DBDM – Data Bases and Data Mining
Real-life dataset import case study

romuald.thion@univ-lyon1.fr
http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/

M1 Informatique Fondamentale 2016–2017

Abstract

The objective of this lab session is to analyse a real-life tabular dataset provided as a spreadsheet and to import it into a structured relational database.

1 Setup

The dataset is a spreadsheet in CSV (*Comma Separated Value*) format containing statistics about professional insertion of graduate students from public French universities. It can be opened with LibreOffice for instance.

http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/TPCSV_fr-esr-insertion.csv
http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/TPCSV_fr-esr-insertion_utf8.csv

The first step is to import the raw dataset into a `TPCSV_INS_BRUT` table using Sqlite3. The schema is given by the `TPCSV_Structure.sql`[1] file. Comments in English describe the different attributes. The `init_script.txt`[2] is provided to ease the import, as shown in the following script :

```
$ sqlite3 -init init_script.txt db.sqlite3
-- Loading resources from init_script.txt

SQLite version 3.8.11.1 2015-07-29 20:00:57
Enter ".help" for usage hints.
sqlite> .q
```

Write a simple query in a file named `query.sql` an try it to check if everything work correctly.

```
$ sqlite3 -echo db.sqlite3 < query.sql
```

[1] http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/TPCSV_Structure.sql
[2] http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/init_script.txt

2 Analysis

Exercise 1 Setup

1. Check that the number of lines in the CSV file is consistent with the number of tuples in the `TPCSV_INS_BRUT` relation.

Exercise 2 Fictitious value

In a domain (attribute `CODE_DOMAINE`) with several disciplines, the first discipline in the lexicographic order is *fictitious*: it is added to store aggregated statistics at the domain level. For example, `disc07` is the fictitious discipline of the `SHS` domain. Similarly, a fictitious university `UNIV` exists to materialize statistics at the national level. Whereas this approach can be debated for a spreadsheet, it is non-sense from a relational perspective.

1. Write a query that compares, for each year and each domain, the number of answers associated to the fictitious discipline and the sum of answers from real disciplines. *There should be no difference*, so suppressing these tuples won't delete any information. You may use *subqueries*[3] in the `FROM` clause and the `IN`[4] operator in the `WHERE` clauses.
2. Delete line involving fictitious disciplines or the fictitious university.

3 Normalization

Exercise 3 Dependencies and normalization

Now the dataset has been somehow cleaned but it still contains a lot of redundancies. For instance, it is intuitive that the `NUM_ETABLISSEMENT` attribute uniquely determines the `NOM_ETABLISSEMENT` attribute, and vice-versa. The job is now to identify the existing functional dependencies to normalize the database.

1. Find a minimal key (i.e., a set of attributes) for `TPCSV_INS_BRUT`. Check the "keyness" with a SQL query and justify its minimality.
2. Find all the relevant functional dependencies between disciplines, domains, academies, universities and statistics. Check your intuition with SQL queries.
3. Based on your findings, create a normalized database[5], in Boyce-Code Normal Form, that keeps all the information initially provided, without redundancies. You should end up with 4 or 5 relations. Pay particular attention to `PRIMARY KEY` and `FOREIGN KEY` constraints[6] when you define your schema.
4. Populate your new schema with `INSERT INTO` queries using `SELECT` statements[7]. Mind the order of insertions to ensure that `FOREIGN KEY` constraints are satisfied.
5. Verify that the number of tuples in `TPCSV_INS_BRUT` is consistent with the normalized data you have now.
6. A *view* is essentially a *named query* that can be used as a full-fledged relation in `FROM` clauses. Define a view [8] that computes the data you suppressed in Exercise 2.

4 Bonus

Exercise 4 Additional exercise

A more exhaustive list of universities is given :

`http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/TPCSV_fr-esr-etablissements.csv`
`http://liris.cnrs.fr/romuald.thion/files/Enseignement/DBDM/TPCSV_fr-esr-etablissements_utf8.csv`

[3] `http://www.techonthenet.com/sqlite/subqueries.php`
[4] `https://www.sqlite.org/lang_select.html`
[5] `https://www.sqlite.org/lang_createtable.html`
[6] `https://www.sqlite.org/foreignkeys.html`
[7] `https://www.sqlite.org/lang_insert.html`
[8] `https://www.sqlite.org/lang_createview.html`

1. Find a couple of interesting queries (e.g., the number of times a discipline has the highest employment rate in a university)
2. Do the same job you've done with professional insertion statistics with this additional dataset.