

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

----- ❦ ★ ❦ -----



BÁO CÁO KẾT QUẢ THỰC HÀNH BẢO MẬT WEB VÀ ỨNG DỤNG

Lab 01: ÔN TẬP KIẾN THỨC CƠ BẢN ỨNG DỤNG WEB (HTML, JAVASCRIPT, PHP, CSDL)

Giảng viên giảng dạy: Nghi Hoàng Khoa

Nhóm sinh viên thực hiện:

- | | |
|-----------------------|----------|
| 1. Phạm Khôi Nguyên | 18520114 |
| 2. Phan Thanh Hải | 18520705 |
| 3. Nguyễn Lý Đình Nhì | 18521205 |

TP. HỒ CHÍ MINH, 03/2021

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

----- ❦ ★ ❦ -----



BÁO CÁO KẾT QUẢ THỰC HÀNH BẢO MẬT WEB VÀ ỨNG DỤNG

Lab 01: ÔN TẬP KIẾN THỨC CƠ BẢN ỨNG DỤNG WEB (HTML, JAVASCRIPT, PHP, CSDL)

Giảng viên giảng dạy: Nghi Hoàng Khoa

Nhóm sinh viên thực hiện:

- | | |
|-----------------------|----------|
| 1. Phạm Khôi Nguyên | 18520114 |
| 2. Phan Thanh Hải | 18520705 |
| 3. Nguyễn Lý Đình Nhì | 18521205 |

TP. HỒ CHÍ MINH, 03/2021

MỤC LỤC

LỜI NÓI ĐẦU	1
BÀI THỰC HÀNH 1.....	2
1. Yêu cầu 1	2
2. Yêu cầu 2	3
3. Yêu cầu 3	4
BÀI THỰC HÀNH 2.....	5
BÀI THỰC HÀNH 3.....	6
BÀI THỰC HÀNH 4.....	8
BÀI THỰC HÀNH 5.....	11
BÀI THỰC HÀNH 6.....	14
BÀI THỰC HÀNH 7.....	16
1. Yêu cầu 1	16
2. Yêu cầu 2	16

LỜI NÓI ĐẦU

Đây là phần bài làm của nhóm cho bài tập thực hành buổi 01 về ôn tập kiến thức cơ bản ứng dụng web. Toàn bộ nội dung thực hành được triển khai trên hệ thống vlab máy ảo của trường.

Cảm ơn anh Nghi Hoàng Khoa trong thời gian 2 tuần vừa qua chịu khó trả lời những câu hỏi, những thắc mắc của nhóm trên mail và cả trên group Facebook.

BÀI THỰC HÀNH 1

Điều chỉnh tập tin mã nguồn SimpleForm.html để thỏa mãn các yêu cầu bên dưới.

Dùng thuộc tính mặc định của HTML, hiện thực các ràng buộc sau:

- Bắt buộc phải nhập **04** trường **Họ và tên**, **Mã số sinh viên**, **Email** và **Số điện thoại** trước khi lưu.
- Trường **Email** cần kiểm tra người dùng có nhập đúng định dạng email không.
- **Số điện thoại** và **Mã số sinh viên** có độ dài tối đa 15 ký tự.

1. Yêu cầu 1

Để bắt buộc nhập các trường **Họ và tên**, **Mã số sinh viên**, **Email** và **Số điện thoại**, ta thêm vào thuộc tính `required` vào thẻ `<input>` tương ứng với 4 trường trên trong tập tin SimpleForm.html.

```
<div class="info">
    <div class="label">HỌ và tên</div>
    <input type="text" name="name" required</input><br>
</div>
<div class="info">
    <div class="label">Mã số sinh viên</div>
    <input type="text" name="student_id" required</input><br>
</div>
<div class="info">
    <div class="label">Email</div>
    <input type="text" name="email" required</input><br>
</div>
<div class="info">
    <div class="label">Số điện thoại</div>
    <input type="text" name="phone" required</input><br>
</div>
```

Trên giao diện, khi người dùng không nhập thông tin vào những trường bắt buộc nhập thì sẽ thông báo như hình dưới đây:

Họ và tên

Mã số sinh viên

Email Please fill out this field.

Số điện thoại

Ngày sinh

Giới tính

Địa chỉ

Ghi chú

2. Yêu cầu 2

Vì trường **Email** chỉ dùng để nhập email nên ta sửa giá trị của thuộc tính `type="text"` thành `type="email"` trong thẻ `<input>` và bổ sung thêm điều kiện định dạng của mail thông qua thuộc tính `pattern` trong thẻ `<input>`. Thuộc tính `pattern` cho phép so khớp thông tin nhập vào với biểu thức chính quy (regular expression) trước khi lưu. Ta lựa chọn giá trị cho thuộc tính `pattern` theo định dạng của mail là: `pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}"` (tham khảo ở đường link <https://www.regular-expressions.info/email.html>).

```
<div class="info">
  <div class="label">Email</div>
  <input type="email" name="email" pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}" required></input><br>
</div>
```

Trên giao diện, khi người dùng nhập sai định dạng của mail thì sẽ thông báo như hình dưới đây:

Họ và tên	<input type="text" value="abc"/>
Mã số sinh viên	<input type="text" value="12345677"/>
Email	<input type="text" value="pth@abq"/>
Số điện thoại	<div>Please match the requested format.</div>
Ngày sinh	<input type="text"/>
Giới tính	<input type="text"/>
Địa chỉ	<input type="text"/>
Ghi chú	<input type="text"/>

3. Yêu cầu 3

Thuộc tính `maxlength` xác định độ dài tối đa cho trường nhập dữ liệu. Để **Số điện thoại** và **Mã số sinh viên** có độ dài tối đa 15 ký tự, ta thêm thuộc tính `maxlength` vào thẻ `<input>` và thiết lập giá trị cho nó: `maxlength="15"`.

```
<div class="info">
  <div class="label">Mã số sinh viên</div>
  <input type="text" name="student_id" maxlength="15" required></input><br>
</div>
<div class="info">
  <div class="label">Email</div>
  <input type="email" name="email" pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}" required></input><br>
</div>
<div class="info">
  <div class="label">Số điện thoại</div>
  <input type="text" name="phone" maxlength="15" required></input><br>
</div>
<div class="info">
  <div class="label">Ngày sinh</div>
  <input type="text" name="dob"></input><br>
</div>
```

Trên giao diện, nếu người dùng nhập hơn 15 ký tự cho trường **Số điện thoại** hoặc **Mã số sinh viên** thì những ký tự dư đó sẽ bị HTML bỏ qua.

BÀI THỰC HÀNH 2

Viết mã nguồn **Javascript** (có thể ở trong hoặc ngoài tập tin HTML) thực hiện kiểm tra trường **Họ và tên** chỉ cho nhập chữ và khoảng trắng.

Ta sẽ thêm sự kiện *keypress* trong thẻ `<input>` của trường **Họ và tên** để bỏ qua các kí tự không thỏa mãn điều kiện ở trên khi người dùng nhập vào:

- Trong thẻ `<input>`:

```
<div class="info">
  <div class="label">Họ và tên</div>
  <input type="text" name="name" required onkeypress="return checkValidName(event)"></input><br>
</div>
```

- Trong thẻ `<script>` (để trong tập tin HTML):

```
function checkValidName(_event) {
    return /^[A-Za-z ]+$/.test(_event.key);
}
```


BÀI THỰC HÀNH 3

Tạo một bảng trong cơ sở dữ liệu MySQL với các trường tương ứng ở **Bài thực hành 1** thỏa mãn các yêu cầu sau.

- Cơ sở dữ liệu có thể lưu chữ dạng **UTF-8**.
- Các trường tương ứng ở **Bài thực hành 1** với **Họ và tên, MSSV, Số điện thoại** và **Email** không được bỏ trống. **Số điện thoại** và **MSSV** có độ dài tối đa 15 ký tự.

Trước tiên, ta tiến hành truy cập MySQL bằng cách gõ lệnh sau: `sudo mysql -u root -p`.

Ta sẽ tiến hành tạo người dùng mới (**lab01**) và gán toàn bộ quyền cho người dùng mới đó thông qua lệnh sau:

```
mysql> CREATE USER 'lab01'@'localhost' IDENTIFIED BY 'Password123#@!';  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'lab01'@'localhost';  
Query OK, 0 rows affected (0.00 sec)
```

Sau đó, ta thoát khỏi MySQL bằng cách gõ `\q`.

Ta vào lại MySQL với quyền truy cập của là người dùng **lab01** bằng cách gõ lệnh sau: `sudo mysql -u lab01 -p`.

Ta tiến hành tạo 1 cơ sở dữ liệu có tên là **QLSV** và sử dụng cơ sở dữ liệu **QLSV** bằng lệnh sau:

```
mysql> CREATE DATABASE QLSV;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> USE QLSV;  
Database changed
```

Để cơ sở dữ liệu có thể lưu chữ dạng **UTF-8**, ta dùng lệnh sau:

```
mysql> ALTER DATABASE QLSV CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;  
Query OK, 1 row affected (0.02 sec)
```

Ta sẽ tạo một bảng **SINHVIEN** để lưu trữ thông tin với các trường tương ứng ở **Bài thực hành 1**. Vì trong biểu mẫu ở tập tin HTML các trường **Họ và tên, MSSV, Số điện thoại** và **Email** không được bỏ trống nên khi qua bên MySQL, ta sẽ thêm ràng buộc

NOT NULL vào cột tương ứng với các trường trên (riêng cột MSSV vì đã có ràng buộc PRIMARY KEY rồi nên không cần phải thêm ràng buộc NOT NULL nữa). Vì cơ sở dữ liệu lưu chữ dạng **UTF-8**, đối với những trường thông tin dạng chuỗi, thì ta sẽ khai báo kiểu dữ liệu là NVARCHAR thay vì VARCHAR. Ngoài ra, độ dài của những trường sẽ do người code quy định, riêng độ dài tối đa của **Số điện thoại** và **MSSV** là 15. Sau đây là code tạo bảng SINHVIEN:

```
mysql> CREATE TABLE SINHVIEN (  
-> MSSV NVARCHAR(15) PRIMARY KEY,  
-> HOTEN NVARCHAR(50) NOT NULL,  
-> EMAIL NVARCHAR(50) NOT NULL,  
-> SDT NVARCHAR(15) NOT NULL,  
-> NGAYSINH NVARCHAR(15),  
-> GIOITINH NVARCHAR(5),  
-> DIACHI NVARCHAR(100),  
-> GHICHU NVARCHAR(200) );  
Query OK, 0 rows affected (0.03 sec)
```

BÀI THỰC HÀNH 4

Điều chỉnh mã nguồn trong tập tin `submit_form.php` để nhận các giá trị được submit và lưu vào CSDL dùng MySQL. Điền và submit thử một form để kiểm tra hoạt động của mã nguồn đã điều chỉnh.

Gợi ý: Cần đảm bảo:

- Nút **Lưu** có type là **submit**.
- Đường dẫn xử lý của form đã trỏ đến `submit_form.php`.
- Thông tin kết nối CSDL đúng với MySQL đã xây dựng.
- Lấy được các tham số đã nhập trong form và tạo được SQL Command đúng.

Nhóm lựa chọn cách kết nối cơ sở dữ liệu thông qua MySQLi dùng hàm (MySQLi Procedural).

Trong tập tin **`submit_form.php`** ta chỉnh sửa code trong thẻ `<php>` như sau:

- Chỉnh sửa thông tin về tên server, tên người dùng, mật khẩu truy cập (nếu có), tên cơ sở dữ liệu và tên bảng:

```
session_start();  
// Change these configs according to your MySQL server  
$servername = "localhost";  
$username = "lab01";  
$password = "Password123#@!";  
$database = "QLSV";  
$table = "SINHVIEN";
```

- Ta sẽ kết nối cơ sở dữ liệu sử dụng hàm `mysqli_connect()`. Nếu kết nối thất bại thì thông báo "Connection failed". Ngược lại, ta tiến hành nhập dữ liệu vào MySQL sử dụng thông tin đã được nhập vào trong form bằng cách sử dụng hàm `mysqli_real_escape_string()` ứng với mỗi trường thông tin (*hàm `mysqli_real_escape_string()` cũng giúp cho loại bỏ những kí tự có thể gây ảnh hưởng đến câu lệnh SQL, làm cho câu lệnh SQL an toàn hơn*). Sau đó tiến hành tạo câu lệnh SQL chèn dữ liệu vào bảng. Chú ý là thứ tự chèn dữ liệu vào bảng phải giống với thứ tự cột được tạo trong bảng và đối với những cột có kiểu dữ liệu là `NVARCHAR` thì khi chèn dữ liệu phải thêm `N` ở đằng trước. Sau đó,

tiến hành chạy câu lệnh trên sử dụng hàm `mysqli_query()` và đóng kết nối cơ sở dữ liệu sử dụng hàm `mysqli_close()`.

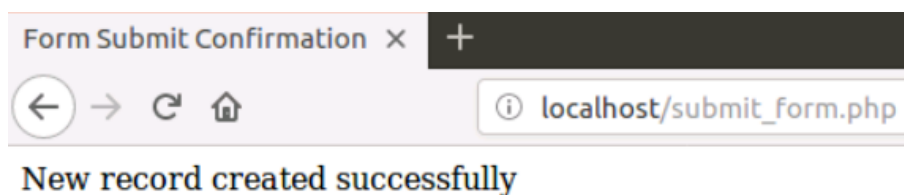
```
// Create connection
$conn = mysqli_connect($servername, $username, $password, $database);
// Check connection
if (!$conn) {
    $SESSION['msg'] = "Connection failed";
    #die("Connection failed: " . $conn->connect_error);
}
else {
    // 2 ways to get fields in form, the later is more secure
    // $name = $_POST['name'];
    // $name = mysqli_real_escape_string($conn, $_POST['name']);
    $name = mysqli_real_escape_string($conn, $_POST['name']);
    $student_id = mysqli_real_escape_string($conn, $_POST['student_id']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $phone = mysqli_real_escape_string($conn, $_POST['phone']);
    $dob = mysqli_real_escape_string($conn, $_POST['dob']);
    $gender = mysqli_real_escape_string($conn, $_POST['gender']);
    $address = mysqli_real_escape_string($conn, $_POST['address']);
    $note = mysqli_real_escape_string($conn, $_POST['note']);
    // Create SQL command to insert data to database
    $sql_command = "INSERT INTO $table VALUES(N'$student_id', N'$name', N'$email',
N'$phone', '$dob', N'$gender', N'$address', N'$note')";
    if (mysqli_query($conn, $sql_command))
        $SESSION['msg'] = "New record created successfully";
    else
        $SESSION['msg'] = mysqli_error($conn);
    mysqli_close($conn);
}
```

Kiểm tra hoạt động của mã nguồn:

- Điền thông tin vào form:

Họ và tên	<input type="text" value="Phan Thanh Hai"/>
Mã số sinh viên	<input type="text" value="18520705"/>
Email	<input type="text" value="18520705@gm.uit.edu.vn"/>
Số điện thoại	<input type="text" value="0912345678"/>
Ngày sinh	<input type="text"/>
Giới tính	<input type="text" value="Nam"/>
Địa chỉ	<input type="text" value="DH CNTT"/>
Ghi chú	<input type="text" value="Sinh viên nam 3"/>
<input type="button" value="Lưu"/>	

- Kết quả trên giao diện sau khi bấm nút **Lưu**:



- Kiểm tra dữ liệu đã thêm trong MySQL sử dụng lệnh sau:

```
mysql> SELECT * FROM SINHVIEN;
+-----+-----+-----+-----+-----+-----+-----+-----+
| MSSV   | HOTEN      | EMAIL                      | SDT      | NGAYSINH | GIOITINH | DIACHI   | GHICHU   |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 18520705 | Phan Thanh Hai | 18520705@gm.uit.edu.vn | 0912345678 |          | Nam      | DH CNTT | Sinh vien nam 3 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

BÀI THỰC HÀNH 5

Điều chỉnh form và viết mã nguồn Ajax để gửi thông tin form lưu vào CSDL qua Ajax và hiển thị thông báo thành công/thất bại cho người dùng.

Gợi ý: Cần đảm bảo:

- Sử dụng nút **Lưu** mới với type là **button**.
- Bắt sự kiện click của nút này và xử lý JavaScript với Ajax để gửi form đến URL của submit_form.php.
- Có thể sử dụng jQuery.

Trước tiên, ta sẽ thay nút **Lưu** có type là **submit** thành type là **button** (lí do là khi sử dụng Ajax, dữ liệu của form sẽ không được gửi trực tiếp từ form mà sẽ được gửi qua hàm). Ta cũng bổ sung thêm một thẻ <p> có id="result" để hiển thị kết quả thêm dữ liệu vào trong MySQL.

```
<!-- Button used in Bai thuc hanh 1-4

```

Trong Javascript (thẻ <script> để trong tập tin HTML) ta định nghĩa hàm ajaxFunction(). Trước tiên, ta khởi tạo đối tượng XMLHttpRequest.

```
function ajaxFunction() {
    var xmlhttp;
    if (window.XMLHttpRequest)
    {
        xmlhttp = new XMLHttpRequest();
    }
    else
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
}
```

Sau đó, kiểm tra nếu như đã gửi request thành công thì in thông báo lên màn hình (thay đổi nội dung của thẻ <p> có id="result" ở trước đó).

```
xmlhttp.onreadystatechange = function(){
    if(this.readyState==4 && this.status==200)
        $("#result").html(this.responseText);
};
```

Ta sử dụng jQuery để lấy thông tin của toàn bộ thẻ <input> nằm trong class="info".

```
// Using JQuery to get all input has class name is info
// the result of allInput is a array, its elements has format
// name | value
var allInput = $(".info").find("input").map(function(){
    return this.name + "=" + this.value;
}).get();

// Combine all of allInput's element by "&"
var data = allInput.join("&");
```

Cuối cùng, ta tiến hành gửi yêu cầu:

```
// Set up information for xmlhttp
xmlhttp.open("POST", "./submit_form.php", true);
xmlhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
xmlhttp.send(data);
}
```

Kiểm tra hoạt động của mã nguồn:

- Điền thông tin vào form:

Họ và tên	<input type="text" value="Nguyen Ly Dinh Nhi"/>
Mã số sinh viên	<input type="text" value="18521205"/>
Email	<input type="text" value="18521205@gm.uit.edu.vn"/>
Số điện thoại	<input type="text" value="0987654321"/>
Ngày sinh	<input type="text"/>
Giới tính	<input type="text" value="Nam"/>
Địa chỉ	<input type="text" value="DH CNTT"/>
Ghi chú	<input type="text" value="Sinh vien nam 3"/>
<input type="button" value="Lưu"/>	

- Kết quả trên giao diện sau khi bấm nút **Lưu**:

Simple Info Form x +

localhost/SimpleForm.html

Họ và tên	<input type="text" value="Nguyen Ly Dinh Nhi"/>
Mã số sinh viên	<input type="text" value="18521205"/>
Email	<input type="text" value="18521205@gm.uit.edu.vn"/>
Số điện thoại	<input type="text" value="0987654321"/>
Ngày sinh	<input type="text"/>
Giới tính	<input type="text" value="Nam"/>
Địa chỉ	<input type="text" value="DH CNTT"/>
Ghi chú	<input type="text" value="Sinh vien nam 3"/>
<input type="button" value="Lưu"/>	

New record created successfully

- Kiểm tra dữ liệu đã thêm trong MySQL sử dụng lệnh sau:

```
mysql> SELECT * FROM SINHVIEN;
```

MSSV	HOTEN	EMAIL	SDT	NGAYSINH	GIOITINH	DIACHI	GHICHU
18520705	Phan Thanh Hat	18520705@gm.uit.edu.vn	0912345678		Nam	DH CNTT	Sinh vien nam 3
18521205	Nguyen Ly Dinh Nhi	18521205@gm.uit.edu.vn	0987654321		Nam	DH CNTT	Sinh vien nam 3

```
2 rows in set (0.00 sec)
```


BÀI THỰC HÀNH 6

Viết một tập tin tương tự khác, trong đó khi mở lên thì tự động gửi một form với các trường tham số như **Bài thực hành 1** đến submit_form.php.

Ta sẽ tạo một tập tin tương tự có tên là SimpleFormCopy.html giống với SimpleForm.html với phần Javascript như sau:

```
// Using Ajax by XMLHttpRequest
function ajaxFunction(data) {
    var xmlhttp;
    if (window.XMLHttpRequest)
    {
        xmlhttp = new XMLHttpRequest();
    }
    else
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");

    xmlhttp.onreadystatechange = function(){
        if(this.readyState==4 && this.status==200)
            $("#result").html(this.responseText);
    };

    // Sset up information for xmlhttp
    xmlhttp.open("POST", "./submit_form.php", true);
    xmlhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    xmlhttp.send(data);
}

// Repaired data for auto call ajaxFunction
var mydata = "name=Pham Khoi
Nguyen&student_id=18520114&email=18520114@gm.uit.edu.vn&phone=0943218765&dob=&gender=Nu&address=DH
CNTT&note=Sinh vien nam 3";

// With window.onload, the ajaxFunction function will be automatically called when the
page load
window.onload = ajaxFunction(mydata);
```

Giờ ta truy cập vào SimpleFormCopy.html rồi tải lại trang. Khi đó, nó sẽ tự động gửi các giá trị đã được chuẩn bị sẵn tới submit_form.php thông qua window.onload, và trả về kết quả "New record created successfully" (nếu nhập đúng), tức là đã thêm dữ liệu thành công vào database.

Simple Info Form

localhost/SimpleFormCopy.html

Họ và tên

Mã số sinh viên

Email

Số điện thoại

Ngày sinh

Giới tính

Địa chỉ

Ghi chú

Lưu

New record created successfully

Kiểm tra dữ liệu đã thêm trong MySQL sử dụng lệnh sau:

```
mysql> SELECT * FROM SINHVIEN;
```

MSSV	HOTEN	EMAIL	SDT	NGAYSINH	GIOITINH	DIACHI	GHICHU
18520114	Pham Khoi Nguyen	18520114@gm.uit.edu.vn	0943218765		Nu	DH CNTT	Sinh vien nam 3
18520705	Phan Thanh Hai	18520705@gm.uit.edu.vn	0912345678		Nam	DH CNTT	Sinh vien nam 3
18521205	Nguyen Ly Dinh Nhi	18521205@gm.uit.edu.vn	0987654321		Nam	DH CNTT	Sinh vien nam 3

3 rows in set (0.00 sec)

BÀI THỰC HÀNH 7

Phân tích file `login.php`, sinh viên thực hiện các yêu cầu bên dưới:

- Tạo một bảng chứa thông tin người dùng đơn giản trong MySQL, với các trường phù hợp dựa trên phân tích file `login.php`.
- Không sửa mã nguồn HTML trong tập tin `SimpleForm.html`, hãy sử dụng Javascript để tìm cách tạo một form cho phép nhập tài khoản và gọi tập tin xử lý `login.php` để đăng nhập.

Cần đảm bảo: Thay đổi chỉ có thể tạo ra lúc đang chạy file `SimpleForm.html`, nội dung của file trên ổ đĩa chỉ cho phép thêm file JavaScript.

1. Yêu cầu 1

Tạo bảng LOGIN chứa thông tin người dùng trong MySQL, với các trường phù hợp dựa trên phân tích file `login.php`:

```
mysql> show fields from user;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| username | varchar(255) | NO   | PRI | NULL    |       |
| password | varchar(255) | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)

mysql>
```

2. Yêu cầu 2

Tạo một form login đơn giản (`LoginForm.html`):

```

<html>
<head>
  <title>Login Page</title>
  <meta charset="UTF-8">
  <style>
    body{
      font-family: Arial;
    }

    .label{
      width: 10%;
      float: left;
    }

    .info{
      padding: 5px;
    }
    form{
      padding-left: 30px;
    }
  </style>
</head>

<body>
  <form id="login_form" action="./login.php" application="x-www-form-urlencoded" method="POST">
    <div class="info">
      <div class="label">username </div>
      <!-- the value of pattern attribute indicate that username is only alphabet character, number, underscore or dash
      <input type="text" name="username" pattern="[a-zA-Z0-9_-]+" required></input><br >
    </div>

    <div class="info">
      <div class="label">password </div>
      <input type="password" name="password" minlength="8" required></input><br>
    </div>

    <input type="submit" name="submit" value="Login"></input><br>
  </form>
</body>

```

Để SimpleForm.html có thể tự động chuyển đến LoginForm.html ta thêm đoạn code sau vào script của SimpleForm.html:

```

// The page will be automatically redirected to http://localhost/LoginForm.html
// by window.onload
window.onload = function() {
  location.replace("http://localhost/LoginForm.html");
}

```

Khi đó, SimpleForm.html sẽ tự động gọi đến hàm tại window.onload khi tải trang. Còn đối với file login.php, để có thể lưu thông tin tài khoản ta chuẩn bị thông tin database như sau:

```

session_start();
//Change these configs according to your MySQL server
$servername = "localhost";
$username = "lab01";
$password = "Password123#@!";
$database = "QLSV";
$table = "user";

```

Bên cạnh đó ta cần chỉnh sửa code 1 chút để có thể đạt được kết quả mong muốn, bằng cách sử dụng lệnh `mysqli_fetch_assoc` để có thể lấy được bản ghi đầu tiên trong trường hợp SQL trả về nhiều kết quả:

```

$username = $_POST['username'];
$password = $_POST['password'];
$hash_pass = hash("sha256", $password);
$isPass = 0;

if(!preg_match("/'+'/", $username)){ // prevent from sql injection which using ' letter

    $sql_command = "SELECT * FROM ".$table." WHERE username='".$username."' and password='".$hash_pass."'";

    $result = mysqli_query($conn, $sql_command);
    if (mysqli_num_rows($result) > 0){
        // mysqli_fetch_assoc($result) will return the first row
        $row = mysqli_fetch_assoc($result);
        $_SESSION['name'] = $row['username'];

        $isPass = 1;
    }

    mysqli_close($conn);
}

if(!$isPass)
{
    $_SESSION['name'] = null;
    $_SESSION['msg'] = "Login failed";
}

```

Chuẩn bị một số bản ghi cho database:

```

mysql> select * from user;
+-----+-----+
| username | password |
+-----+-----+
| dinhnhhi | d3c11863384fa17c1bb70e0ebe2595adc0a30573f18305cf6f2d24eeded79691 |
| khoinguyen | 65ace90b77fa09f00dd08a32bbff4fd788135b2a9905ed853b222806e8970f68 |
| thanhhai | dc6d23263d796303c80742b5a24ec45d8ebd7083dd13dc42cd78bfdd77577032 |
+-----+-----+
3 rows in set (0.01 sec)

mysql>

```

Kết quả khi ta truy cập SimpleForm.html thì sẽ được tự động chuyển đến LoginForm.html:

The image shows two browser windows. The first window, titled 'Login Page', is at 'localhost/LoginForm.html'. It contains a login form with 'username' set to 'dinhnhhi' and 'password' masked with dots. A 'Login' button is visible. The second window, titled 'Simple Info Form', is at 'localhost/login.php' and displays the text 'Welcome dinhnhhi'.