Lab 1. Installing Postgresql and Python libraries. Creating our database and schema

Our first lab will be to install Postgres and the relevant Python libraries as well as configure our first database.

First, we need to install Postgres. To do so, run the following commands:

```
$ sudo apt-get update
$ sudo apt-get install postgresql postgresql-contrib
$ sudo passwd postgres
```

Make sure to set a strong password for the postgres user account.

Next, we'll install the python setup tools.

```
$ sudo apt-get install python-setuptools
$ sudo apt-get install python3-pip
$ sudo pip3 install –upgrade pip
```

The last command upgrades pip (Python Install Program) to the latest version.

Now, we'll install the psycopg2 python library.

```
$ pip3 install psycopg2
```

By default, there is only one role (user) in the system, the postgres user. Let's create a new one for student1. First, we switch to the postgres account, and then we'll create the new user by running the createuser binary.

```
$ su – postgres
$ createuser --interactive
```

Create the student1 user and make the role a superuser account.

We are going to create a database called *ebooks*. We'll be using this database for our labs.

```
$ createdb ebooks
```

Now log on to the postgres sql command line by running the following:

```
$ psql –d ebooks
```

We will create our database schema using the following tables:

Table: Source

Column name	**Column type**
Source ID	Serial Primary key Not NULL incremental ascending
Source Description	VarChar(10)

Table: Subject

Column name	**Column type**
Subject ID	Serial Primary Key NOT NULL INCREMENTAL ASCENDING
Subject	Varchar (50)

Table: Publisher

Column name	**Column type**
Publisher ID	Serial Primary Key NOT NULL INCREMENTAL ASCENDING
Publisher Name	Varchar (50)

Table: EBook

Column name	**Column type**
ISBN	PRIMARY KEY VARCHAR (17)
Subject ID	FOREIGN KEY
Source ID	FOREIGN KEY
Publisher id	FOREIGN KEY
Book Title	VARCHAR (50)
Release Year	VARCHAR (4)
Status	VARCHAR (8)
Availability	DATE
Remarks	VARCHAR (100)

Write a Python program using the psycopg2 library to create all of the tables in this database. This program should have two python functions, *create_table()* and *drop_table()*.

The create table function will create the tables, drop table will drop the tables. Note that for tables that have fields that are being used in foreign key constraints, you will need to use the CASCADE option to DROP.