CS166 Section 22 Alejandro Sherman

Lab 7 Report Images:

First I logged into my designated lab machine - wch 133-38.
Next I copied over the new and needed files from the lab 7 zip folder to the /tmp folder.
Then I ran source ./startPostgreSQL.sh inside the /tmp folder
Next I ran source ./createPostgreDB.sh

Next I prepared the database with the given data from lab 7 with:
cp part_nyc.dat /tmp/$USER/myDB/data/
cp part_sfo.dat /tmp/$USER/myDB/data/
psql -h localhost -p $PGPORT $USER"_DB" < create_tables.sql (Ran twice)

```
/tmp
asher011@wch133-38 $ cp part_nyc.dat /tmp/$USER/myDB/data/
/tmp
asher011@wch133-38 $ cp part_sfo.dat /tmp/$USER/myDB/data/
/tmp
asher011@wch133-38 $ psql -h localhost -p $PGPORT $USER"_DB" < create_tables.sql
DROP TABLE
CREATE TABLE
COPY 49995
DROP TABLE
CREATE TABLE
COPY 39996
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
DROP TABLE
CREATE TABLE
INSERT 0 1
INSERT 0 1
/tmp
asher011@wch133-38 $
```

Next I test my triggers.sql file with
source ./test.sh
I ran this command a few times to show the sequence increasing each time (ignoring each error due to redefinitions)

```
alejandro@Alejandros-Laptop: ~
CREATE TABLE
INSERT 0 1
INSERT 0 1
/tmp
asher011@wch133-38 $ source ./test.sh
Creating sequence...
CREATE SEQUENCE
Creating trigger and procedure...
ERROR:  relation "part_number_seq" already exists
ERROR:  language "plpgsql" already exists
CREATE FUNCTION
CREATE TRIGGER
Testing insert...
INSERT 0 1
 part_number
-------------
       50000
(1 row)

/tmp
asher011@wch133-38 $ source ./test.sh
Creating sequence...
ERROR:  relation "part_number_seq" already exists
Creating trigger and procedure...
ERROR:  relation "part_number_seq" already exists
ERROR:  language "plpgsql" already exists
CREATE FUNCTION
ERROR:  trigger "parttrigger" for relation "part_nyc" already exists
Testing insert...
INSERT 0 1
 part_number
-------------
       50000
       50001
(2 rows)
```

```
INSERT 0 1
 part_number
-------------
       50000
       50001
(2 rows)

/tmp
asher011@wch133-38 $ source ./test.sh
Creating sequence...
ERROR:  relation "part_number_seq" already exists
Creating trigger and procedure...
ERROR:  relation "part_number_seq" already exists
ERROR:  language "plpgsql" already exists
CREATE FUNCTION
ERROR:  trigger "parttrigger" for relation "part_nyc" already exists
Testing insert...
INSERT 0 1
 part_number
-------------
       50000
       50001
       50002
(3 rows)

/tmp
asher011@wch133-38 $
```

This is the contents of my triggers.sql file.

```
asher011@wch133-38 $ vi triggers.sql
CREATE SEQUENCE part_number_seq START WITH 50000;

CREATE LANGUAGE plpgsql;
CREATE OR REPLACE FUNCTION partnum()
RETURNS "trigger" AS
$BODY$
BEGIN
NEW.part_number := nextval('part_number_seq');
RETURN NEW;
END;
$BODY$
LANGUAGE plpgsql VOLATILE;

CREATE TRIGGER parttrigger BEFORE INSERT
ON part_nyc FOR EACH ROW
EXECUTE PROCEDURE partnum();
```

Also to run test.sh I had to make some adjustments to match my database:

```
asher011@wch133-38 $ vi test.sh
#!/bin/bash

echo "Creating sequence..."
cat <(echo 'CREATE SEQUENCE part_number_seq  START WITH 50000;')|psql -h localhost -p $PGPORT $USER"_DB"

echo "Creating trigger and procedure..."
psql -h localhost -p $PGPORT $USER"_DB" < triggers.sql

echo "Testing insert..."
cat <(echo "Insert into part_nyc(supplier, color, on_hand, descr) Values (0, 0, 20, 'Description')")|psql -h localhost -p
 $PGPORT $USER"_DB"
cat <(echo 'SELECT part_number FROM part_nyc WHERE on_hand=20;')|psql -h localhost -p $PGPORT $USER"_DB"
```

I then ran source ./stopPostgreDB.sh to shut down the database.

In elearn I will submit my triggers.sql file and my output screenshots along with this document.