

# LAB 5: GỬI NHẬN DỮ LIỆU GIỮA CLIENT-SERVER

## 1. Mục tiêu

- Hiểu cách truyền dữ liệu giữa client và server thông qua form và URL
- Nắm vững và sử dụng các mảng có sẵn của PHP: `$_GET`, `$_POST`, `$_REQUEST`, `$_FILES`, ...
- Hiểu và vận dụng các mảng: `$_SERVER`, `$_SESSION`, `$_COOKIE`

## 2. Tóm tắt lý thuyết

- ✓ Để giao tiếp với server, người sử dụng client thường dùng một form để nhập liệu và gửi lên server hoặc click vào url.

Các thông tin nhập vào form hay truyền qua url được lưu trữ trong các mảng có sẵn trên server.

- `$_GET`: lưu các thông tin gửi từ form có method=get hoặc từ url
  - `$_POST`: lưu giữ các thông tin từ form có method=post
  - `$_REQUEST`: có thể dùng thay thế `$_GET` hoặc `$_POST`.
  - `$_FILES`: mảng lưu giữ các thông tin file được gửi kèm theo form.
- ✓ Vấn đề gửi nhận dữ liệu:
    - URL: dữ liệu QueryString trong url có thể được phân tích và truy xuất từ server thông qua mảng `$_GET` hoặc `$_REQUEST`.

Ví dụ: path/a.php?**x**=1&**y**=2

Khi đó trang a.php có thể lấy giá trị x và y qua mảng `$_GET` và `$_REQUEST` như sau:

<code>\$v1 = \$_GET["x"];</code>	<code>// \$v1 = 1</code>
<code>\$v2 = \$_REQUEST["y"];</code>	<code>// \$v2 = 2</code>

- Những phần tử trong form không đặt thuộc tính name sẽ không thể gửi dữ liệu lên máy chủ. Các phần tử có giá trị của thuộc tính name sẽ có thể được gửi đi. Trên server, các giá trị của thuộc tính name này sẽ trở thành index của các mảng tương ứng. (`$_GET`, `$_POST`, `$_FILES`, `$_REQUEST`).

- Với các nhóm thành phần: (radio, checkbox – là các thành phần thường sử dụng một tên cho tất cả các phần tử của nhóm): chỉ có thành phần nào được chọn (checked) mới được gửi đi.
- Nút submit: nút nào được click mới được gửi lên server.
- Các thành phần button, reset: không được gửi lên máy chủ.
- Để gửi các tập tin lên máy chủ, form phải thiết lập giá trị cho thuộc tính: `enctype="multipart/form-data"`.
- Mảng `$_FILES`: lưu các thông tin của file được gửi lên máy chủ. Mỗi phần tử này là một mảng lưu trữ nhiều thông tin của file: tên file, file tạm trên máy chủ, kích thước file, mã lỗi khi gửi file, kiểu file.

### 3. Ví dụ

- 3.1 Ví dụ lab5\_1.php:
  - a. Phân tích các QueryString trong ví dụ khi click vào các link trong trang.
  - b. Sử dụng form và phương thức get tương ứng với các QueryString trong câu.
- 3.2 Ví dụ lab5\_2.php: sử dụng form đơn giản và sử dụng phương thức post. Nhập dữ liệu trong form sau đó submit form lên máy chủ và hiển thị các kết quả vừa nhập.
- 3.3 Ví dụ **lab5\_3**: kết hợp gửi nhiều thông tin và upload hình ảnh từ client lên máy chủ
  - a. lab5\_3/1.php: chứa form nhập liệu.
  - b. lab5\_3/2.php: nhận dữ liệu từ form 1.php và kiểm tra dữ liệu nhập:
    - Nếu thông tin không hợp lệ: (Tên chưa nhập, không chọn giới tính hoặc file upload có lỗi và không phải file hình): Báo lỗi.
    - Nếu thông tin hợp lệ: Lưu hình vào thư mục image. Kiểm tra giới tính và in ra các thông báo thích hợp cùng với hình vừa upload.
- 3.4 Ví dụ **lab5\_4**: Phân chia các chức năng của từng phần (font-end và back-end) trong website thành nhiều module nhỏ hơn để dễ quản lý. Các module này được load (include) vào trang chính dựa vào các tham số được truyền trên URL (mod và ac).  
 Xem và giải thích các dòng trong file mod.php và các file index.php trong các thư mục con của module.

### 4. Vận dụng

- 4.1 Nhận dữ liệu trên server thông qua URL. Thư mục **Lab5\_4\_1/**  
File **1.php**: Cho mảng \$arr có nội dung như sau:

```
$arr = array();  
$r = array("id"=>1, "name"=>"Product1");  
$arr[] = $r;  
$r = array("id"=>2, "name"=>"Product2");  
$arr[] = $r;  
$r = array("id"=>3, "name"=>"Product3");  
$arr[] = $r;  
$r = array("id"=>4, "name"=>"Product4");  
$arr[] = $r;
```

Hãy sử dụng mảng \$arr để in ra màn hình các liên kết. Mỗi liên kết gửi id của phần tử mảng tương ứng lên trang 2.php và text hiển thị của liên kết là giá trị thuộc tính name tương ứng. Ví dụ, với phần tử đầu tiên, liên kết có dạng: `<a href="lab5_4_2.php?id=1">Product1</a>`

File: **2.php** in ra màn hình giá trị id nhận được.

- 4.2 Xây dựng trang (hoặc các trang) tìm kiếm thông tin sản phẩm. (Lưu trong thư mục **lab5\_4\_2/**)

Trang gồm một form để nhập thông tin cần tìm kiếm.

Form gồm:

- Textbox: nhập tên sản phẩm cần tìm.
- Hai radiobutton cách tìm: lựa chọn một trong hai: gần đúng, chính xác.
- Combobox: loại sản phẩm (là một combobox có 4 lựa chọn: tất cả, loại 1, loại 2, loại 3).
- Form sử dụng phương thức get.

Hãy in ra màn hình các giá trị mà người sử dụng vừa nhập vào form:

- Tên sản phẩm:
- Cách tìm: (gần đúng hay chính xác).
- Loại sản phẩm: in ra loại sản phẩm nếu không phải chọn tất cả.

- 4.3 Xây dựng trang để đăng ký thông tin thành viên (trang lab5\_4\_3.php).  
Trang gồm 2 phần.

a. Phần form nhập liệu. Form có các thành phần:

- Tên đăng nhập (\*)
- Mật khẩu (\*)
- Nhập lại mật khẩu. (\*)
- Giới tính (\*)

- Sở Thích
- Hình ảnh (tùy chọn).
- Tỉnh (là combobox) (\*)
- Submit, reset

b. Phần nhận dữ liệu: (chính trang này): Kiểm tra dữ liệu nhập và:

- In ra thông tin vừa nhập nếu dữ liệu đúng.
- In ra thông báo nếu dữ liệu nhập vào không hợp lệ.
- **Dữ liệu hợp lệ:** Các thành phần đánh dấu (\*) bắt buộc phải nhập liệu, mật khẩu và nhập lại mật khẩu phải trùng nhau, hình ảnh nếu có phải là một loại file hình (.jpg, .png, .bmp, .gif).

4.4 Xây dựng lại các chức năng cho admin dựa vào ví dụ **lab5\_4**. Khi click vào các link sẽ include các trang tương ứng trong admin.

## 5. Nâng cao

- 5.1 Xây dựng lại ví dụ 5\_2 ở trên, nhưng file chứa form và in ra kết quả là cùng một file và form giữ nguyên giá trị value vừa nhập trong các thành phần khi đã submit.
- 5.2 Viết lại ví dụ 5\_3 ở trên nhưng cho phép upload nhiều hình ảnh.
- 5.3 Viết lại trang form đăng ký thành viên trong phần 4 (lab5\_4\_3.php). Tuy nhiên thêm phần kiểm tra form bằng javascript trước khi kiểm tra trên server. (gợi ý: có thể sử dụng javascript thông thường, sử dụng kết hợp với HTML5, hoặc sử dụng plug-in jQuery form validation).