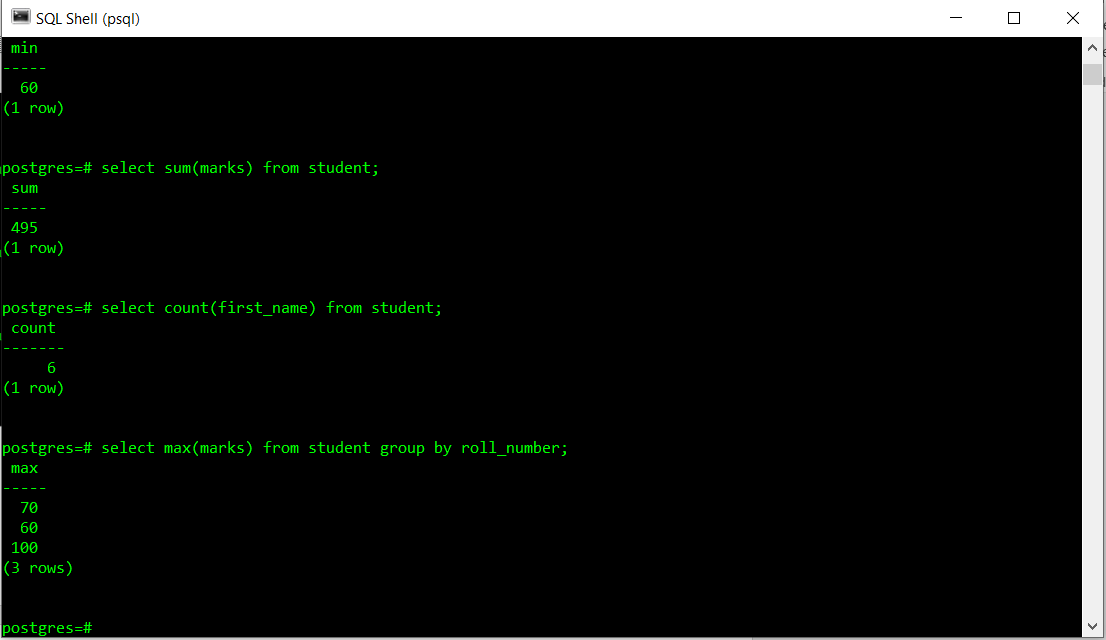
1. COUNT function: This function is used to count the number of values in a given column.

SELECT COUNT<attribute name> from<table name>;



1. GROUP BY clause: The GROUP BY clause groups the rows in the resulting table by columns that have the same values, so that each group is reduced to a single row.

SELECT MAX<attribute name> FROM<table name> GROUP BY <attribute name>;

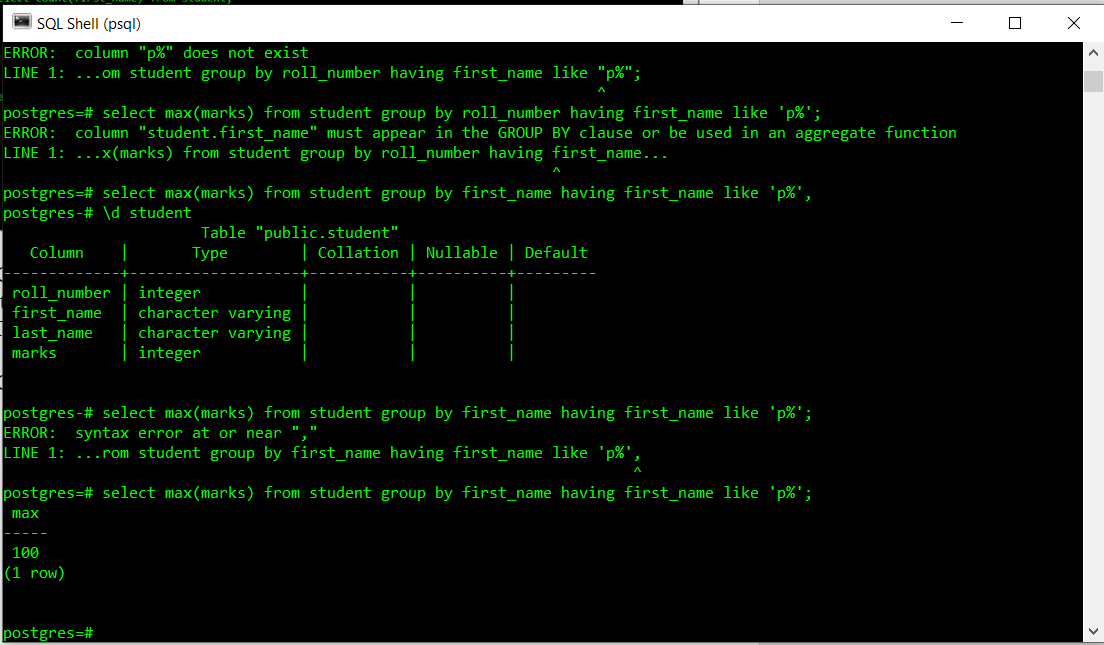


1. HAVING clause: The HAVING clause restricts grouped rows that appear in the resulting table.

SELECT MAX<attribute name> FROM<table name>

GROUP BY <attribute name>

HAVING<attribute name> LIKE “S%”;



A created function :

Script

CREATE OR REPLACE FUNCTION total\_person()

RETURNS integer as $total$

DECLARE

total integer;

BEGIN

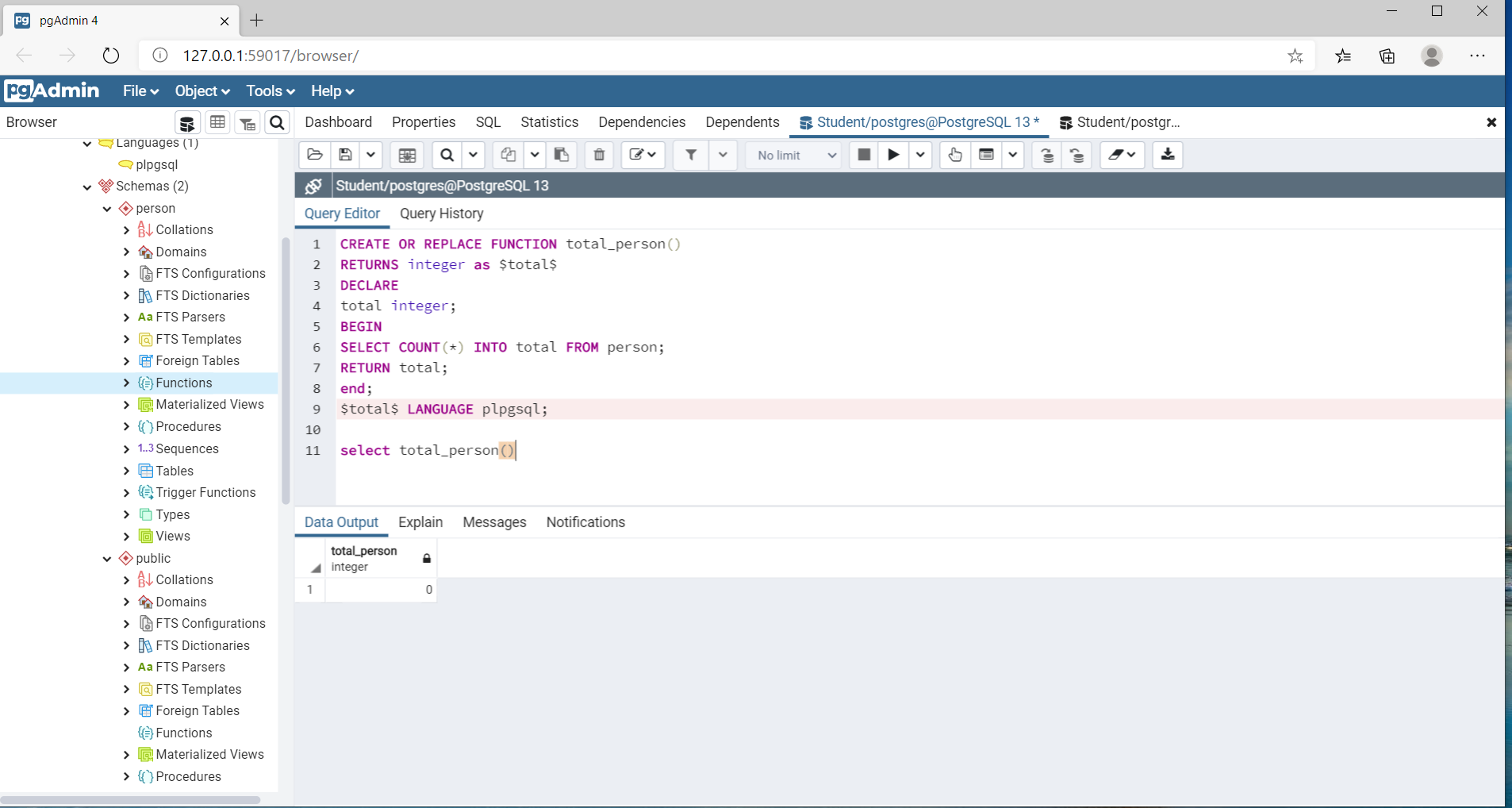
SELECT COUNT(\*) INTO total FROM person;

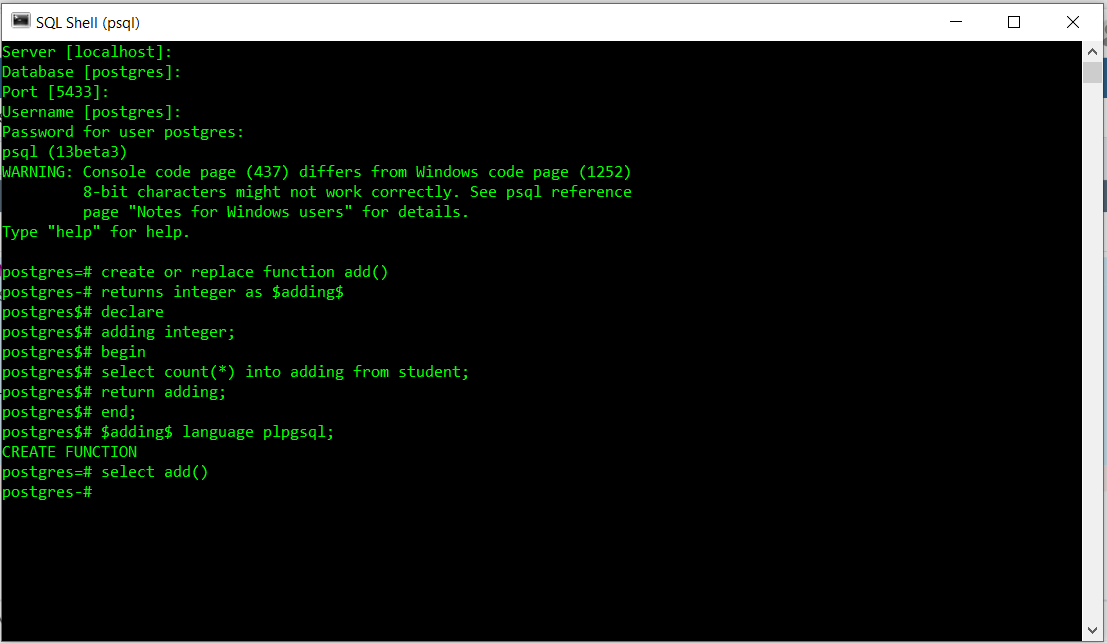
RETURN total;

end;

$total$ LANGUAGE plpgsql;

select total\_person()





Nested Functions

