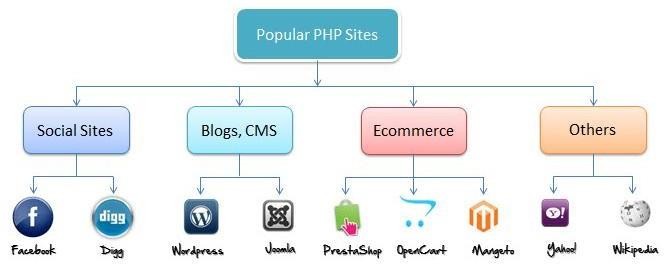
**Lab Session 06**

**Lab Session #06 (*Connectivity: PHP with MySQL*)**

**Lab Session 06**

**PHP:**

PHP is the most popular scripting language (i-e A scripting language is a language that interprets scripts at runtime) for web development. It is free, open source and server-side (the code is executed on the server).



**MySQL:**

MySQL is a Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). It is also free and open source. The combination of PHP and MySQL gives unmet options to create just about any kind of website - from small contact form to large corporate portal.

**What is required?**

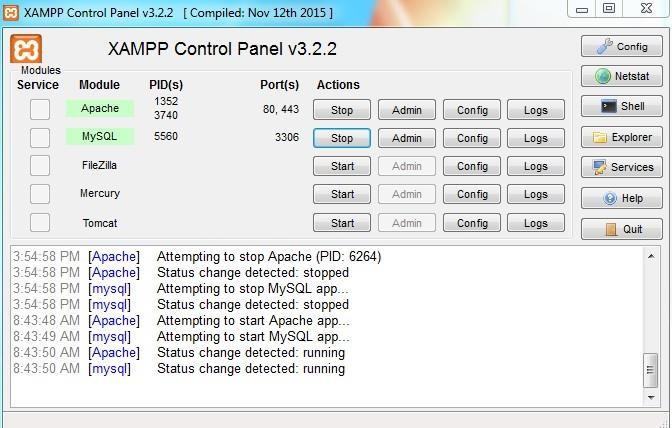
1. Text Editor (i.e., Notepad++, Brackets, Dreamweaver etc.)

2. Wamp or Xampp Server

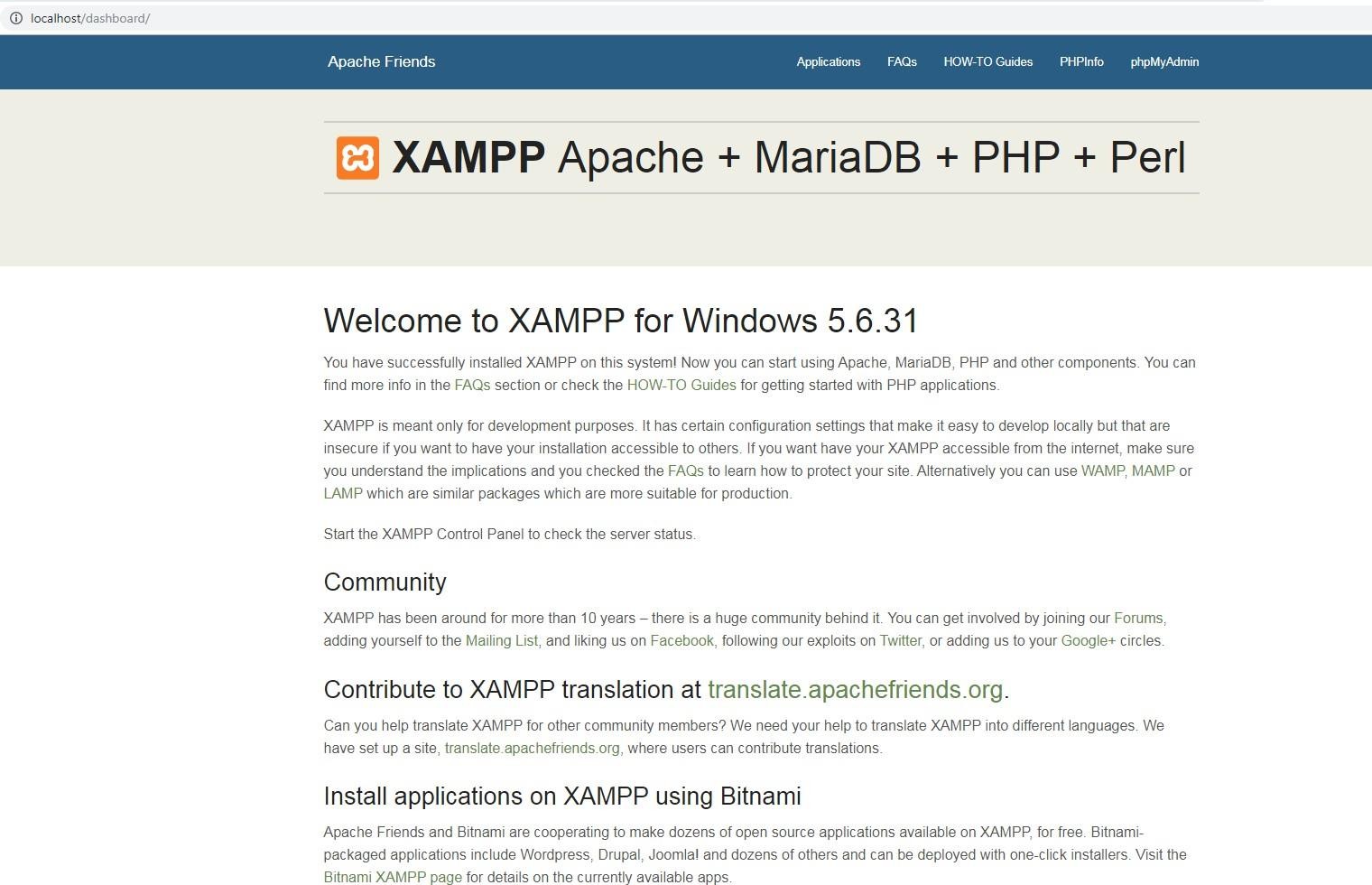
3. Web browser (Firefox, Chrome etc.)

**Lab Session 06**

**Xampp\_Control Panel:**

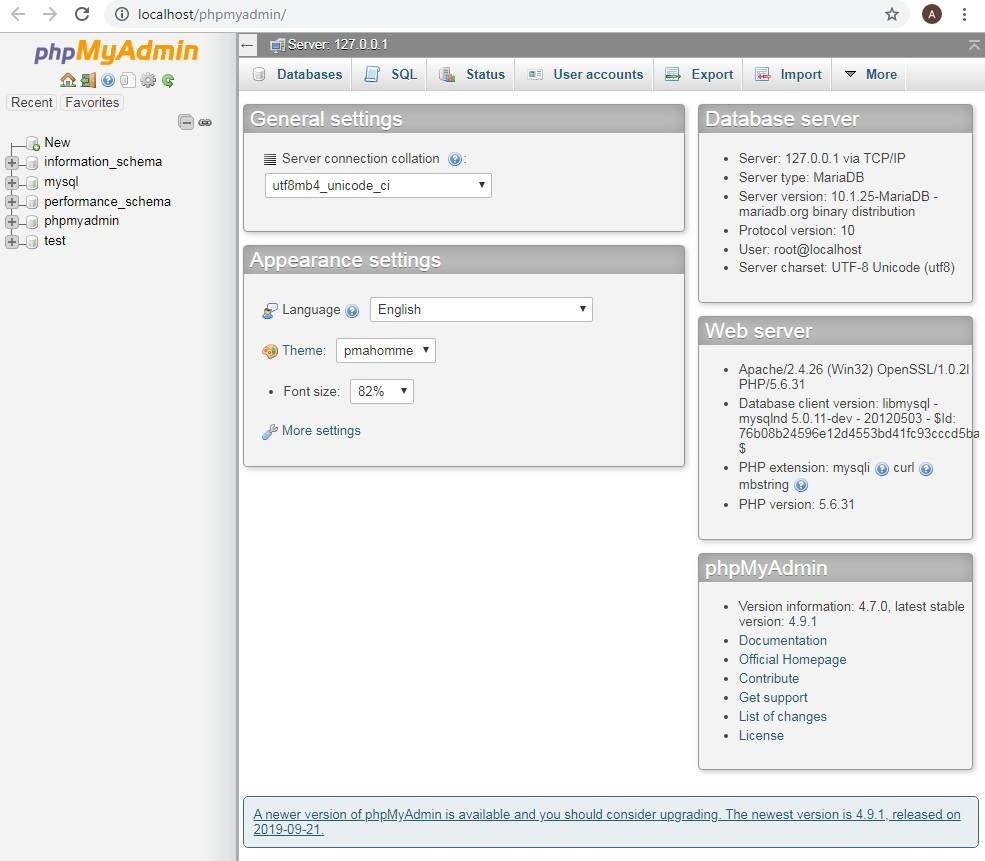


**Apache \_Admin Console:**



**Lab Session 06**

**MySQL-phpMyAdmin Panel:**



**Creating Users:**

Follow these Steps:

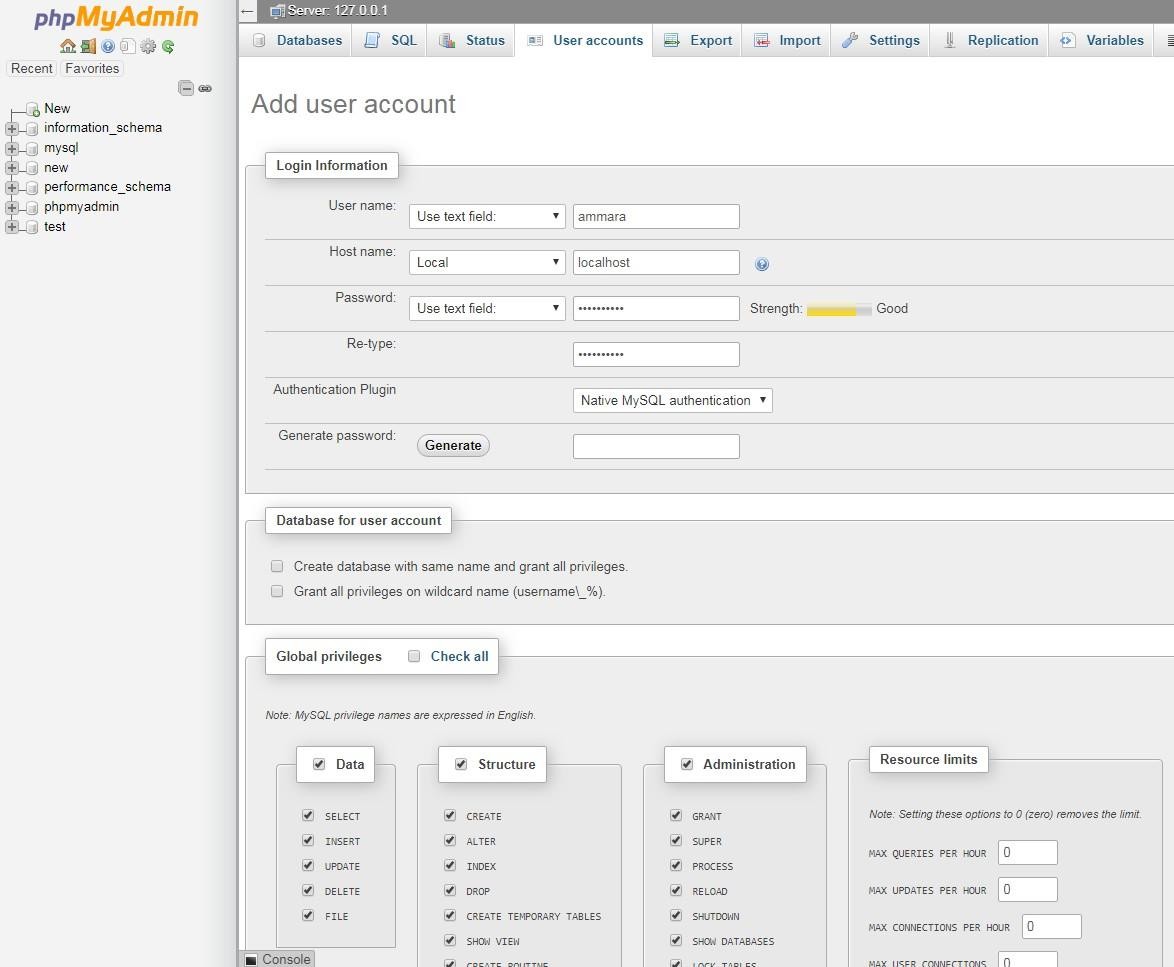
1. Go to User Accounts in phpMyAdmin.

2. Click Add user account.

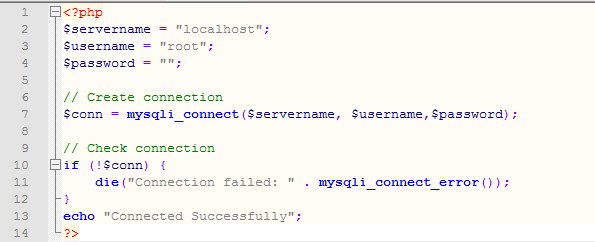
3. Set username, hostname, password and privileges for this user.

4. Click Go.

**Lab Session 06**



**PHP Connection to MySQL:**



**Lab Session 06**

Save this file in C:\xampp\htdocs folder.

To run the script type in the browser localhost/filename.php

**Creating Database:**

Two ways

1. Directly on phpMyAdmin

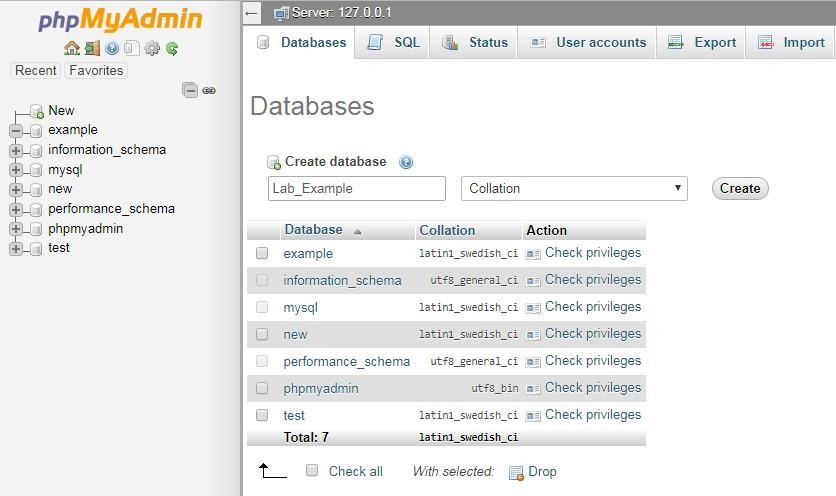
2. Front end (Text Editor)

**1. Direct on phpMyAdmin:**

Follow these steps:

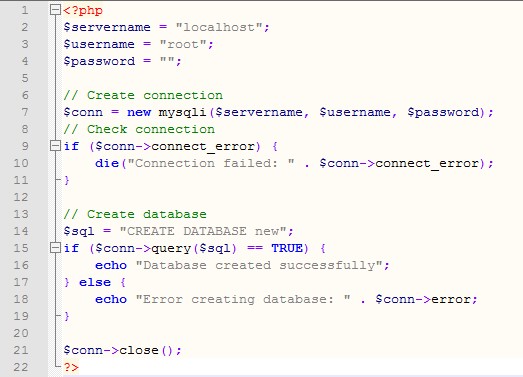
1. Go to Databases in phpMyAdmin.

2. Fill the Database name field then Click Create.



**Lab Session 06**

**2. Front End (Text Editor)**



**Creating Tables:**

Two ways

1. Directly on phpMyAdmin

2. Front end (Text Editor)

**1. Direct on phpMyAdmin:**

Follow these steps:

1. Select the database in which you want to create table.

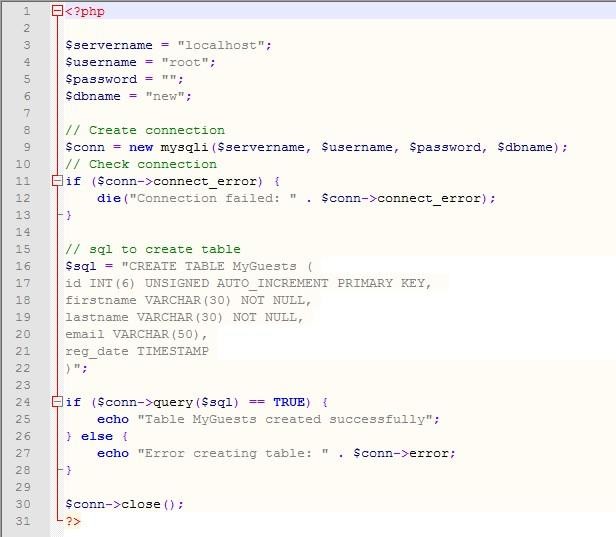
2. Fill out the table name and quantity of fields then click Go.

3. Give every field a proper name ,data type, size and Constraint (if any).

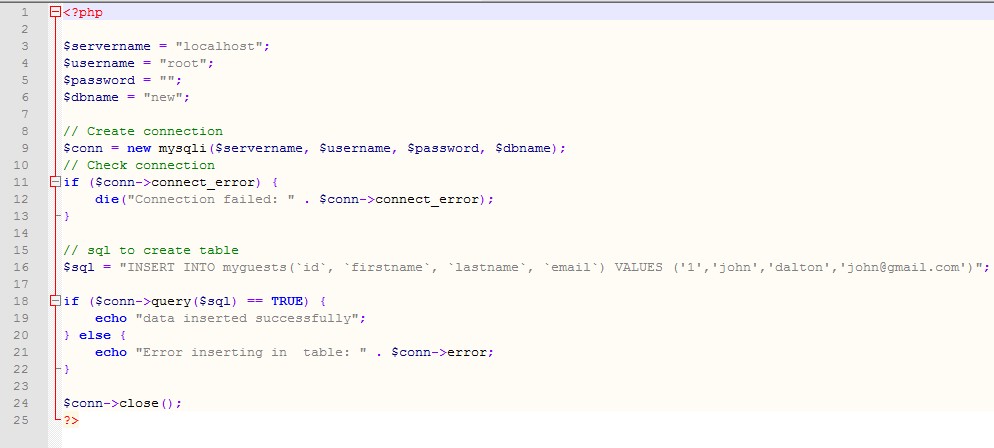
4. Click Go.

**Lab Session 06**

**2.Front end (Text Editor)**

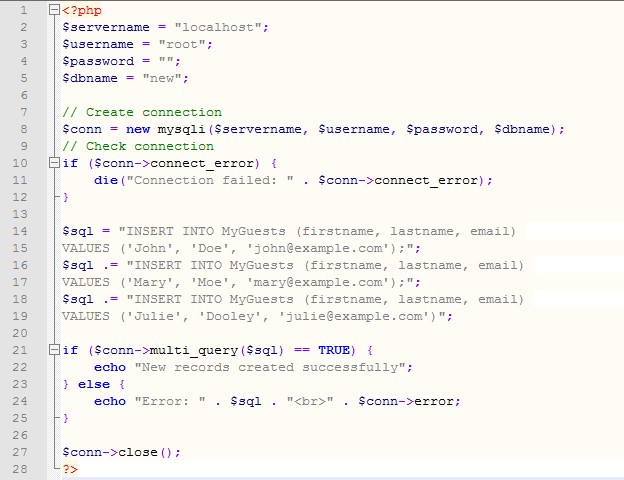


**Inserting Data into Table:**



**Lab Session 06**

**Inserting multiple records:**



**Prepared Statements and Bound Parameters:**

• A prepared statement is a feature used to execute the same (or similar) SQL statements repeatedly with high efficiency.

• Prepared statements basically work like this:

 Prepare: An SQL statement template is created and sent to the database. Certain values are left unspecified, called parameters (labeled "?"). Example: INSERT INTO MyGuests VALUES(?, ?, ?)

 The database parses, compiles, and performs query optimization on the SQL

statement template, and stores the result without executing it

 Execute: At a later time, the application binds the values to the parameters, and

the database executes the statement. The application may execute the statement as many times as it wants with different values

**Lab Session 06**

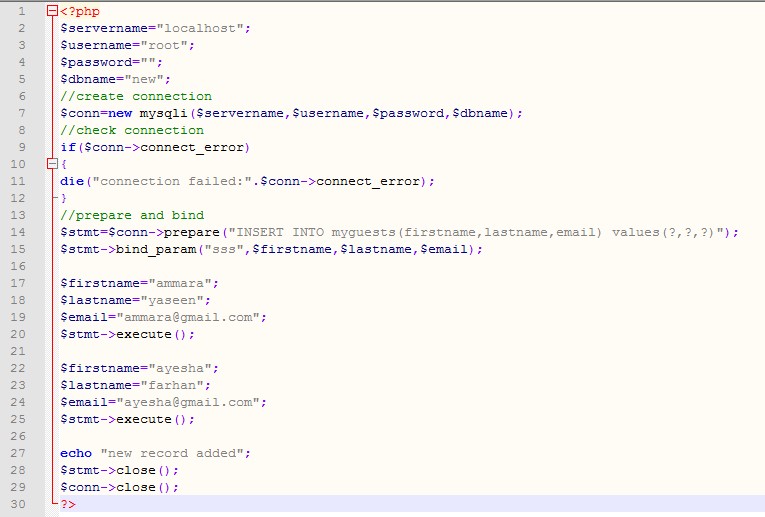
**Advantages:**

Compared to executing SQL statements directly, prepared statements have three main advantages:

• Prepared statements reduces parsing time as the preparation on the query is done only once (although the statement is executed multiple times)

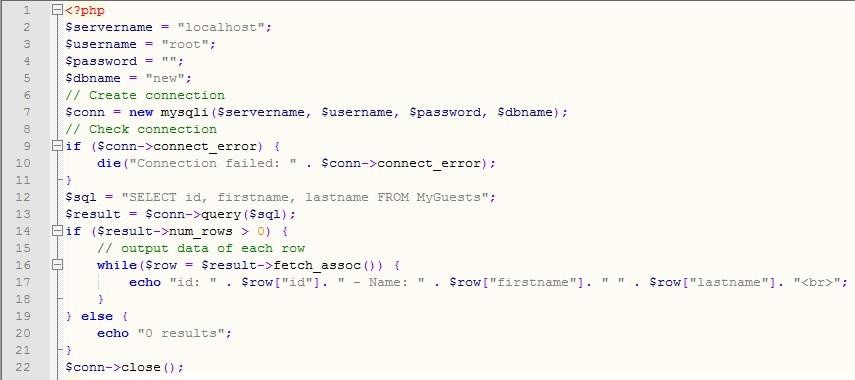
• Bound parameters minimize bandwidth to the server as you need send only the parameters each time, and not the whole query

• Prepared statements are very useful against SQL injections, because parameter values, which are transmitted later using a different protocol, need not be correctly escaped. If the original statement template is not derived from external input, SQL injection cannot occur.

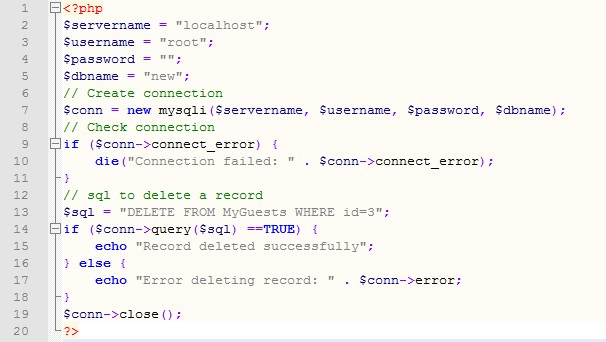


**Lab Session 06**

**Selecting Data:**



**Deleting Data:**



**Lab Session 06**

**Updating Data:**

