School of Computer Science
Uwe Roehm

DATA2001/2901: Data Science, Big Data, and Data Diversity **Semester 1, 2020**

Tutorial Week 3: Introduction to Relational Databases and SQL

In this tutorial you will learn how to access and analyse data in a relational database system such as PostgreSQL. We will use pgAdmin version III or IV to connect to your local or our central School of Computer Science's database server.

The central CS database server address is

`soitpw11d59.shared.sydney.edu.au` port: 5432

There, we have create you logins with the following format: `y20s1d2x01`_*yourUnikey* and **the database password is your SID** (note: this is not your normal unikey login/password!). For example, if your unikey is abcd1234 then your username would be y20s1d2x01_abcd1234.

Exercise 1. Database Preparation

We are working in this tutorial on the example WaterInfo database as introduced in the lecture. To generate and populate this database, perform the follow steps:

1. Login to our (or your own) PostgreSQL server using pgAdmin.
2. Open a new SQL query window.
3. Download `waterinfodata_ddl.sql` and `waterinfodata_dml.sql` files from Canvas
4. Open and run `waterinfodata_ddl.sql`
5. Open and run `waterinfodata_dml.sql`

This will generate a new schema, *WaterInfo* and four tables (following schema option 2 as discussed in this week's lecture):
`Measurements`, `Oganisations`, `Sensors` and `Stations`.

Exercise 2. SQL Querying

To execute SQL queries against the water info data, we need to first set our search path to WaterInfo schema (PostgreSQL supports multiple schemas, and to tell it which schema to use by default, it provides the search_path variable):

```
set search_path to WaterInfo;
```

As a first warm-up exercise, then execute and understand the following SQL queries:

```
SELECT * FROM Stations;
SELECT * FROM Measurements WHERE sensor='level';
SELECT COUNT(*) FROM Measurements WHERE obsdate<'2008-01-01';
SELECT * FROM Measurements JOIN Stations USING (stationid) LIMIT 10;
SELECT *
  FROM Measurements AS m JOIN Stations s USING (stationid)
 WHERE stationid = 409204;
```

Syntax of most commonly used SQL statements:

SQL Command	**Meaning**
`CREATE TABLE` T `( ... )`	creates a new table T; list the attributes in brackets in the form attribute type
`DROP TABLE` T	if needed - removes an existing table T
`INSERT INTO` T `VALUES (...)`	insert a new row into table T
`DELETE FROM` T	deletes all rows from table T; use a `WHERE` clause to restrict deletion to specific rows
`SELECT * FROM` T	list the full content of table T
`SELECT COUNT(*) FROM` T	count how many tuples are stored in table T
`SELECT AVG(attr) FROM` T	determine the average value of $attr$ in table T; similar with `MIN`(attr) and `MAX(attr)`

Now it is your turn – answer each of the following questions with an SQL query:

(a) List all sensors from the sensor table.

(b) List all organisations in alphabetical order of their name.

(c) List all stations where sampling is still ongoing (i.e. unknown 'cease' date).

(d) Find all non-zero measurements from the Measurements table.

(e) How many measurements from the Measurements table have a 0 value?

(f) List all measurements from a 'temp' sensor with an observed value above 28.

(g) List all measurements related to 'disc' sensors done before 01-01-2011.

(h) How many measurements related to 'level' sensor have been done at site 'Murray River at Barham' between 01-01-2008 and 31-12-2010?

(i) What was the average 'level' value measured at site 'Murray River at Barham' between 01-01-2008 and 31-12-2010?

(j) List all stations from the organisation 'Victoria Government' that have not ceased operation yet.

(k) List the five highest 'temp' measurements (just obsdate and obsvalue) at station 'Murray River at Swan Hill'.

(l) How many measurements have been done by the organisation named 'NSW Department of Water and Energy', and between which start and end year?

Exercise 3. SQL Online Tutorial in Grok

We have made available an online tutorial covering all parts of SQL in 8 modules. It is linked in Canvas under 'Modules - Unit Information - SQL Online Tutorial'.

Please go through this tutorial in your own time and finish all eight modules by Week 7 as preparation for the (online) SQL quiz in Week 8.

Exercise 4. Advanced Task (DATA2901)

(a) Advanced students should load the raw CSV and TSV (tabs-separated) data files from the WaterInfo scenario into `SQLite`. They are linked in Canvas. Note that only two of those files are comma-separated, the other two are using tabs!

(b) Add to the `Measurements` table in sqlite a foreign key column stationid that links the Measurements table with the Stations table on stationid. To do so, you will need to transform the identifiers of the stations in the stations CSV file though first. There are different ways on how you can achieve this: Either add a new combined attribute to the imported Stations table (ALTER TABLE statement and an UPDATE with string concatenation), or with a Unix script before the actual import. Try both.

(c) Once you have now loaded the data in SQLite, try to answer the same questions as above.

(d) Optional: Solve subquestion (c) inside a Jupyter notebook using the ipython-sql extension.