In this lab you will write a simple web application with form field validation using PHP.

Step-by-step instructions

1. Start VMWare Workstation.
2. Start you Ubuntu VM and login.
3. Use one of the web browsers on the VM to download lab5.sql from the Lab 5 drop box on FOL to your VM. You can download the file to the directory of your choice.
4. Open a terminal window and change directory to the directory where you put the lab5.sql file.
5. In the terminal window start the MySQL CLI and sign in as **lamp1user** with a password of **!Lamp1!** (if you did not complete Lab 3, see the instructions for lab 3 to create the necessary database and user).
6. Change to the demo database, the command is **USE demo;**
7. Use the command **source lab5.sql** to create a table called 'lab5' in the demo database. You are to use this table to store the data input in you web application. The source command is used to run SQL script files.

   The structure of the lab5 table is:

   ```
   mysql> show columns from lab5;
   +----------------+--------------+------+-----+---------+----------------+
   | Field          | Type         | Null | Key | Default | Extra          |
   +----------------+--------------+------+-----+---------+----------------+
   | id             | int(11)      | NO   | PRI | NULL    | auto_increment |
   | first_name     | varchar(50)  | NO   |     | NULL    |                |
   | last_name      | varchar(50)  | NO   |     | NULL    |                |
   | email          | varchar(128) | NO   |     | NULL    |                |
   | email_personal | int(2)       | NO   |     | NULL    |                |
   | phone          | varchar(20)  | NO   |     | NULL    |                |
   | phone_personal | int(2)       | NO   |     | NULL    |                |
   +----------------+--------------+------+-----+---------+----------------+
   7 rows in set (0.00 sec)
   ```

8. [10 marks] Write a simple PHP web-based contacts application that uses a single form to to accept data for all of the fields listed above except the 'id' field. The form and database manipulation code are to be placed in a single PHP source file. Database operations are to be performed only when the form is submitted using the POST method. The table below give a description of each of the fields and gives the input element type that is to be used on the HTML form. Use the appropriate PHP superglobal to retrieve the submitted data.

Field	Description	Input type
first_name	The first name of the contact	Text
last_name	The last name of the contact	Text
email	The email address of the contact	Text
email_personal	An indicator whether the contact’s email is a personal email address or not. The field is to contain 1 if the email address is personal, otherwise it is to contain a 0.	Checkbox
phone	The phone number for the contact	Text
phone_personal	An indicator whether the contact’s phone number is a personal one or not. The field is to contain 1 if the phone number is personal, otherwise it is to contain a 0.	Checkbox

Remember that the id field has the auto_increment property, your SQL insert statements must take advantage of this.

9. Thoroughly test the web application created in step 8. Make sure it works before you proceed to step 10.
10. [10 marks] Add form validation to your PHP code that performs the tests listed below. Not that you must do this validation in PHP, NO JavaScript allowed.
    a. Checks that the first_name field is not empty and does not exceed the length of the field in the database.
    b. Check that the last_name is not empty and does not exceed the length of the field in the database.
    c. Check that the email field is not empty and does not exceed the length of the field in the database.
    d. Check that the phone field is not empty and does not exceed the length of the field in the database.

    If validation fails on any field, the form is to be re-displayed containing all of the data that was entered originally, along with messages indicating which field(s) have problems and a description of the problems.
11. Test the data validation added in step 10. Try submitting empty fields and fields containing more data than the database field will allow.
12. [5 marks] Add functionality to the application that protects the application from SQL injection attacks via the 4 text input fields. You may use any of the techniques discussed in class.
13. Test you application to make sure that the injection attack prevention is applied. You DO NOT need to try to perform an injection attack.
14. Upload you PHP source file to the drop box on FOL
15. Shutdown you VM
16. Exit VMWare Workstation.