**1, Phương thức GET.**

**Lý Thuyết**

-Phương thức GET gửi dữ liệu từ client lên server thông qua các paramter( tham số) trên URL, từ đó server sẽ phân tích dữ liệu để thực thi một hành động nào đó. Các tham số mà phương thức GET gửi lên sẽ bắt đầu sau dấu **?**. Và sẽ nối nhau giữa các khóa thông qua dấu **&**

**VD1**: Với url **http://toidicode.com?id=17**lúc này chúng ta đang gửi dữ liệu bằng phương thức GET lên server với tham số id=17.

**VD2**: Với url **http://toidicode.com?age=21&name=VuThanhTai**lúc này chúng ta đang gửi dữ liệu bằng phương thức GET lên server với tham số age=21 và name=VuThanhTai

**SERVER nhận dữ liệu**

-Để thông báo cho server nhận dữ liệu GET khi có thì chúng ta chỉ cần sử dụng **$\_GET** đây là một biến toàn cục lưu trữ dưới dạng mảng bất tuần tự.

**VD3**: như với URL ở **VD2** thì trên server sẽ lưu trữ cho chúng ta như sau:

$\_GET = [

'age' => 21,

'name' => 'VuThanhTai'

];

Như thế thì chúng ta chỉ cần lấy giá trị theo kiểu mảng là xong

echo $\_GET['age']; // lấy age

echo $\_GET['name']; //lấy name;

**Thực Hành**

-Đầu tiên mọi người tạo cho mình 2 tệp tin index.html và get.php ở cùng cấp thư mục trên server hoặc server ảo.

Tiếp theo chúng ta cần phải tạo ra một form để gửi dữ liệu get lên server trong file index.html

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<title>Form</title>

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">

<!-- Optional theme -->

<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap-theme.min.css">

<!-- Latest compiled and minified JavaScript -->

<script src="//netdna.bootstrapcdn.com/bootstrap/3.1.1/js/bootstrap.min.js"></script>

</head>

<body>

<div class="container">

<form action="get.php" method="GET" role="form">

<legend>Get files</legend>

<div class="form-group">

<label for="">Tên</label>

<input type="text" class="form-control" id="" placeholder="Nhập vào tên" name="name">

</div>

<div class="form-group">

<label for="">Tuổi</label>

<input type="number" class="form-control" id="" placeholder="Nhập vào tuổi" name="age">

</div>

<button type="submit" class="btn btn-primary">Submit</button>

</form>

</div>

</body>

</html>

-**Giải thích**: Form trên mình đã tạo ra 2 trường name và age và gửi dữ liệu lên server và cụ thể là file get.php

-Sau khi đã tạo được form gửi dữ liệu lên server thì tiếp theo chúng ta cần viết code để server nhận dữ liệu trong file get.php

<?php

echo 'Tên: ' . $\_GET['name'] . '<br>';

echo 'Tuổi: ' . $\_GET['age'];

?>

**Chạy**

**-**Ngoài cách gửi dữ liệu bằng form trên thì vì đây là phương thức GET lên hoàn toàn các bạn có thể truyền tham số bằng tay thông qua paramter trên URL. ví dụ như: http://localhost/get.php?name=vuthanhtai&age=21

**2, Phương thức POST.**

**Lý thuyết**

-Khác với phương thức GET phương thức POST không gửi dữ liệu thông qua paramter trên URL, nên chúng ta không thể nhìn thấy được dữ liệu đang được gửi là gì.

**Server nhận dữ liệu**

-Để thông báo cho server nhận dữ liệu POST khi có thì chúng ta chỉ cần sử dụng **$\_POST**đây là một biến toàn cục lưu trữ dưới dạng mảng bất tuần tự.

-Và để lấy giá trị của POST chúng ta sử dụng cú pháp:

$\_POST['name'];

**Trong đó:**name là tên của post data chúng ta muốn nhận

**Thực hành**

-Đầu tiên mọi người tạo cho mình 2 tệp tin index.html và post.php ở cùng cấp thư mục trên server hoặc server ảo.

-Tiếp theo chúng ta cần phải tạo ra một form để gửi dữ liệu POST lên server trong file index.html

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<title>Form</title>

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">

<!-- Optional theme -->

<link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap-theme.min.css">

<!-- Latest compiled and minified JavaScript -->

<script src="//netdna.bootstrapcdn.com/bootstrap/3.1.1/js/bootstrap.min.js"></script>

</head>

<body>

<div class="container">

<form action="post.php" method="POST" role="form">

<legend>Get files</legend>

<div class="form-group">

<label for="">Tên</label>

<input type="text" class="form-control" id="" placeholder="Nhập vào tên" name="name">

</div>

<div class="form-group">

<label for="">Tuổi</label>

<input type="number" class="form-control" id="" placeholder="Nhập vào tuổi" name="age">

</div>

<button type="submit" class="btn btn-primary">Submit</button>

</form>

</div>

</body>

</html>

-**Giải thích**: Form trên mình đã tạo ra 2 trường name và age và gửi dữ liệu lên server và cụ thể là file post.php

-Sau khi đã tạo được form gửi dữ liệu lên server thì tiếp theo chúng ta cần viết code để server nhận dữ liệu trong file post.php

<?php

echo 'Tên: ' . $\_POST['name'] . '<br>';

echo 'Tuổi: ' . $\_POST['age'];

?>

Tiếp theo các bạn chạy code lên và cảm nhận nhé!

**3, Kiểm tra với isset.**

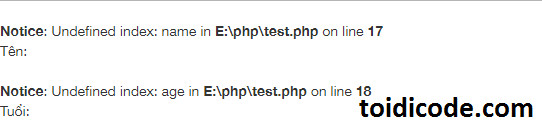
-Hàm isset() trong PHP có chức năng kiểm tra xem biến có tồn tại hay không. Nó sẽ trả về TRUE nếu biến đó có tồn tại và ngược lại FALSE nếu biến đó không tồn tại

**Cú pháp:**

isset($bien);

Trong đó: **$bien** là biến mà bạn muốn kiểm tra sự tồn tại.

Đi vào với phần thực hành ở trên thì các bạn có thể truy cập vào 2 files get.php và post.php các bạn sẽ thấy ngay dòng thông báo.



Vậy sau khi đã tìm hiểu về hàm isset thì chúng ta chỉ cần chỉnh sửa nhỏ trong 2 files get.php và post.php như sau:

get.php

<?php

if (isset($\_GET['name']) && isset($\_GET['age'])) {

echo 'Tên: ' . $\_GET['name'] . '<br>';

echo 'Tuổi: ' . $\_GET['age'];

}

 post.php

<?php

if (isset($\_POST['name']) && isset($\_POST['age'])) {

echo 'Tên: ' . $\_POST['name'] . '<br>';

echo 'Tuổi: ' . $\_POST['age'];

}

**Kết luận:** qua phần thực hành trên các bạn thấy isset() hoàn toàn rất hữu dụng đúng không nhỉ, và mình cũng khuyên các bạn thường xuyên sử dụng isset() để ràng buộc dữ liệu cho chặt chẽ nhé!

**4, GET VS POST.**

**Giống nhau:**

-Cả GET và POST đều dùng để gửi dữ liệu lên server

**Khác nhau:**

-Xét về độ bảo mật thì phương thức POST sẽ bảo mật hơn phương thức GET.

Xét về tốc độ thì phương thức GET nhanh hơn phương thức POST.