PostgreSQL

Robert M. Dondero, Ph.D
Princeton University

Objectives

- The lecture will:
  - Provide *a taste of* the **PostgreSQL** DBMS
  - Give you enough information about **PostgreSQL** to:
    - Help you decide if you want to use it in your project
    - Help you get started with it

PostgreSQL

- **Who**: Michael Stonebreaker, UC Berkeley
- **When**: 1996
- **Why**:
  - Successor to *Ingres*
  - Add the fewest features needed to completely support data types

Motivation

- Why study PostgreSQL?
  - Popular
    - Preferred by Heroku
  - Different
    - Contrasts with SQLite
    - More typical DBMS
      - Runs as a **process** that is distinct from the application process
  - Reasonable for COS 333 project
    - Used by many project teams

Installing PostgreSQL

- Linux (Debian, Ubuntu)
  - At a bash shell prompt:
    - `sudo apt-get install postgresql`
    - `sudo apt-get install libpq-dev`

Installing PostgreSQL

- Mac
  - Install Homebrew
    - Browse to https://brew.sh/ and follow instructions
  - At a bash shell prompt:
    - `brew install postgresql`

Installing PostgreSQL

- Mac (cont.)
  - Optionally, to enforce passwords
    - Edit file
      `/usr/local/var/postgres/pg_hba.conf`

```
local   all             rdondero                                trust
local   all             all                                     password
host    all             all             127.0.0.1/32            password
host    all             all             ::1/128                 password
local   replication     all                                     password
host    replication     all             127.0.0.1/32            password
host    replication     all             ::1/128                 password
```

  - Substitute your Mac username for `rdondero`

Installing PostgreSQL

- MS Windows
  - Use a browser to download `postgresql-12.3-1.exe` (or a newer version) from `enterprisedb.com/downloads/postgres-postgresql-downloads`
  - In File Manager double click on `postgresql-12.3-1.exe`
    - Use default settings
    - For “password for the database supervisor (postgres)” enter `xxx`.

Installing PostgreSQL

- MS Windows (cont.)
  - Add `C:\Program Files\PostgreSQL\12\bin` to your `PATH` environment variable

Starting the PostgreSQL Server

- Linux
  - At a bash shell prompt
    - `service postgresql start`
- Mac
  - At a bash shell prompt
    - `pg_ctl -D /usr/local/var/postgres start`
- MS Windows
  - PostgreSQL server is started automatically

Starting the PostgreSQL Server

- DBMS listens for connections at default port 5432

Creating a User and a Database

- Linux
  - At a bash shell prompt
    - `sudo -u postgres psql postgres`
  - At `psql` prompt
    - `CREATE ROLE rmd LOGIN PASSWORD 'xxx';`
    - `CREATE DATABASE bookstore WITH OWNER = rmd;`
    - `\q`

Creating a User and a Database

- Mac
  - At a bash shell prompt
    - `psql postgres`
  - At `psql` prompt
    - `CREATE ROLE rmd LOGIN PASSWORD 'xxx';`
    - `CREATE DATABASE bookstore WITH OWNER = rmd;`
    - `\q`

Creating a User and a Database

- MS Windows
  - In Command Prompt window
    - `psql -U postgres`
      - Enter `xxx` when prompted for password
  - At `psql` prompt
    - `CREATE ROLE rmd LOGIN PASSWORD 'xxx';`
    - `CREATE DATABASE bookstore WITH OWNER = rmd;`
    - `\q`

The psql Program

- At bash prompt or in Command Prompt window
  - `psql -h localhost -p 5432 -U rmd -d bookstore`
    - Enter password if prompted
- At `psql` prompt
  - `\i bookstore.sql`
  - `SELECT * FROM books;`
  - …
  - `\q`

15

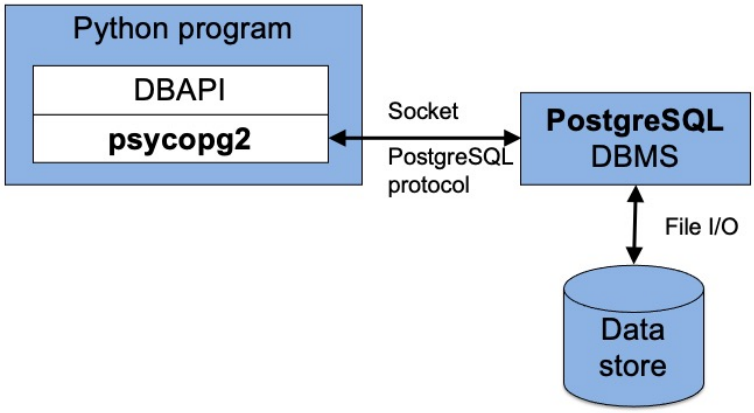
Just as SQLite has the sqlite3 command-line client…
So PostgreSQL has the psql command-line client

The psql Program

Some `psql` commands

psql Command	sqlite3 Command
`\h`	`.help`
`\q`	`.quit`
`\list`	`(list databases)`
`\d`	`.tables`
`\d` *table*	`.schema` *table*
`\i` *file*	`.read` *file*

PostgreSQL via Python



Installing PostgreSQL Driver

- Linux and MS Windows
  - At a bash shell prompt:
    - Activate your cos333 virtual env
    - `python -m pip install psycopg2`

Installing PostgreSQL Driver

- Mac
  - At a bash shell prompt:
    - Activate your cos333 virtual env
    - `env LDFLAGS="-I/usr/local/opt/openssl/include -L /usr/local/opt/openssl/lib" pip --no-cache install psycopg2`

PostgreSQL Pgmming in Python

- See **PostgreSQL/create.py**
  - No database file!
  - DBMS runs as a distinct process on a specified computer (localhost)
  - DBMS listens for connections at a known port (5432)

PostgreSQL Pgmming in Python

- See **PostgreSQL/display.py**
  - PostgreSQL uses standard SQL
- See **PostgreSQL/authorsearch.py**
  - PostgreSQL uses standard SQL
- See **PostgreSQL/order.py**
  - PostgreSQL uses standard SQL

PostgreSQL Pgmming in Python

- See **PostgreSQL/authorsearchprep.py**
  - PostgreSQL supports prepared statements
  - Note placeholder syntax
- See **PostgreSQL/orderprep.py**
  - (Much the same)

PostgreSQL Pgmming in Python

- See **PostgreSQL/purchase.py**
  - PostgreSQL supports transactions
- See **PostgreSQL/recovery.py**
  - Transactions work in PostgreSQL!

Stopping the PostgreSQL Server

- Linux
  - At a bash shell prompt
    - `service postgresql stop`
- Mac
  - At a bash shell prompt
    - `pg_ctl -D /usr/local/var/postgres stop`
- MS Windows
  - PostgreSQL server is stopped when computer shuts down

PostgreSQL Assessment

- PostgreSQL assessment (vs. SQLite)
  - (con) Setup is much more complex
  - (con) Programmatic use is a little more complex
  - (pro) Production-quality
    - Distinct process
    - Authentication
    - Row-level (vs. database-level) locking
    - …
  - (pro) Preferred by Heroku

Summary

- The lecture has:
  - Provided *a taste of* the **PostgreSQL** DBMS
  - Given you enough information about **PostgreSQL** to:
    - Help you decide if you want to use it in your project
    - Help you get started with it