Tutorial 1 – connecting to psql, downloading and running scripts

Running SQL scripts in postgres to create and populate tables:

1. Connect to psql on linux.socs.uoguelph.ca

   It is recommended that you create a new folder for 3530 on linux, change to that folder each time you want to connect to psql. For example:

```
ritu@george:~$ cd 3530
ritu@george:~/3530$ psql -h db
```

   Reminder: to connect from **off campus**, you must follow these steps:

   - connect to portkey.socs.uoguelph.ca
   - ssh linux.socs.uoguelph.ca
   - connect to psql (as shown in step 1)

2. This step is optional. You can create a schema and create tables in that schema. If you do not follow this step, all your tables will be created in a schema called public (which is also fine).

```
chaturvr=>
chaturvr=> CREATE SCHEMA SP;
CREATE SCHEMA
chaturvr=> SET SEARCH_PATH TO SP;
SET
chaturvr=> \d+
Did not find any relations.
```

   The above statement CREATE SCHEMA SP creates a schema in the current database. When we connect to postgres, the default schema is “public”. We can change that to the newly created schema SP by SET SEARCH_PATH TO.

   \d+ displays all relations in the current schema (we have none so far).

3. Download create_sp.sql and insert_sp.sql from Courselink (you will find it in Contents -> Week 4 and also in the dropbox for lab2). Script create_sp.sql has commands to create the required tables (S, P and SP). Script insert_sp.sql has commands to populate them.
4. To run a script on postgres, the command is \i scriptFileName. So you must run the given sql scripts on postgres as:

```
chaturvr=> \i create_sp.sql
CREATE TABLE
CREATE TABLE
CREATE TABLE
chaturvr=> \i insert_sp.sql
```

After the insert script runs successfully, \d+ should display these 3 relations S, P and SP.

```
chaturvr=> \d+
                        List of relations
 Schema | Name | Type  |  Owner   |    Size    | Description
--------+------+-------+----------+------------+-------------
 sp     | p    | table | chaturvr | 8192 bytes |
 sp     | s    | table | chaturvr | 8192 bytes |
 sp     | sp   | table | chaturvr | 8192 bytes |
(3 rows)
```

5. Verify that the relations are created and populated:

```
chaturvr=> select * from s;
 sno | sname | status |  city
-----+-------+--------+--------
 S1  | SMITH |     20 | LONDON
 S2  | JONES |     10 | PARIS
 S3  | BLAKE |     30 | PARIS
 S4  | CLARK |     20 | LONDON
 S5  | ADAMS |     30 | ATHENS
 S6  | HENRY |     12 | GUELPH
(6 rows)

chaturvr=> select * from p;
 pno | pname | color | weight |  city
-----+-------+-------+--------+--------
 P1  | NUT   | RED   |  12.00 | LONDON
 P2  | BOLT  | GREEN |  17.00 | PARIS
 P3  | SCREW | BLUE  |  17.00 | ROME
 P4  | SCREW | RED   |  14.00 | LONDON
 P5  | CAN   | BLUE  |  12.00 | PARIS
 P6  | COG   | RED   |  19.00 | LONDON
(6 rows)

chaturvr=> select * from sp;
 sno | pno | qty
-----+-----+-----
 S1  | P1  | 200
 S2  | P3  | 400
 S2  | P5  | 100
 S3  | P3  | 200
 S3  | P4  | 500
 S4  | P6  | 300
 S5  | P2  | 200
 S5  | P5  | 500
 S5  | P6  | 200
 S5  | P1  | 100
 S5  | P3  | 200
 S5  | P4  | 800
(12 rows)
```

6. To exit from postgres, type \q.