PHỎNG VẤN PHP

1. **PHP có mấy kiểu dữ liệu cơ bản**

Integer, double, Boolean, string, array, object.

1. **Các đặc điểm của biến $\_GET**

* Khi phương thức HTTP GET được sử dụng trong form, các tham số sẽ được gắn vào URL khi form đó được đệ trình. Khi đó, các tham số này sẽ được lưu vào mảng $\_GET
* Biến $\_GET là mảng chứa các cặp khóa/giá trị cho dữ liệu được truyền vào theo yêu cầu HTTP
* Khi viết mã hoặc nhập URL để yêu cậu một trang PHP, bạn có thể thêm vào một danh sách tham số bắt đầu bằng dấu chấm hỏi và không có khoảng trắng. Mỗi tham số truyền vào sẽ bao gồm tên, dấu bằng “=” và giá trị. Để phân tách các tham số, dung ký hiệu &.
* Thẻ <a> luôn sử dụng phương thức HTTP GET khi muốn truyền tham số đến một trang.
* Biến $\_GET được gọi là biến siêu toàn cục (supperglobal variable) vì nó có thể được truy cập bất kì đâu trong mã PHP

1. **Đặc điểm của biến $\_POST**

Phương thức HTTP POST hoạt động tương tự như phương thức GET nhưng các tham số không được gắn vào sau URL

1. **Khi nào nên sử dụng HTTP GET, HTTP POST**

* HTTP GET:

Khi yêu cầu trang lấy dữ liệu từ server cơ sở dữ liệu.

Khi muốn thực thi yêu cầu nhiều lần mà không gây ra lỗi

* HTTP POST

Khi yêu cầu trang viết dữ liệu lên server cơ sở dữ liệu

Khi việc thực hiện nhiều yêu cầu có thể gây ra lỗi

Khi bạn không muốn truyền tham số vào URL vì lý do bảo mật

Khi bạn không muốn tham số đi kèm với URL trong trường hợp người sử dụng đánh dấu trang

Khi bạn muốn truyền hơn 4KB dữ liệu

1. **Phân biệt isset và empty**

* Isset($var)

Trả về giá trị TRUE nếu $var đã được thiết lập giá trị khác NULL

* Empty($var)

Trả về giá trị TRUE nếu $var chưa được thiết lập giá trị, hoặc chứa giá trị NULL hoặc chuỗi rỗng

1. **Phân biệt include, include\_once, require, require\_once, exit, die**

|  |  |
| --- | --- |
| Tên | Mô tả |
| Include | Chèn và chạy file được chèn. Tuy nhiên, nếu hàm lỗi, nó chỉ đưa ra cảnh báo và cho phép đoạn mã tiếp tục chạy. |
| Include\_once | Giống như hàm include, nhưng hàm này đảm bảo chỉ bao hàm file đúng 1 lần |
| Require | Hoạt động tương tự như include. Tuy nhiên nếu hàm lỗi (không có file), nó gây ra lỗi hệ thống và dừng đoạn mã |
| Require\_once | Giống require, những đãm bão file này chỉ được yêu cầu đúng một lần |
| Exit | Thoát đoạn PHP hiện thời. |
| die | Hoạt động tương tự như exit |

1. **Cookie là gì?**

Cookie cung cấp cho ứng dụng web một phương thức lưu trữ thông tin trên trình duyệt của người dung và truy xuất khi trình duyệt gửi yêu cầu xem trang.

1. **Tính chất của cookie?**

* Cookie là cặp tên/giá trị được lưu trên trình duyệt
* Trên server, ứng dụng web tạo ra cookie và gửi nó tới trình duyệt. Trên máy client, trình duyệt lưu cookie và gửi nó trở lại server mỗi khi truy cập trang từ server đó.
* Mặc định, cookie chỉ có hiệu lực cho đến khi người dung đóng trình duyệt. Tuy nhiên, bạn có thể thiết lập để cookie tồn tại trong trình duyệt của người dùng với thời trên lên đến 3 năm
* Có một số người dùng tắt cookie. Do đó, chúng ta không thể giả định tất cả người dùng đều bật cookie
* Thông thường trình duyệt chỉ cho phép lưu 20 cookie cho một trang và lưu tổng cộng 300 cookie. Ngoài ra, mỗi cookie cũng chỉ được giới hạn 4Kb dung lượng.
* Cookie có thể được liên kết với một hoặc nhiều tên miền con

1. **Những cách dùng phổ biến của cookie**

* **Cho phép người dùng không cần đăng nhập và điền form đăng ký** để thu thập những dữ liệu như tên người dùng, mật khẩu, địa chỉ hoặc thông tin thẻ tín dụng
* **Tùy chỉnh trang** hiển thị các thông tin như dự báo thời tiết, tỷ số các môn thể thao và giá cổ phiếu
* **Quảng cáo hướng đích**, ví dụ hiển thị banner quảng cáo nhằm vào khách hang mục tiêu

1. Session là gì?

Session đơn giản là 1 cách để chúng ta (các lập trình viên) lưu lại dữ liệu của người dùng sử dụng website. Giá trị của Session được lưu trong một tập tin trên máy chủ. Ví dụ khi bạn đăng nhập vào trang hoclaptrinh.org sử dụng địa chỉ email và mật khẩu người dùng mà bạn đã đăng ký trước đó. Máy chủ sau khi xác thực được thông tin bạn cung cấp là đúng nó sẽ sinh ra một tập tin (hay chính là session của trình duyệt của bạn) chứa dữ liệu cần lưu trữ của người dùng.

1. Tại sao cần session

Việc giao tiếp giữa trình duyệt với máy chủ của website (hay webserver) được thực hiện thông qua hàng loạt các router trên mạng internet. Một vấn đề đặt ra trong quá trình giao tiếp này đó là làm sao để phân được giữa cá trình duyệt (máy tính) khác nhau. Ví dụ như bạn truy cập một trang bán hàng thì máy chủ cần:

* Phân biệt được các máy tính (trình duyệt) khác nhau: nói cách khác máy chủ cần phân biệt được trong những lượt truy cập gửi tới máy chủ thì đâu là từ máy tính bạn và đâu là của người khác
* Lưu thông tin của giỏ hàng: ví dụ như thông tin về số lượng hàng hoá bạn đã thêm vào giỏ hàng

Và để giải quyết vấn đề trên khái niệm **session** được ra đời.

## Hoạt động của session

## Mở đầu, trình duyệt trện máy client gửi yêu cầu xem trang PHP tới server web. Sau đó, PHP kiểm tra liệu yêu cầu đả bao gồm session ID chưa. Nếu chưa, PHP sẽ tạo một session mới trên server và gán cho nó một session ID duy nhật. Lúc này, ứng dụng có thể lưu dữ liệu vào phiên. Sau đó, session ID sẽ được gửi trả lại trình duyệt như cookie trong phản hồi.

## Khi trình duyệt gửi các yêu cầu sau đó, cookie session ID được gộp trong yêu cầu. PHP cũng kiểm tra liệu yêu cầu đã bao gồm session ID PHP chưa. Nếu có, PHP sử dụng session ID để truy cập, điều chính hoặc thêm dữ liệu khi cần

## Mặc định, PHP 5.3 chỉ sử dụng cookie để làm việc với session ID. Vì vậy, việc theo dõi phiên chỉ hoạt động khi người dùng bật cookie. Đối với hầu hết các ứng dụng web, đó là điều chúng ta mong muốn vì hầu hết các trình duyệt đều cho phép cookie.

## Tuy nhiên, nếu bạn muốn ứng dụng làm việc chính xác ngay cả với những người dùng tắt cookie, thì PHP có thể viết session ID trên URL. Cách này đội khi được gọi là mã hóa URL (URL encoding) và bạn có thể cấu hình hệ thống của mình sao cho PHP có thể viết mã tường minh cho session ID trên URL. Đáng tiếc là cách này gây ra vấn đề liên quan đến việc hiển thị trang web trên các công cụ tìm kiếm.