Introduction to Databases

- This document: https://docs.google.com/document/d/11mRo_nH-CRJYH_0EDcu3Z60MZaO_qJdVl0l5vM2W1e8/edit?usp=sharing

- The slides set: https://www.dropbox.com/s/r9mcwggb6joub5u/MC3.pptx

- The YouTube broadcast of the Google Hangouts on Air presentation: tba

Agenda

1. Recap Google Datastore
2. Introduction to Databases
3. Exercises on Database design and SQL
4. SQL vs. NoSQL(http://tinyurl.com/orbital-sql-nosql)
5. Homework

Exercise

1. Design a database for maintaining inventory data of NTUC FairPrice, the largest grocery retailer in Singapore with a network of more than 230 outlets.
   This exercise is more on theory (ERD and schema design), and further information can be found in the slide and video of this mission control.

2. Getting started with SQLite tool
   a. Download the tool from http://www.sqlite.org/download.html
      (sqlite-shell-win32 for windows, sqlite-shell-osx for mac)
   b. Launch the tool (double click the downloaded execution file)
   c. Try basic commands (see more examples in http://www.sqlite.org/cli.html)
      ```
      sqlite> .open "C:/mc3.db"
      sqlite> create table tbl1(id varchar(10), value smallint);
      sqlite> insert into tbl1 values('hello!',10);
      sqlite> insert into tbl1 values('goodbye', 20);
      sqlite> select * from tbl1;
      ```

3. Populate a sample of the above inventory database.
   a. Download the exercise file from
      https://www.dropbox.com/s/7m2duy18wyqo5em/mc3data.zip
   b. Unzip the downloaded file to a folder, e.g., C:/mc3data
   c. Execute the following commands in SQLite:

```
sqlite> .read "C:/mc3data/create.sql"
sqlite> .read "C:/mc3data/store.sql"
sqlite> .read "C:/mc3data/product.sql"
sqlite> .read "C:/mc3data/inventory.sql"
```

4. Translate the following queries into SQL and execute them.
    a. List the different streets in which there is a store.

    ```
    sqlite> SELECT DISTINCT street FROM store;
    ```

    b. List the different streets in which there is a store in the CLEMENTI area.

    ```
    sqlite> SELECT DISTINCT s.street FROM store s WHERE s.area='CLEMENTI';
    ```

    c. List the different streets where there is more than one store.

    ```
    sqlite> SELECT DISTINCT s1.street FROM store s1, store s2 WHERE
    s1.postal_code <> s2.postal_code and s1.street = s2.street;
    ```

    d. List the addresses of stores keeping more than 800 jars of PETER PAN PEANUT BUTTER – CREAMY.

    ```
    sqlite> SELECT s.blkno, s.street FROM product p , inventory v, store s WHERE
    p.name='PETER PAN PEANUT BUTTER - CREAMY' AND v.id=p.id AND
    v.postal_code=s.postal_code AND v.quantity>= 800;
    ```

    e. List the addresses of the stores, the names of the products that they carry where the names contain ‘wine’, together with the dollar value of their stock of these products, for each product. Order the results by postal codes of the stores and by category of the products.

    ```
    sqlite> SELECT s.blkno, s.street, p.name, v.quantity * p.price FROM product p ,
    inventory v, store s WHERE v.id=p.id AND v.postal_code=s.postal_code AND
    p.name LIKE '%wine%' ORDER BY s.postal_code, p.category;
    ```

Technical Terminology

Relational database - *a database that has a collection of tables of data items, all of which is formally described and organized according to the relational model. Data in a single table represents a relation, from which the name of the database type comes. In typical solutions, tables may have additionally defined relationships with each other.*

More detail: (from http://en.wikipedia.org/wiki/Relational_database) In the relational model, each table schema must identify a column or group of columns, called the primary key, to uniquely identify each row. A relationship can

then be established between each row in the table and a row in another table by creating a foreign key, a column or group of columns in one table that points to the primary key of another table.

Relational database management system (RDBMS) – *a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd. Many popular databases currently in use are based on the relational database model.*
More detail: http://en.wikipedia.org/wiki/Relational_database_management_system

Entity–relationship model - *a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.*
More detail: http://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model

SQL - *SQL stands for Structured Query Language, which lets you access and manipulate relational databases and is an ANSI (American National Standards Institute) standard.*
More detail: http://en.wikipedia.org/wiki/SQL

NoSQL - *Next Generation Databases mostly addressing some of the points: being non-relational, distributed, open-source and horizontally scalable.*
More detail: http://en.wikipedia.org/wiki/NoSQL

Resources

1. http://www.sqlite.org/cli.html : Command Line Shell For SQLite.
2. http://www.sqlite.org/lang.html : SQL As Understood By SQLite.
3. http://www.sqlite.org/whentouse.html : Situations Where SQLite Works Well and Situations Where Another RDBMS May Work Better.

4. http://www.php.net/manual/en/sqlite3.open.php , http://www.tutorialspoint.com/sqlite/sqlite_php.htm : Accessing SQLite databases from PHP.

5. http://www.tutorialspoint.com/sqlite/sqlite_python.htm : Accessing SQLite databases from Python.

6. http://www.w3schools.com/sql/sql_intro.asp : Introduction to SQL.
7. http://nosql-database.org/ : List of NoSQL databases[currently 150].

Homework

Design a database for a patient appointment system:

- A doctor can be scheduled for many appointments, but may not have any scheduled at all. Each appointment is scheduled with exactly 1 doctor.
- A patient can schedule 1 or more appointments. One appointment is scheduled with exactly 1 patient.
- An appointment must generate exactly 1 bill, a bill is generated by only 1 appointment.
- One payment is applied to exactly 1 bill, and 1 bill can be paid off over time by several payments. A bill can be outstanding, having nothing yet paid on it at all. One patient can make many payments, but a single payment is made by only 1 patient.
- Some patients are insured by an insurance company. If they are insured, they can only carry insurance with one company.
- An insurance company can have many patients carry their policies. For patients that carry insurance, the insurance company will make payments, each single payment is made by exactly 1 insurance company.