CSE 180, Fall 2019
Week 1 Lab

Gradiance Accounts

PostgreSQL Access

Running SQL Scripts

Moving files from unix.ucsc.edu to your computers

A very simple SQL example

Cartesian Product

Be Sure to Read …

… GeneralInformation.pdf file that's on Piazza under Resources→General Resources

- Info about Gradiance, which we'll discuss
- Info about PostgreSQL accounts, which you should have
- Info about moving files between unix.ucsc.edu and your computer, so that you can post it on Canvas
- Sample SQL create, load and drop statements, which are also posted on Piazza in a zip file called BeerScriptsRI.zip that's under Resources→Lab Section Notes

Gradiance

> Short homework assignments

> Automatically graded

> Create an account : http://www.gradiance.com/services

> User ID : Cruz ID

> More info: http://www.gradiance.com/pub/stud-guide.html

> Use code **0283B382**

Canvas

> The primary webportal for UCSC class content.

https://canvas.ucsc.edu

> Submit lab assignments here.

> See your grades here.

PostgreSQL

- PostgreSQL is a major open-source relational database management system
  - https://www.postgresql.org/

- Class PostgreSQL server: **`cse180-db.lt.ucsc.edu`**

- Login Process

1. Using unix/linux-based terminal:

   ```
   my_computer $ ssh <CruzID>@unix.ucsc.edu
   ```

2. Using Putty

   ```
   Host Name : unix.ucsc.edu
   Login as : <CruzID>
   Password : <Blue Password>
   ```

PostgreSQL (cont'd)

3. From unix server to psql server :

```
unix4:~$ psql -h cse180-db.lt.ucsc.edu -U my_psql_username
my_psql_username-#
```

4. Change password (optional):

```
my_psql_username-# ALTER ROLE username WITH PASSWORD 'newpassword';
```

OR

```
my_psql_username-# \password
```

Components of a Database

> Schemas

```
CREATE SCHEMA Rel ;
```

> Relations

```
CREATE TABLE table_name (
    column_name_1 TYPE column_constraint,
    column_name_2 TYPE column_constraint
) ;
```

example_create.sql

example_create.sql

```
CREATE TABLE products (

    productID INT,

    name VARCHAR(80),

    price DECIMAL(10,2),

    retailPrice DECIMAL(10,2)

);
```

example_create.sql

```
CREATE TABLE products (

    productID INT,

    name VARCHAR(80),

    price DECIMAL(10,2),

    retailPrice DECIMAL(10,2)

);
```

[Syntax Lesson:]

DECIMAL(precision, scale)

precision := the total number of digits*
scale := the number of digits in the fraction part*

Price	Data Type
$12.99	DECIMAL(4,2), or DECIMAL(N,2) w/ N>=4
$5.99	DECIMAL(3,2), or DECIMAL(N,2) w/ N>=3
$199.99	DECIMAL(5,2), or DECIMAL(N,2) w/ N>=5
$3.998	DECIMAL(4,2), or DECIMAL(N,2) w/ N>=4
$1,499,999.98	DECIMAL(9,2), or DECIMAL(N,2) w/ N>=9

*http://www.postgresqltutorial.com/postgresql-numeric/

Loading Data into Tables :

1. From a CSV File :

   COPY table_name FROM ‘path_to_csv_file.csv’ DELIMITERS ‘,’ CSV;

2. To load data using stdin :

   COPY products FROM stdin USING DELIMITERS '|';

   1419|American Greetings CreataCard Gold V4.0|21.49|25.24

   1424|Barbie(R) Nail Designer(TM)|20.74|25.99

   1427|Panzer Commander|21.99|30.24

   1431|Riven: The Sequel to Myst|31.99|40.24

   \.

Getting Files from the Unix Timeshare

1. Copy and paste. (hint: does not scale)

2. SCP/SFTP

```
unix4:~$ ls <some_path>/lab1/
my_cool_soln.sql
unix4:~$

my_computer $ scp <ucsc_username>@unix.ucsc.edu:<some_path>/lab1/my_cool_soln.sql
<some_local_path>


Example:
[ ~ ]$ scp shel@unix.ucsc.edu:~/cmps180_f19/lab1/lab1_soln.sql /cse180/lab1
```

Getting Files from the Unix Timeshare

3. For the GUI people:

FileZilla: https://filezilla-project.org

Host : **unix.ucsc.edu**

UserName : **<CruzID>**
Password : **<Blue Password>**
Port : **22**
Drag & Drop the Required Files

Relational Model and Cartesian Product

- The relational data model (Edgar F. Codd, 1970)
  - Data is described and represented by the mathematical concept of a *relation*.

- What is a relation?
  - A structure with rows and columns
  - A subset of a Cartesian product of sets
  - What is the Cartesian product of {a,b,c,d} and {1,2,3}?

Cartesian Product

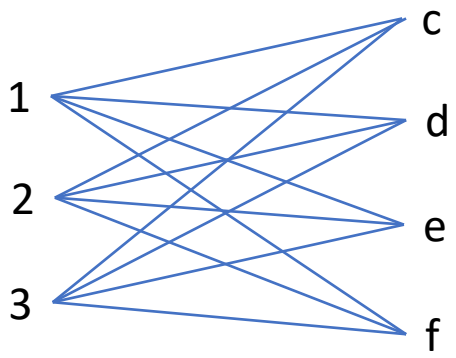
- What is the Cartesian product of {a,b,c,d} and {1,2,3}?

  { (a,1), (a,2), (a,3),
  (b,1), (b,2), (b,3),
  (c,1), (c,2), (c,3),
  (d,1), (d,2), (d,3) }

- What are some examples of relations from that Cartesian product?

Another Cartesian Product Example

- A: {1,2,3}
- B: {d,e,f,g}
- A × B = { (1,d), (1,e), (1,f), (1,g),
  (2,d), (2,e), (2,f), (2,g),
  (3,d), (3,e), (3,f), (3,g) }



- Suppose that C = {x,y}. What would A x B x C be?

Tuples and Relations

- *Tuple*:
  - A *k-tuple* is an ordered sequence of k values (not necessarily different)
    - (1,2) is a binary tuple or 2-tuple
    - (a,b,b) is a ternary tuple or 3-tuple
    - (112, 'Ann', 'CS', 'F', 3.95) is a 5-tuple

- If $D_1, D_2, \ldots, D_k$ are sets of elements, then the *Cartesian product* $D_1 \times D_2 \times \ldots \times D_k$ is the set of all k-tuples $(d_1, d_2, \ldots, d_k)$ such that $d_i \in D_i$, for all i with $1 \leq i \leq k$.

- *Relation*:
  - A *k-ary relation* is a subset of $D_1 \times D_2 \times \ldots \times D_k$, where each D_i is a set of elements
  - D_i is the *domain (or datatype)* of the *i*th column of the relation
  - Domains may be enumerated {'AMS', 'CMPS', 'TIM'}, or may be of standard types (INTEGER, FLOAT, DATE, …)