

## Lab 6 – Using MySQL with PHP

### 1. Creating Database and Tables

- Open phpMyAdmin. You can either click from WampServer icon, or go to <http://localhost/phpmyadmin/index.php>. Enter username : root, password :
- Click new to create a new database.

The screenshot shows the phpMyAdmin interface with the following details:

- General settings:** Includes options for changing password, server connection collation (set to utf8mb4\_unicode\_ci), and appearance settings (language set to English, theme set to pmahomme, font size set to 82%).
- Database server:** Displays information about the MySQL server, including its version (5.7.19), protocol version (10), and PHP version (5.6.31).
- Web server:** Displays information about the Apache and PHP configurations.
- Sidebar:** Shows recent databases like MySQL, information\_schema, mysql, performance\_schema, and sys. A red box highlights the "New" button.

- On the **Create database**, insert **Registration**.
- Then, create table as follows.

The screenshot shows the phpMyAdmin interface with the following details:

- Database:** Registration
- Structure:** Tab selected.
- No tables found in database.** Message displayed.
- Create table:** Dialog box open with the following fields:
  - Name: User
  - Number of columns: 6
  - Go button

- e) Create table **User** as follows. Then, click **Save**. Please note the **A\_I** (Auto Increment) and **Storage Engine**.

The screenshot shows the MySQL Workbench interface for creating a new table named "User".

**Table Structure:**

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.
user_id	INT	5	None			PRIMARY		<input checked="" type="checkbox"/>
first_name	VARCHAR	20	None			---		<input type="checkbox"/>
last_name	VARCHAR	20	None			---		<input type="checkbox"/>
email	VARCHAR	20	None			---		<input type="checkbox"/>
pass_word	VARCHAR	20	None			---		<input type="checkbox"/>
registration_date	DATETIME		None			---		<input type="checkbox"/>

**Table comments:** (empty)

**Collation:** (empty)

**Storage Engine:** InnoDB (highlighted with a red box)

**PARTITION definition:** (empty)

**Partition by:** (empty)

**Partitions:** (empty)

Buttons at the bottom: Preview SQL, Save (highlighted with a red box)

- f) Your table will look like this.

Table	Action	Rows	Type	Collation	Size	Overhead
user	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	1	InnoDB	latin1_swedish_ci	16 Kib	-
Sum			MyISAM	latin1_swedish_ci	16 Kib	0 B

## 2. Inserting Data into Table

- a) Click SQL tab.

```
INSERT INTO user (first_name, last_name, email, pass_word, registration_date)
VALUES ('John','Hacker','john@gmail.com','abc123',NOW())
*make sure you rewrite ' on the textarea, else you will get error
```

- b) To insert a new row of data into **user** table, enter the following command in the Run SQL Query area:

```
INSERT INTO user (first_name, last_name, email, pass_word, registration_date)
VALUES ('John','Hacker','john@gmail.com','abc123',NOW())
*make sure you rewrite ' on the textarea, else you will get error
```

- c) Successful inserted data will look like this.

Show query box

✓ 1 row inserted.  
Inserted row id: 1 (Query took 0.1460 seconds.)

```
INSERT INTO user (first_name, last_name, email, pass_word, registration_date) VALUES
('John','Hacker','john@gmail.com','abc123',NOW())
```

[Edit inline] [ Edit ] [ Create PHP code ]

- d) To insert a new row of data into **user** table, enter the following command:

```
INSERT INTO USER (first_name, last_name, email, pass_word, registration_date)
VALUES ('Amy','Scammer','amy@gmail.com','123456',NOW()),
('Mary','Phisher','mary@gmail.com','456789',NOW())
```

- e) Click **Browse** to view all inserted data.

	user_id	first_name	last_name	email	pass_word	registration_date
<input type="checkbox"/>	1	John	Hacker	john@gmail.com	abc123	2019-04-08 13:02:33
<input type="checkbox"/>	2	Amy	Scammer	amy@gmail.com	123456	2019-04-08 13:06:24
<input type="checkbox"/>	3	Mary	Phisher	mary@gmail.com	456789	2019-04-08 13:06:24

- f) You may use the SQL tab or console (newer version of phpMyAdmin) to run your insert SQL. Now, insert 7 more records by using the **Insert** tab. Note: Please leave empty for the user\_id as this column is auto generated.

### 3. Retrieving Data

- a) Use your SQL tab or console to run these SQL.

	user_id	first_name	last_name	email
<input type="checkbox"/>	1	John	Hacker	john@gmail.com
<input type="checkbox"/>	2	Amy	Scammer	amy@gmail.com
<input type="checkbox"/>	3	Mary	Phisher	mary@gmail.com

- a) To retrieve all the data from the **user** table, enter the following command:

**SELECT \* FROM user**

- b) To retrieve only the first and last names from **user** table, enter the following command:

**SELECT first\_name, last\_name FROM user**

- c) Using conditionals:

- (i) To select all the first names of users whose last name is ‘Phisher’, enter the command:

**SELECT \* FROM user WHERE last\_name =’Phisher’**

- (ii) To select all data from every record in the **user** table that does not have email address, enter the following command:

**SELECT \* FROM user WHERE email is NULL**

- d) Using the LIKE keyword:

- (i) To select all of the records in which the last name starts with ‘Hacker’, enter the following command:

**SELECT \* FROM USER WHERE last\_name LIKE 'Hacker%'**

- (ii) To select the name for every record whose email address is not of the form ‘%.com’, enter the following command:

**SELECT first\_name, last\_name FROM USER WHERE email NOT LIKE '%.com'**

- e) To sort data:

- (i) Enter the following command to select all of the users in alphabetical order by **last\_name**:

**SELECT first\_name, last\_name FROM user ORDER BY last\_name ASC**

- (ii) Enter the following command to display all of the users in alphabetical order by last name then first name:

```
SELECT first_name, last_name FROM user ORDER BY last_name ASC, first_name
```

- (iii) Enter the following command to show all of the users by date registered, in descending order:

```
SELECT * FROM user ORDER BY registration_date DESC
```

- f) To limit the amount of data returned:

- (i) Enter the following command to select the last five registered users:

```
SELECT * FROM user ORDER BY registration_date DESC LIMIT 5
```

- (ii) Enter the following command to select the second person to register, enter the following command:

```
SELECT * FROM user ORDER BY registration_date ASC LIMIT 1,1
```

#### 4. Updating Data

- a) To determine which record will be updated, enter the following command:

```
SELECT user_id FROM USER WHERE first_name='Mary'  
AND last_name='Scammer'
```

- b) To update the record, enter the following command:

```
UPDATE USER SET email='amy@gmail.com' WHERE user_id=2
```

- c) To confirm that the change has been made, enter the following command:

```
SELECT * FROM user WHERE user_id=2
```

#### 5. Deleting Data

- d) To determine which record will be deleted, enter the following command:

```
SELECT user_id FROM USER WHERE first_name='Mary'  
AND last_name='Phisher'
```

- b) To preview what will happen when the delete is made, enter the following command:

**SELECT \* FROM user WHERE user\_id=3**

- c) To delete the record, enter the following command:

**DELETE FROM user WHERE user\_id=3**

- d) To confirm that the deletion has been made, enter the following command:

**SELECT \* FROM user ORDER BY user\_id ASC**