Department of
Computer Science and Engineering

Title: Introduction to Database and MySQL

Databse System Lab

CSE 210

Green University of Bangladesh

1 Objective(s)

- To install MySQL Database Server.
- To introduce Data Types used in Database System.

2 Problem analysis

Database is a key course of computer science. At the beginner level, most of the students are not familiar with how to use database in computer system. This section helps you get started with a very common database system named MySQL. We will start installing MySQL, and creating a database in the MySQL server for practicing.

Figure 1: Logo of MySQL

3 Procedure

If you want to install MySQL on Windows environment, using MySQL installer is the easiest way. MySQL installer provides you with an easy-to-use wizard that helps you to install MySQL with the following components:

- MySQL Server
- All Available Connectors
- MySQL Workbench with Sample Data Models
- MySQL Notifier
- Tools for Excel and Microsoft Visual Studio
- MySQL Sample Databases

To download MySQL installer, following link

https://www.apachefriends.org/download.html

Required toll are available for all operating systems in the above link.

After installation of the downloaded XAMPP, you have to run the XAMPP control panel system. It will be like the Figure 2.
To start the service, you have to press the **Start** button of the **Apache** and **MySQL** module. Now, the system is ready to work with.

4 Practicing With XAMPP

To start the practice session, you have to press the **Admin** button of MySQL module or you have to use the following link in your web browser: http://localhost/phpmyadmin/

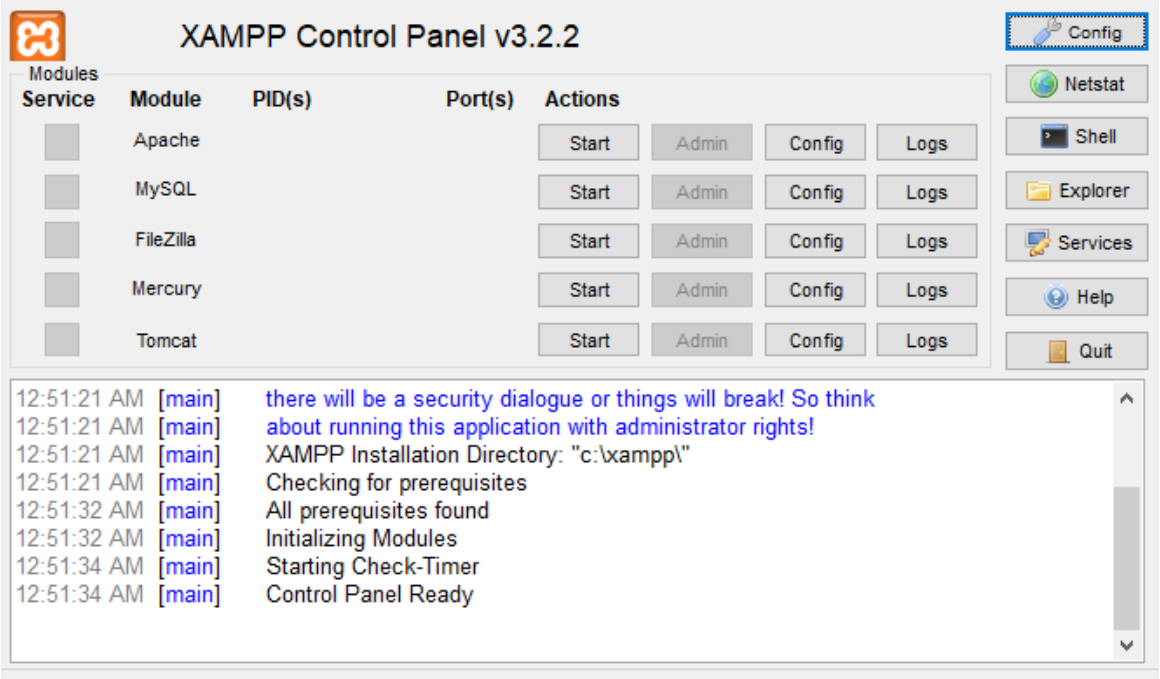

Figure 2: XAMPP Control Panel

Table 1: Basic Data Types

Data Type	Description
CHAR(size)	Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis. Can store up to 255 characters
VARCHAR(size)	Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters. Note: If you put a greater value than 255 it will be converted to a TEXT type
TINYTEXT	Holds a string with a maximum length of 255 characters
TEXT	Holds a string with a maximum length of 65,535 characters
TINYINT(size)	-128 to 127 normal. 0 to 255 UNSIGNED*. The maximum number of digits may be specified in parenthesis
SMALLINT(size)	-32768 to 32767 normal. 0 to 65535 UNSIGNED*. The maximum number of digits may be specified in parenthesis
MEDIUMINT(size)	-8388608 to 8388607 normal. 0 to 16777215 UNSIGNED*. The maximum number of digits may be specified in parenthesis
INT(size)	-2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED*. The maximum number of digits may be specified in parenthesis
BIGINT(size)	-9223372036854775808 to 9223372036854775807 normal. 0 to 18446744073709551615 UNSIGNED*. The maximum number of digits may be specified in parenthesis
FLOAT(size,d)	A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DOUBLE(size,d)	A large number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DECIMAL(size,d)	A DOUBLE stored as a string, allowing for a fixed decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DATE()	A date. Format: YYYY-MM-DD Note: The supported range is from '1000-01-01' to '9999-12-31'
DATETIME()	*A date and time combination. Format: YYYY-MM-DD HH:MI:SS Note: The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'
TIMESTAMP()	*A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD HH:MI:SS Note: The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC
TIME()	A time. Format: HH:MI:SS Note: The supported range is from '-838:59:59' to '838:59:59'
YEAR()	A year in two-digit or four-digit format. Note: Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069

You will get a web page like Figure 3 in your tab. Now, to create a database, you have to select the **New** option in phpMyAdmin panel (Upper most option in the left side of the web page). Then, you have to input a name in the **Database name** field as your wish and press the **Create** button. After that, to input data in your required structure, you have to create table with table name and required number of column. For this, you have to open **SQL** option and write necessary syntax.

5 Input/Output

For example, to create a table named Student with 4 attributes named StudentID, Name, Address, and Contact.

```
CREATE TABLE Student( StudentID varchar(9), Name varchar(20), Address varchar(50), Contact varchar(11))
```

Created table will be like Figure 4.

6 Discussion & Conclusion

In this example, I have used **varchar** as data type for all the attributes. Numbers indicates the length of that specific attribute. Besides varchar, there many types of data type which is described in the Table 1.

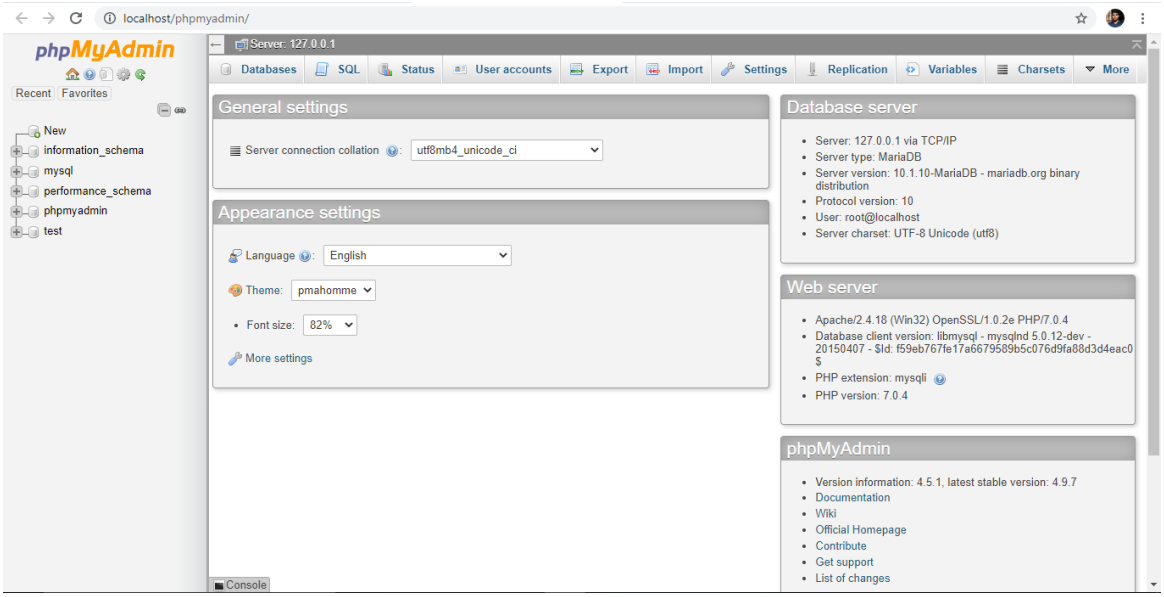

Figure 3: Session in Localhost

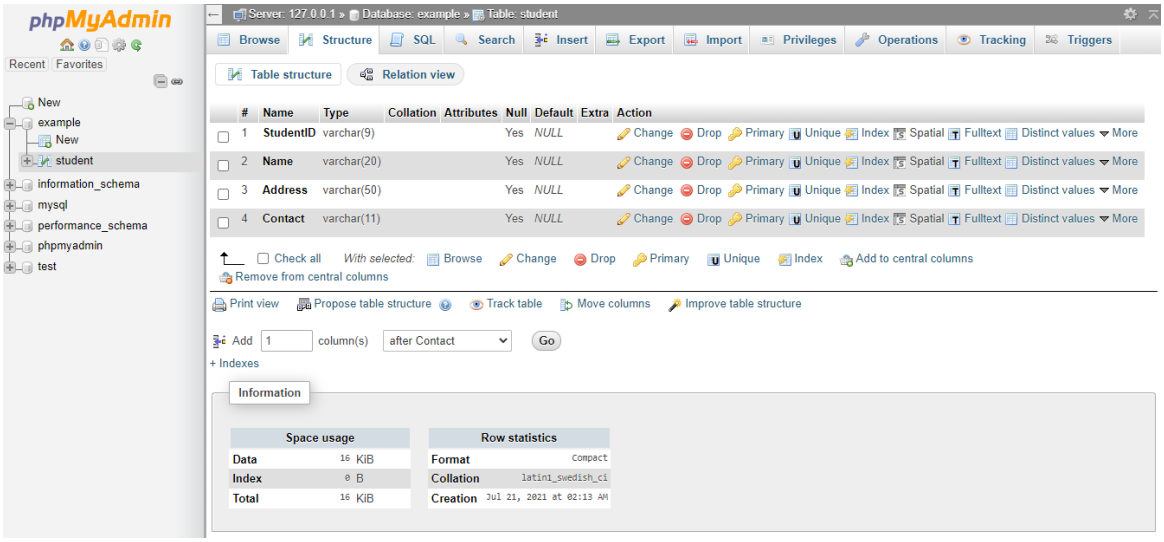

Figure 4: Created Table named Student

We have installed a MySQL in our computer, created a database with a table. We are also known to basic data types used in MySQL. That's meant, we have achieved the lab objectives.

7 Lab Task (Please implement yourself and show the output to the instructor)

1. Create a database named "University"
2. Create a Table named "Teacher" with attributes named TeacherID, Name, Designation, Address, and Email.
3. Create a Table named "Student" with attributes named StudentID, Name, Address, and Phone.
4. Create a Table named "Staff" with attributes named StaffID, Name, Position, Address, and Phone.

7.1 Problem analysis

1. You have to create a database named "University" with the help of create option provided in pypMyAdmin panel.
2. You have to create a table named "Teacher" with attributes named TeacherID, Name, Designation, Address, and Email. You must use proper data type and size.
3. You have to create a table named "Student" with attributes named StudentID, Name, Address, and Phone. You must use proper data type and size.
4. You have to create a table named "Staff" with attributes named StaffID, Name, Position, Address, and Phone. You must use proper data type and size.

8 Lab Exercise (Submit as a report)

- Create a Database with five or six tables.
- Use all the basic database described in table 1.

9 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.