Skill Pill: Database Manipulation — Cheat Sheet

More on https://en.wikibooks.org/wiki/MySQL/CheatSheet

CONNECT/DISCONNECT

Connecting to the OIST MySQL server and launching MySQL.

```
$ ssh <OIST_username>@skillpills
$ mysql -u <user> -p -h <host>
mysql> exit
```

BROWSING

Browsing through the content of the MySQL server.

```
SHOW DATABASES
SHOW TABLES
DESCRIBE table
USE database
```

QUERY

Sending requests.

```
SELECT * FROM table
SELECT field1, field2, ... FROM table
SELECT ... FROM ... WHERE condition
SELECT ... FROM ... WHERE condition ORDER BY field1,
    field2 DESC
SELECT ... FROM ... WHERE condition GROUP BY field
SELECT ... FROM ... WHERE condition GROUP BY field HAVING
    condition2
SELECT ... FROM ... WHERE condition LIMIT 10
SELECT DISTINCT field1, field2 FROM ...

SELECT ... FROM table1, table2, ...
SELECT ... FROM t1 JOIN t2 ON t1.id1 = t2.id2 WHERE
    condition
SELECT ... FROM t1 LEFT JOIN t2 ON t1.id1 = t2.id2 WHERE
    condition
SELECT ... FROM t1 RIGHT JOIN t2 ON t1.id1 = t2.id2 WHERE
    condition
SELECT ... FROM t1 JOIN (t2 JOIN t3 ON ...) ON ...
SELECT ... FROM t1 JOIN t2 USING(id)
(SELECT a1,b1,c1 FROM ...) UNION (SELECT a1,b2,c2 FROM
    ...)
```

CONDITIONALS

Adding conditions to the requests.

```
field1 = value1
field1 <> value1
field1 LIKE ’value_’
field1 LIKE ’value%’
field1 IS NULL
field1 IS NOT NULL
field1 IN (value1, value2)
field1 NOT IN (value1, value2)
condition1 AND condition2
condition1 OR condition2
```

AGGREGATE FUNCTIONS

Grouping and analyzing data.

```
COUNT(), COUNT(DISTINCT )
MIN(), MAX()
SUM(), AVG(), STD()
```

MODIFYING DATABASE

Creating new databases.

```
CREATE DATABASE [IF NOT EXISTS] mabase [CHARACTER SET
    charset] [COLLATE collation]
CREATE DATABASE mabase CHARACTER SET utf8
DROP DATABASE mabase
ALTER DATABASE mabase CHARACTER SET utf8
```

ENTRY MANIPULATION

Adding, deleting or updating values.

```
INSERT INTO table1 (f1, f2, ...) VALUES (v1, v2, ...)
INSERT table1 SET field1=value_1, field2=value_2 ...

LOAD DATA INFILE ’/tmp/mydata.txt’ INTO TABLE table1
FIELDS TERMINATED BY ’,’ OPTIONALLY ENCLOSED BY ’"’
    ESCAPED BY ’\n’

DELETE FROM table1 / TRUNCATE table1
DELETE FROM table1 WHERE condition
DELETE FROM table1, table2 WHERE table1.id1 = table2.id2
    AND condition

UPDATE table1 SET field1=new_value1 WHERE condition
UPDATE table1, table2 SET field1=new_value1,
    field2=new_value2, ... WHERE table1.id1 = table2.id2
    AND condition
```

TABLE MANIPULATION

Modyfing tables.

```
CREATE TABLE table (field1 type1, field2 type2, ...)
CREATE TABLE table (field1 type1 unsigned not null
    auto_increment, field2 type2, ...)
CREATE TABLE table (field1 type1, field2 type2, ...,
    INDEX (field))
CREATE TABLE table (field1 type1, field2 type2, ...,
    PRIMARY KEY (field1))
CREATE TABLE table1 (fk_field1 type1, field2 type2, ...,
    FOREIGN KEY (fk_field1) REFERENCES table2 (t2_fieldA)
    [ON UPDATE] [CASCADE|SET NULL|RESTRICT]
    [ON DELETE] [CASCADE|SET NULL|RESTRICT])
CREATE TABLE table IF NOT EXISTS (...)
CREATE table new_table_name as SELECT [ *|column1,
    column2 ] FROM table_name

DROP TABLE table1, table2, ...
DROP TABLE IF EXISTS table

ALTER TABLE old_name RENAME new_name;
ALTER TABLE table MODIFY field1 type1
ALTER TABLE table MODIFY field1 type1 NOT NULL ...
ALTER TABLE table CHANGE old_name_field1 new_name_field1
    type1
ALTER TABLE table ADD new_name_field1 type1
ALTER TABLE table ADD new_name_field1 type1 FIRST
ALTER TABLE table ADD new_name_field1 type1 AFTER
    another_field
ALTER TABLE table DROP field1
ALTER TABLE table ADD INDEX (field);
ALTER TABLE table ADD PRIMARY KEY (field);
ALTER TABLE table CHANGE old_name_field1 new_name_field1
    type1 FIRST
```