CIS 3530 Assignment 3

Objectives: built-in SQL functions, procedural extensions to SQL in postgres, independent learning

Submission Instruction: There are a total of 6 files that you need to create for this assignment (1 txt file for question 1, and 5 sql files for question 2). You must submit a zip or a tar file that contains these 6 files – name the zip file as lastnameFirstnameA3.zip or lastnameFirstnameA3.tar

Question 1: (10 marks) Review and list five SQL built-in functions, their description in your own words and a SELECT query that uses the function. The query **must use** one or more relations in the movies database from lab3 (i.e., relations movies, rating, reviewer).

An example of built-in function POWER is shown in the table below. The given example uses a numeric function called POWER that takes integers as input and returns an integer.

DO NOT INCLUDE

- **any other function that is numeric.**
- **any of the aggregate functions (i.e., SUM, AVG, MIN, MAX, COUNT)**

You may use functions that are string functions, date functions, date conversion functions and similar.

You may find the following tutorial helpful:
https://www.postgresql.org/docs/9.2/functions.html

Fill in the table with your answers and submit in a file named lastnameFirstnameA3Q1.txt. Note: We will compile and run the SQL statements that you submit on the movies database that was created for lab3.

	Function name	**description**	**SELECT query in Postgres**

0	**POWER**	**Takes 2 integer inputs m and n**
		Returns an integer, which is
		m raised to the power of n (m^n)
			SELECT POWER (2, stars)
			FROM RATING;

1			
2			
3			
4			
5			

Question 2: (5 x 6 = 30 marks) This question has 5 parts (a,b, .. e). For each part of this question, you must use one or more relations in the movies database from lab3 (i.e., relations movies, rating, reviewer).

To answer this question, you may find the following tutorials helpful:

1. tutorial.pdf posted on Courselink under Contents -> A3
2. https://www.postgresql.org/docs/11/sql-createprocedure.html
3. https://www.postgresql.org/docs/14/plpgsql-control-structures.html#PLPGSQL-ERROR-TRAPPING
4. https://www.postgresql.org/docs/current/plpgsql-cursors.html
5. https://www.postgresql.org/docs/current/sql-createtrigger.html

a. Write a PL/SQL code to declare variables of anchored type belonging to type of at least one or two columns of your **movies** database. Your code should also include appropriate executable section instructions to print some values.
Name your file as lastnameFirstnameA3Q2a.sql

b. Write a PL/SQL code that declares a cursor and utilizes this cursor in a loop instruction to implement appropriate action.
Name your file as lastnameFirstnameA3Q2b.sql

c. Write a PL/SQL code that includes an EXCEPTION section and handles the exception (you may use pre-defined exceptions such as unique_violation).
Name your file as lastnameFirstnameA3Q2c.sql.

d. Write a PL/SQL function to perform appropriate action. Also write the anonymous block that calls this procedure.
Name your file as lastnameFirstnameA3Q2d.sql.

e. Write a PL/SQL code to create a trigger to perform appropriate action as well as the code to show the effect of this trigger.
Name your file as lastnameFirstnameA3Q2e.sql.