A brief MySQL tutorial

CSE 134A: Web Service Design and Programming

Fall 2001

9/28/2001

Creating and Deleting Databases

```
1) Creating a database

mysql> CREATE database 134a;
Query OK, 1 row affected (0.00 sec)

2) Deleting a database

mysql> DROP database 134a;
Query OK, 0 rows affected (0.00 sec)
```

Creating a Table

3) After we have created the database we use the USE statement to change the current database;

```
mysql> USE 134a;
Database changed
```

4) Creating a table in the database is achieved with the CREATE table statement

```
mysql> CREATE TABLE president (
    ->     last_name varchar(15) not null,
    ->     first_name varchar(15) not null,
    ->     state varchar(2) not null,
    ->     city varchar(20) not null,
    ->     birth date  not null default '0000-00-00',
    ->     death date null
    -> );
Query OK, 0 rows affected (0.00 sec)
```

Examining the Results

5) To see what tables are present in the database use the SHOW tables:

```
mysql> SHOW tables;
+----------------+
| Tables_in_134a |
+----------------+
| president      |
+----------------+
1 row in set (0.00 sec)
```

6) The command DESCRIBE can be used to view the structure of a table

```
mysql> DESCRIBE president;
+------------+-------------+------+-----+------------+-------+---------------------------------+
| Field      | Type        | Null | Key | Default    | Extra | Privileges                      |
+------------+-------------+------+-----+------------+-------+---------------------------------+
| last_name  | varchar(15) |      |     |            |       | select,insert,update,references |
| first_name | varchar(15) |      |     |            |       | select,insert,update,references |
| state      | char(2)     |      |     |            |       | select,insert,update,references |
| city       | varchar(20) |      |     |            |       | select,insert,update,references |
| birth      | date        |      |     | 0000-00-00 |       | select,insert,update,references |
| death      | date        | YES  |     | NULL       |       | select,insert,update,references |
+------------+-------------+------+-----+------------+-------+---------------------------------+
6 rows in set (0.00 sec)
```

Inserting / Retrieving Data into / from Tables

7) To insert new rows into an existing table use the INSERT command:

```
mysql> INSERT INTO president values ('Washington',
                                     'George',
                                     'VA',
                                     'Westmoreland County',
                                     '17320212',
                                     '17991214');
Query OK, 1 row affected (0.00 sec)
```

8) With the SELECT command we can retrieve previously inserted rows:

```
mysql> SELECT * FROM president;
+------------+------------+-------+---------------------+------------+------------+
| last_name  | first_name | state | city                | birth      | death      |
+------------+------------+-------+---------------------+------------+------------+
| Washington | George     | VA    | Westmoreland County | 1732-02-12 | 1799-12-14 |
+------------+------------+-------+---------------------+------------+------------+
1 row in set (0.00 sec)
```

Selecting Specific Rows and Columns

```
9) Selecting rows by using the WHERE clause in the SELECT command

mysql> SELECT * FROM president WHERE state="VA";
+------------+------------+-------+---------------------+------------+------------+
| last_name  | first_name | state | city                | birth      | death      |
+------------+------------+-------+---------------------+------------+------------+
| Washington | George     | VA    | Westmoreland County | 1732-02-12 | 1799-12-14 |
+------------+------------+-------+---------------------+------------+------------+
1 row in set (0.00 sec)

10) Selecting specific columns by listing their names

mysql> SELECT state, first_name, last_name FROM president;
+-------+------------+------------+
| state | first_name | last_name  |
+-------+------------+------------+
| VA    | George     | Washington |
+-------+------------+------------+
1 row in set (0.00 sec)
```

Deleting and Updating Rows

11) Deleting selected rows from a table using the DELETE command

```
mysql> DELETE FROM president WHERE first_name="George";
Query OK, 1 row affected (0.00 sec)
```

12) To modify or update entries in the table use the UPDATE command

```
mysql> UPDATE president SET state="CA" WHERE first_name="George";
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Loading a Database from a File

13) Loading a your data from a file into a table.

Assuming we have a file named "president_db" in the current directory, with multiple INSERT commands in it, we can use the LOAD DATA command to insert the data into the table president.

```
mysql> LOAD DATA LOCAL INFILE 'president_db' INTO TABLE president;
Query OK, 45 rows affected (0.01 sec)
Records: 45  Deleted: 0  Skipped: 0  Warnings: 0
```

Note, that any ascii file that contains a valid sequence of MySql commands on separate lines can be read in from the command line as:

```
>mysql -u USERNAME -p < MY_Mysql_FILE
```

More on SELECT

A general form of SELECT is:

SELECT ***what to select***
FROM ***table(s)***
WHERE ***condition that the data must satisfy***;

Comparison operators are: < ; <= ; = ; != or <> ; >= ; >
Logical operators are: AND ; OR ; NOT
Comparison operator for special value NULL: IS

More on SELECT (cont.)

14) The following MySQL query will return all the fields for the presidents whose state field is "NY";

```
mysql> SELECT * FROM president WHERE state="NY";
+-----------+-------------+-------+---------------+------------+------------+
| last_name | first_name  | state | city          | birth      | death      |
+-----------+-------------+-------+---------------+------------+------------+
| Van Buren | Martin      | NY    | Kinderhook    | 1782-12-05 | 1862-07-24 |
| Fillmore  | Millard     | NY    | Cayuga County | 1800-01-07 | 1874-03-08 |
| Roosevelt | Theodore    | NY    | New York      | 1858-10-27 | 1919-01-06 |
| Roosevelt | Franklin D. | NY    | Hyde Park     | 1882-01-30 | 1945-04-12 |
+-----------+-------------+-------+---------------+------------+------------+
4 rows in set (0.00 sec)
```

More on SELECT (cont.)

15) We can limit the values of the returned fields as it is shown bellow:

```
mysql> SELECT last_name, first_name FROM president WHERE state="NY";
+-----------+-------------+
| last_name | first_name  |
+-----------+-------------+
| Van Buren | Martin      |
| Fillmore  | Millard     |
| Roosevelt | Theodore    |
| Roosevelt | Franklin D. |
+-----------+-------------+
4 rows in set (0.01 sec)
```

More on SELECT (cont.)

16) The following entry SELECT will return the last name and
birth date of presidents who are still alive

Note: The comparison operator will not work in this case:

```
mysql> SELECT * FROM president WHERE death = NULL;
Empty set (0.00 sec)

mysql> SELECT last_name, birth FROM president WHERE death is NULL;
+-----------+------------+
| last_name | birth      |
+-----------+------------+
| Ford      | 1913-07-14 |
| Carter    | 1924-10-01 |
| Reagan    | 1911-02-06 |
| Bush      | 1924-06-12 |
| Clinton   | 1946-08-19 |
| Bush      | 1946-07-06 |
+-----------+------------+
6 rows in set (0.00 sec)
```

More on SELECT (cont.)

17) This command will select the presidents who were born in the 18th century

```
mysql> SELECT last_name, birth FROM president WHERE birth<"1800-01-01";
+------------+------------+
| last_name  | birth      |
+------------+------------+
| Washington | 1732-02-12 |
| Adams      | 1735-10-30 |
| Jefferson  | 1735-04-13 |
| Madison    | 1751-03-16 |
| Monroe     | 1758-04-28 |
| Adams      | 1767-07-11 |
| Jackson    | 1767-03-15 |
| Van Buren  | 1782-12-05 |
| Harrison   | 1773-02-09 |
| Tyler      | 1790-03-29 |
| Polk       | 1795-11-02 |
| Taylor     | 1784-11-24 |
| Buchanan   | 1791-04-23 |
+------------+------------+
13 rows in set (0.00 sec)
```

More on SELECT (cont.)

18) The following command will select the president who was born first

```
mysql> SELECT last_name, birth from president ORDER BY birth ASC LIMIT 1;
+------------+------------+
| last_name  | birth      |
+------------+------------+
| Washington | 1732-02-12 |
+------------+------------+
1 row in set (0.00 sec)
```

More on SELECT (cont.)

19) The following query will return the names of fist 5 states (in descending order) in which the greatest number of presidents have been born

```
mysql> SELECT state, count(*) AS times FROM president GROUP BY state
    -> ORDER BY times DESC LIMIT 5;
+-------+-------+
| state | times |
+-------+-------+
| VA    |     8 |
| OH    |     7 |
| MA    |     4 |
| NY    |     4 |
| NC    |     2 |
+-------+-------+
5 rows in set (0.00 sec)
```

More on SELECT (cont.)

20) The following query will select presidents who have been born in the last 60 years

```
mysql> SELECT * FROM president WHERE(YEAR(now())- YEAR(birth)) < 60;
+-----------+------------+-------+-----------+------------+-------+
| last_name | first_name | state | city      | birth      | death |
+-----------+------------+-------+-----------+------------+-------+
| Clinton   | Bill       | AR    | Hope      | 1946-08-19 | NULL  |
| Bush      | George W.  | CT    | New Haven | 1946-07-06 | NULL  |
+-----------+------------+-------+-----------+------------+-------+
2 rows in set (0.00 sec)
```

Useful function to retrieve parts of dates are: **YEAR(), MONTH(), DAYOFMONTH(), TO_DAY().**

More on SELECT (cont.)

21) The following query will sort presidents who have died by their age and list the first 10 in descending order.

```
mysql> SELECT last_name, birth, death, FLOOR((TO_DAYS(death) - TO_DAYS(birth))/365) AS age
    -> FROM president
    -> WHERE death is not NULL ORDER BY age DESC LIMIT 10;
+------------+------------+------------+------+
| last_name  | birth      | death      | age  |
+------------+------------+------------+------+
| Jefferson  | 1735-04-13 | 1826-07-04 |   91 |
| Adams      | 1735-10-30 | 1826-07-04 |   90 |
| Hoover     | 1874-08-10 | 1964-10-20 |   90 |
| Truman     | 1884-05-08 | 1972-12-26 |   88 |
| Madison    | 1751-03-16 | 1836-06-28 |   85 |
| Nixon      | 1913-01-09 | 1994-04-22 |   81 |
| Adams      | 1767-07-11 | 1848-02-23 |   80 |
| Van Buren  | 1782-12-05 | 1862-07-24 |   79 |
| Jackson    | 1767-03-15 | 1845-06-08 |   78 |
| Eisenhower | 1890-10-14 | 1969-03-28 |   78 |
+------------+------------+------------+------+
```

Working with Multiple Tables

22) Often it is useful to separate data in conceptually distinct groups and store them in separate tables. Assuming we have a table that contains students' personal information, and we have another table that contains test scores of students. We can create a common field in each table, say "ssn" and work with the two tables together as follows:

```
SELECT last_name, address, test_date, score
FROM test, student
WHERE test.ssn = student.ssn;
```

For further examples, tutorials, and syntax visit:

http://www.mysql.com/documentation/index.html

http://www.mysql.com/documentation/mysql/bychapter/manual_Introduction.html#General-SQL