



MySQL Reading

Introduction

In order to store data for your website, you're going to want to use a database which is a structured collection of data. In particular, we're going to teach you about the relational database management system, MySQL. MySQL abstracts away the gritty details of the database and allows you to interact with the data at a higher level using the SQL programming language. You'll be able to insert, retrieve, delete, and alter data while MySQL will take care of maintaining the underlying data structures and executing queries quickly and reliably.

Overview

To get started with MySQL, you need to understand how the data is organized (at a high level) and the basic commands you can use to interact with this data.

A database is just a set of tables of data. For your website, you'll want to create a database. Let's call that database 6470db. In this database, you'll create a handful of tables. While the number and names of tables depends on your specific site, you will all want a users table which might look something like this:

user_id	username	name	school
1	mark	Mark Zhang	MIT
2	charles	Charles Liu	MIT
3	victor	Victor Costan	MIT
4	jharvard	John Harvard	Harvard

This is the basic idea. Whenever you want to store something else about a user (e.g. gender, birthdate, address, etc), you can just add a column to this table. Similarly, you can create new tables to save other information your website needs (e.g. questions, comments, etc).

Things get a lot more complicated when we start having to worry about the relationships between tables. In fact, this is at the core of MySQL (remember, it's a *relational* database management system). To give a quick example of this, let's look at what a comment table might look like:

comment_id	user_id	comment	date_posted
1	1	Happy new year!	2013-01-01 00:00:01
2	2	woooo!	2013-01-01 02:35:22
3	3	Ready for 6.470...	2013-01-05 19:49:37
4	2	woooooo!	2013-01-11 11:00:01

The important thing to notice here is that the `user_id` references a column in the `users` table. So when we want to go and display all the comments on our website along with usernames, we're going to need information from both tables. MySQL gives us some powerful tools for dealing with the relationships between tables, in particular JOIN's.

A JOIN allows you to combine two tables on a column (in this case `user_id`) so that you can get all the information you need at once. JOIN's can get complicated fast and can be avoided for the most part. You can do just everything you'll need for 6.470 with the basic SELECT, INSERT, UPDATE, and DELETE statements.

SQL References

http://www.w3schools.com/sql/sql_intro.asp - This tutorial gives you a quick crash course on the basics of what you need to know. Nothing too fancy, but it gets the job done and it gets you up to speed quickly.

<https://www.coursera.org/db> - Coursera has an online database class with video lectures and exercises that allow you to test your solutions online with realtime feedback. We definitely recommend this. There's a lot of extra material in the course that isn't fundamental to getting a website up though so you may end up learning more than you need (and thus not having as much time to focus on other areas of your site).

<http://dev.mysql.com/doc/index.html> - We have to mention the official MySQL documentation. It's thorough but rather user unfriendly. We don't really suggest this for beginners but you should at least know about it.

Prior to the Workshop

Getting setup on `scripts.mit.edu`:

1. Follow their instructions: <http://scripts.mit.edu/faq/66/how-signup-locker-for-mysql-service>
 - a. Sign into Athena via ssh.
 - b. Run these commands:

```
athena$ add scripts
athena$ signup-sql
```

- c. You can now find your password in `/mit/[your Athena name]/.sql/my.cnf` So this command would print it out:

```
cat /mit/[your Athena name]/.sql/my.cnf
```

2. Go to `sql.mit.edu` and login.
3. Add a new database (let's call it `6470test`).
4. Now you can go to <https://sql.scripts.mit.edu/phpMyAdmin/> and use `phpMyAdmin` to play around with your database.

Topics to read about before the workshop

To facilitate solving of workshop exercises, here are some essential topics you should know, as well as some optional topics that you should learn throughout the month.

We *highly* suggest the video lectures on <https://www.coursera.org/db> for watching before workshop. The videos are clear and concise. These are the ones we want you to watch (or read up on using `w3schools`) before workshop:

- Relational Databases
 - The relational model
 - Querying relational databases
- SQL
 - Introduction to SQL
 - Basic SELECT statement
 - Table variables and set operators
 - [Optional] Subqueries in WHERE clause
 - [Optional] Subqueries in FROM and SELECT
 - The JOIN family of operators
 - Aggregation
 - NULL values
 - Data modification statements