CS 166: Lab Assignment: Triggers and Stored Procedures

The purpose of the following assignment is to explore how triggers and stored procedures can be used to implement custom domain logic and response to specified changes in the database. You are encouraged to learn it yourself. Here are several helpful links:

1. PostgresSQL docs on triggers
2. Triggers and Stored Procedures
3. PostgresSQL docs on sequences

For this assignment you are given the scripts used in Lab 7 with which you should initialize your database. Rember to execute "source ./startPostgreSQL" and "source ./createPostgreDB.sh".

Your task is to implement a trigger and procedure to automatically populate part_number with incremented value upon insertion of the new row into part_nyc. After that your insert statements should not include value for part_number and will look like this:

```
Insert into part_nyc(supplier,color,on_hand,descr) Values (0,0,20,'Desc');
```

Implement your trigger&procedure in triggers.sql. You can test your code using *test.sh*

1. First create a sequence using the following SQL:

   ```
   CREATE SEQUENCE part_number_seq  START WITH 50000;
   ```

2. Create a procedure that will return next value of the aforementioned sequence. Use function nextval('part_number_seq') to get the next value from the sequence.

   Use the following syntax to create your procedure:

   ```
   CREATE LANGUAGE plpgsql;
   CREATE OR REPLACE FUNCTION func_name
    RETURNS "trigger" AS
    $BODY$
    BEGIN
       ...
    END;
    $BODY$
    LANGUAGE plpgsql VOLATILE;
   ```

3. Use the following syntax to create a trigger calling the procedure upon insertion of the new record:

   ```
   CREATE TRIGGER name { BEFORE | AFTER } { INSERT | UPDATE | DELETE }
   ON table FOR EACH { ROW | STATEMENT }
   EXECUTE PROCEDURE funcname ( arguments )
   ```