A brief introduction to MySQL

Ch. Morisset
IA-UNAM

History

MySQL is a relational database management system (RDBMS). It was created in 1995.

It is the second widely used (after SQLite, which is included in any Android and iOS device...)

SQL stands for “Structured Query Language”.

The “My” comes from the name of the co-founder daughter, My.

In 2008, Sun Microsystems bought MySQL for $1 billion. In 2009 Oracle entered into an agreement to purchase Sun and to continue to enhance MySQL.

In January 2009, prior to Oracle's acquisition of Sun and MySQL, Monty Widenius started a GPL-only fork, MariaDB.

SQL

- The request language is SQL. Other RDBMS using SQL are Oracle, PostgreSQL, SQLite, Microsoft SQL server, Microsoft Access, between many others.
- Some small differences in syntax can exist between the RDBMS.
- ADQL is the Astronomical Database Query Language, used in Virtual Observatory, see: http://www.ivoa.net/documents/latest/ADQL.html

Client and server

- Most of the users will only need to have a **client** access to a database, but not to manage themselves a database.
- It's similar to have an access to the web using a browser. Everybody does it, it's easy. Another story is having a server managing its own web pages.
- Almost every Linux distribution comes with MySQL or MariaDB installed.
- OSX: have to install from MySQL web page (http://dev.mysql.com/downloads/mysql/). Need to register to Oracle.

Connect to a db

To connect to the 3MdB database (as an example):

```
mysql -h 3mdb.astro.unam.mx 3MdB_17 -u OVN_user -p

Enter password: *********

mysql>help
```

Basic commands

```
mysql> show tables; # list the tables available in the current
base
+-------------------+
| Tables_in_3MdB_17 |
+-------------------+
| abion_17          |
| lines_17          |
| seds_17           |
| tab_17            |
| teion_17          |
| temis_17          |
+-------------------+
```

Basic commands

```
mysql> describe lines_17;
+--------+-------------+------+-----+---------+----------------+
| Field  | Type        | Null | Key | Default | Extra          |
+--------+-------------+------+-----+---------+----------------+
| Nl     | bigint(20)  | NO   | PRI | NULL    | auto_increment |
| label  | varchar(15) | YES  |     | NULL    |                |
| id     | varchar(20) | YES  |     | NULL    |                |
| lambda | double      | YES  |     | NULL    |                |
| name   | varchar(40) | NO   |     | NULL    |                |
| used   | int(2)      | YES  |     | 1       |                |
+--------+-------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)
```

Basic commands

SELECT is the command to obtain a result from a query.

```
mysql> select count(*) from tab_17; # number of elements
+----------+
| count(*) |
+----------+
|  1582006 |
+----------+

mysql> select max(N) from tab_17; # arithmetic operations are available
+---------+
| max(N)  |
+---------+
| 1582006 |
+---------+
```

Aliases and limit

```
mysql> SELECT min(N) AS MIN, max(N) AS MAX FROM tab_17;
+---------+---------+
| MIN     | MAX     |
+---------+---------+
|    1    | 1582006 |
+---------+---------+

mysql> SELECT id, lambda, name FROM lines_17 LIMIT 10;
+------+---------+---------------+
| id   | lambda  | name          |
+------+---------+---------------+
| Bac  |    3646 | Bac  3646.0A  |
| cout |    3646 | cout 3646.0A  |
| cref |    3646 | cref 3646.0A  |
| H  1 | 4861.33 | H  1 4861.33A |
| Ca B | 4861.33 | Ca B 4861.33A |
| H  1 | 6562.81 | H  1 6562.81A |
| Ca B | 6562.81 | Ca B 6562.81A |
| H  1 | 4340.46 | H  1 4340.46A |
| Ca B | 4340.46 | Ca B 4340.46A |
| H  1 | 3889.05 | H  1 3889.05A |
+------+---------+---------------+
```

Where

```
mysql> SELECT name FROM lines_17 WHERE lambda > 5000 AND lambda <
5500;
+------------------+
| name             |
+------------------+
| N  1 5197.9A     |
| N  1 5200.26A    |
| N  1 5199.0A+    |
| N 1R 5199.0A     |
| N 2R 5005.0A     |
| N 2R 5005.15A PN |
| N 2R 5001.14A PN |
| N 2R 5001.48A PN |
| O  3 5006.84A    |
| Ar 3 5191.82A    |
| Fe 3 5270.40A    |
| Fe 6 5145.76A    |
+------------------+
```

Where and order

```
mysql> SELECT count(*) from lines_17 WHERE lambda > 5000 AND lambda < 6000;
+----------+
| count(*) |
+----------+
|       26 |
+----------+

mysql> SELECT name from lines_17 WHERE lambda > 5000 AND lambda < 5500 ORDER BY lambda;
+------------------+
| name             |
+------------------+
| N 2R 5001.14A PN |
| N 2R 5001.48A PN |
| N 2R 5005.0A     |
| N 2R 5005.15A PN |
| O  3 5006.84A    |
| Fe 6 5145.76A    |
| Ar 3 5191.82A    |
| N  1 5197.9A     |
| N 1R 5199.0A     |
| N  1 5199.0A+    |
| N  1 5200.26A    |
| Fe 3 5270.40A    |
+------------------+
```

Count and group

```
mysql> SELECT ref, count(*) AS number FROM tab_17 GROUP BY
ref ORDER by number;
+----------+--------+
| ref      | number |
+----------+--------+
| CALIFA   |  84208 |
| BOND     | 113420 |
| CALIFA_2 | 154405 |
| BOND_2   | 210600 |
| PNe_2021 | 294987 |
| PNe_2020 | 724386 |
+----------+--------+
```

Join tables

In some databases, the data are disseminated in multiple tables.

Keys are used to associate entries from one table with entries from another table.

Ex: N in `tab` and N in `teion` are referring to the same model.

Join tables

```
mysql> SELECT
   O__3_500684A/H__1_486133A as O3,
   N__2_658345A/H__1_486133A as N2,
   T_OXYGEN_VOL_2 as T_O3
FROM
   tab_17, teion_17
WHERE
   tab_17.N = teion_17.N
   AND
   tab_17.ref = 'PNe_2020' # need the tab_17.ref, as ref is also in teion_17
LIMIT
   10;

+------------------------+--------------------+-------------------+
| O3                     | N2                 | T_O3              |
+------------------------+--------------------+-------------------+
|      12.26256147084059 |  9.895269181969748 | 8597.189525129013 |
|     7.8322296370914195 |  8.142016751116824 | 7960.999703633087 |
| 0.00019526195904746857 | 1.8043785472665244 | 5374.200535435795 |
| 0.00021171427179948898 | 1.9535294329419632 | 5374.200535435795 |
|     23.848914888569013 |  13.16919604150879 |  9909.79468166116 |
|     14.009960081674256 |   7.52666817544795 |  8597.25784469428 |
| 0.00023484565503742043 | 2.1509429803073057 | 5374.200535435796 |
| 0.00027635902289709834 |  2.466695457624854 | 5434.241275749856 |
| 0.00019395293502557574 | 1.7923518587384621 | 5374.200535435795 |
|     11.573139408261161 |  9.472795211519573 | 8597.144794598968 |
+------------------------+--------------------+-------------------+
```

Format the output

Using ROUND or FORMAT functions:

```
mysql> SELECT
  ROUND(O__3_500684A/H__1_486133A, 3) as O3,
  ROUND(N__2_658345A/H__1_486133A,3) as N2,
  ROUND(T_OXYGEN_VOL_2,0) as T_O3
FROM
  tab_17, teion_17
WHERE
  tab_17.N = teion_17.N
AND
 tab_17.ref = 'PNe_2020'
LIMIT     10;
+--------+--------+------+
| O3     | N2     | T_O3 |
+--------+--------+------+
| 12.263 |  9.895 | 8597 |
|  7.832 |  8.142 | 7961 |
|  0.000 |  1.804 | 5374 |
|  0.000 |  1.954 | 5374 |
| 23.849 | 13.169 | 9910 |
| 14.010 |  7.527 | 8597 |
|  0.000 |  2.151 | 5374 |
|  0.000 |  2.467 | 5434 |
|  0.000 |  1.792 | 5374 |
| 11.573 |  9.473 | 8597 |
+--------+--------+------+
```

Running requests

Put the request into a file (e.g. req1.sql) containing for example:

```
SELECT     ROUND(O__3_500684A/H__1_486133A, 3) as O3,
ROUND(N__2_658345A/H__1_486133A,3) as N2,     ROUND(T_OXYGEN_VOL_2,0)
as T_O3  FROM     tab_17, teion_17  WHERE     tab_17.N = teion_17.N
AND     tab_17.ref = 'PNe_2020'
```

Notice that there is no LIMIT anymore (a lot of results). Run it from a terminal:

```
mysql -h 3mdb.astro.unam.mx 3MdB_17 -u OVN_user -p < req1.sql >
req1.res
```

The result is obtained in less than a second and store in req1.res. It contains 7855 lines, easy to read from python, starting like:

```
O3      N2      T_O3
0.4108  1.1447  5409.4
2.5814  0.1484  18626.0
0.0021  0.9481  3460.5
4.1895  0.0581  11935.3
0.0251  0.1854  5271.2
4.7034  0.0064  16001.7
2.6313  0.0036  18857.0
1.6673  0.6845  14112.8
1.6881  0.1417  6489.9
6.5569  0.3675  13707.0
...
```

Special functions

- The reference manual is 3500 pages big!...
- I'll here describe a few number of functions, have a look at the online doc for more.

Strings

- Substring, reverse, ltrim, etc...
- like and %:

```
mysql> SELECT count(*), ref FROM tab_17
WHERE ref LIKE 'PNe_202%' GROUP BY ref;
+----------+----------+
| count(*) | ref      |
+----------+----------+
|   724386 | PNe_2020 |
|   294987 | PNe_2021 |
+----------+----------+
```

http://www.tutorialspoint.com/mysql/mysql-string-functions.htm

Numeric functions

- Log10, avg, sqrt, pow, abs, sin, …
- http://www.tutorialspoint.com/mysql/mysql-numeric-functions.htm

Optimization

- EXPLAIN before SELECT.

```
mysql> EXPLAIN SELECT
ROUND(O__3_500684A/H__1_486133A, 3) as O3,
ROUND(N__2_658345A/H__1_486133A,3) as N2,
ROUND(T_OXYGEN_VOL_2,0) as T_O3  FROM     tab_17,
teion_17  WHERE     tab_17.N = teion_17.N     AND
 tab_17.ref = 'PNe_2020'  LIMIT     10;
+------+-------------+----------+--------+---------------+---------+---------+------------------+--------+-----------------------+
| id   | select_type | table    | type   | possible_keys | key     | key_len | ref              | rows   | Extra                 |
+------+-------------+----------+--------+---------------+---------+---------+------------------+--------+-----------------------+
|    1 | SIMPLE      | tab_17   | ref    | PRIMARY,ref   | ref     | 122     | const            | 724516 | Using index condition |
|    1 | SIMPLE      | teion_17 | eq_ref | PRIMARY       | PRIMARY | 8       | 3MdB_17.tab_17.N |      1 |                       |
+------+-------------+----------+--------+---------------+---------+---------+------------------+--------+-----------------------+
```

Links

- https://www.mariadbtutorial.com/
- http://www.mysqltutorial.org/