# CÁCH LẤY GIÁ TRỊ TRONG FORM

# Với radio button

* Để lấy được giá trị của radio, bắt buộc phải khai báo giá trị cho thuộc tính value của input này, thường là 1 giá trị number, sau đó dựa vào giá trị này để hiển thị text tương ứng với radio đó
* Ví dụ:

<**input type="radio" name="gender" value="1"** /> Male

<**input type="radio" name="gender" value="2"** /> Female

* Cách hiển thị, giả sử form có method POST

…

$gender = $\_POST[**'gender'**];  
**if** ($gender == 1) {  
 **echo 'Male'**;  
}  
**else** {  
 **echo 'Female'**;  
}

…

# Với checkbox

* Cần khai báo name của các checkbox giống hệt nhau, và phải có dạng array, cần sử dụng vòng lặp foreach để lặp mảng này, để hiển thị ra text tương ứng
* Ví dụ

<**input type="checkbox" name="checkbox[]" value="1"** /> Check1  
<**input type="checkbox" name="checkbox[]" value="2"** /> Check2  
<**input type="checkbox" name="checkbox[]" value="3"** /> Check3

* Cách hiển thị, giả sử form có method POST

…

$checkboxArr = $\_POST[**'checkbox'**]; //name of field  
**foreach** ($checkboxArr **as** $checkbox) {  
 **if** ($checkbox == 1) {  
 **echo 'Check1'**;  
 }  
 **else if** ($checkbox == 2) {  
 **echo 'Check2'**;  
 }  
 **else** {  
 **echo 'Check3'**;  
 }  
}

…

# Với select

* Cần khai báo name cho thẻ select, các option của select cần khai báo thuộc tính value, sau đó check để hiển thị text tương ứng tương tự radio button
* Ví dụ

<**select name="select"**>  
 <**option value="1"**>Option 1</**option**>  
 <**option value="2"**>Option 2</**option**>  
</**select**>

* Cách hiển thị, giả sử form có method POST
* …

$gender = $\_POST[‘**select’**];  
**if** ($gender == 1) {  
 **echo 'Option 1'**;  
}  
**else** {  
 **echo 'Option 2'**;  
}

* …

## EXERCISE DAY 3

## Bài tập 1: Sử dụng PHP làm bài tập sau

# Xây dựng form Login như sau sử dụng HTML/CSS/JS

# 

# Chú ý: Trường username và password là 2 trường bắt buộc nhập, validate trường này thông qua server trước.

# Kiểm tra thông tin: Nếu thông tin nhập tương ứng: username là “admin” và password là “admin” thì sẽ hiển thị tên người dùng ở cuối form login, hoặc ở vị trí nào tuỳ ý muốn. Tuy nhiên nếu nhập thông tin bị sai thì sẽ hiển thị bằng chữ đỏ trên màn hình: “Thông tin đăng nhập không chính xác. Xin hãy kiểm tra lại”.

# Màu của phần trên và dưới là lightgreen.

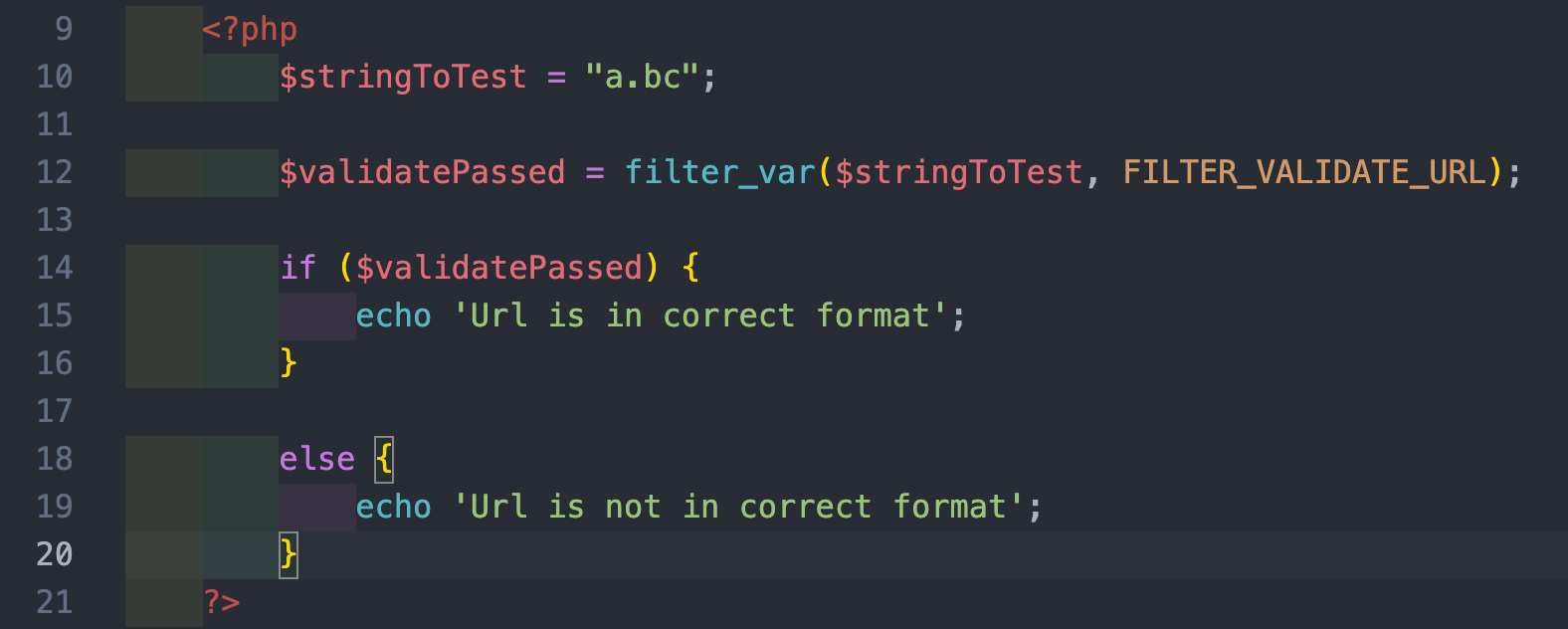
## Bài tập 2: Sử dụng Bootstrap và PHP để thực hiện bài tập sau

### 

* Sử dụng Bootstrap để dựng form
* Có responsive trên mobile
* State: Có 2 giá trị:
  + 1 – Johor
  + 2 – Massachusetts
  + 3 – Washington
* Thực hiện validate dữ liệu cho form, sử dụng validate trên PHP:
* Firstname không được để trống, và hiển thị thông báo lỗi màu đỏ sau “Firstname không được để trống”
* Lastname không được để trống, và hiển thị thông báo lỗi màu đỏ sau “Lastname không được để trống”
* Email không được để trống, validate sử dụng filter\_var() hiển thị thông báo lỗi màu đỏ “Email không được để trống”.

filter\_var($stringToTest, $filterMode)

Ví dụ như sau:



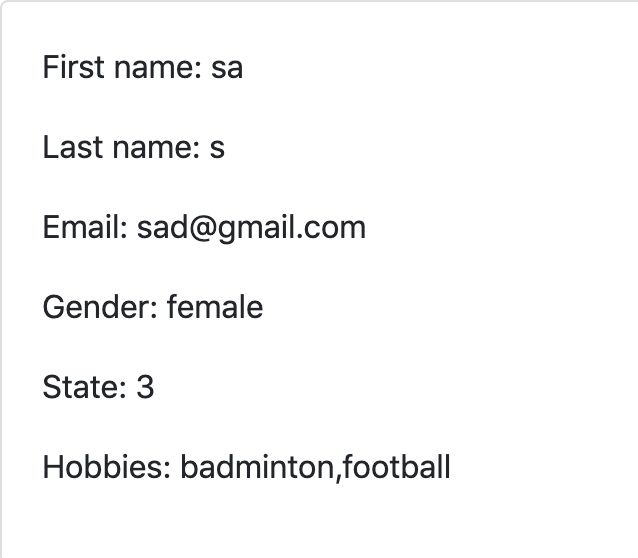
🡺 In ra: “Url is not in correct format”.

Chú ý: Trong trường hợp form có lỗi validation, phải giữ nguyên tất cả các giá trị đã nhập trên form, không được xoá dữ liệu cũ đã có trên form

Sau khi validate thành công và submit form, hiển thị toàn bộ các thông tin mà user đã nhập, có dạng sau:

Đăng ký thành công!

Thông tin của bạn:

****

Đồng thời lưu trữ giá trị nhập trên form tương ứng, bất kể trường hợp nhập đúng lẫn nhập sai.

Hàm in\_array($find, $searchedArray) trả về true khi $find là một phần tử thuộc $searchedArray

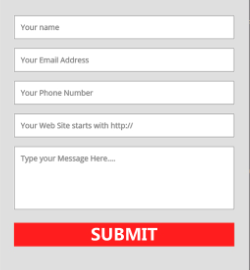
Ví dụ:

* in\_array(‘badminton’, [‘badminton’,’football’,’soccer’]) => TRUE

Tham khảo thêm về cách hiển thị errors trong bootstrap

[**https://getbootstrap.com/docs/4.5/components/forms/#how-it-works**](https://getbootstrap.com/docs/4.5/components/forms/#how-it-works)

## Bài tập 3: Sử dụng PHP để hoàn thành bài tập sau



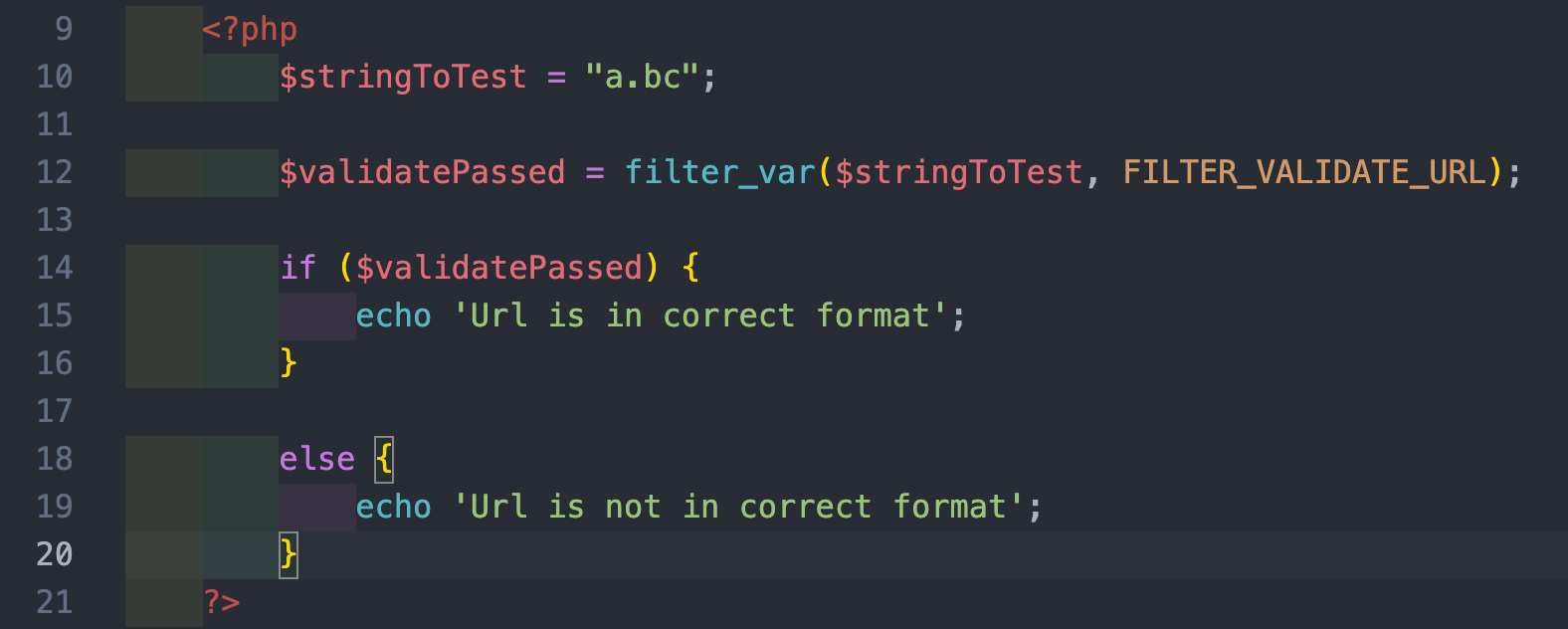
Yêu cầu xử lý validate cho form bằng PHP như sau có sử dụng Bootstrap

* Tất cả các trường không được để trống, nếu 1 trong các trường để trống thì hiển thị thông báo lỗi màu đỏ sau “Không được để trống các trường” trên đầu.
* Trường email phải có type = email, validate email
* Trường phone number chỉ cho phép nhập số
* Trường website cần có định dạng url, nếu không sẽ thông báo lỗi “Trường website cần phải có định dạng URL”.

Gợi ý: Sử dụng filter\_var để filter URL

filter\_var($stringToTest, $filterMode)

Ví dụ như sau:



🡺 In ra: “Url is not in correct format”.

Danh sách $filterMode: <https://www.php.net/manual/en/filter.filters.validate.php>

Sau khi validate form thành công, và click nút Submit, thực hiện các yêu cầu sau:

* Set value mặc đinh cho tất cả các trường chính bằng các giá trị mà user đã nhập thành công, tương tự như bài 2 ở trên.
* Hiển thị tất các thông tin mà người dùng đã nhập, theo định dạng sau:

Send contact thành công!

**Your name: <giá-trị>**

**Your email: <giá-trị>**

**Your phone number: <giá-trị>**

**Your Website: <giá-trị>**

**Your message: <giá-trị>**

## Bài tập 4: Sử dụng PHP Form để giải quyết bài toán như sau.

Trong phần này các bạn được phép sử dụng các hàm có sẵn để giải quyết bài toán sau

﻿Khi bấm nút Process nếu form chưa được điền đầy đủ, một thông báo sẽ hiển thị thông báo với người sử dụng.

- Trường first name không được để trống

- Trường last name không được để trống

- Trường address không được để trống

- Người sử dụng phải chọn ít nhất một tạp chí

- Nếu form được điền đầy đủ, một hộp thoại xác nhận được hiển thị (hình 2). Những tạp

chí (magazins) và phương thức thanh toán (payment) phải hiển thị đúng.

- Nếu sau đó nút OK được bấm, một hộp thông báo khác phải đuợc hiển thị (hình 3).

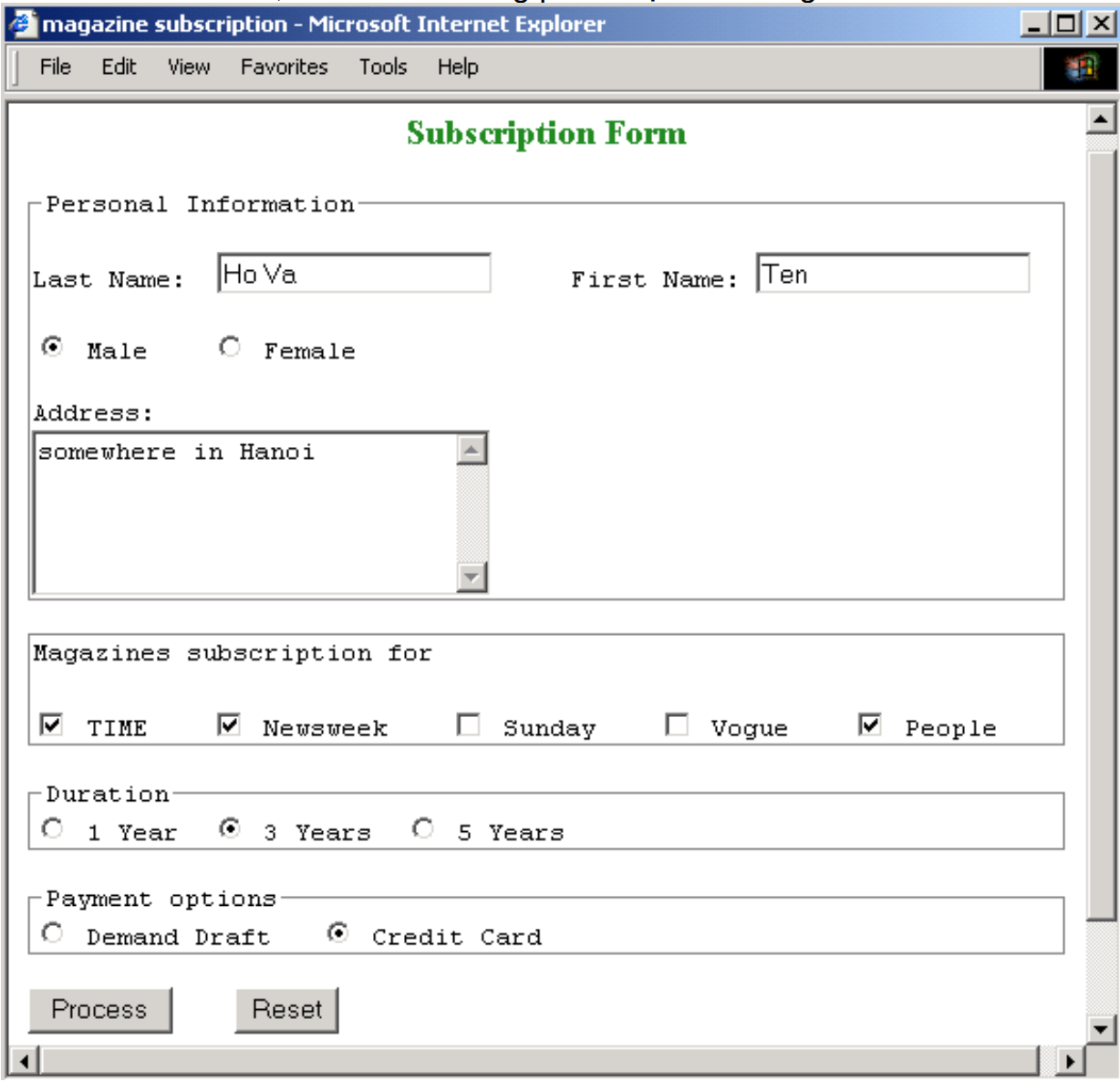
Giới tính (gender) và địa chỉ phải hiển thị đúng.

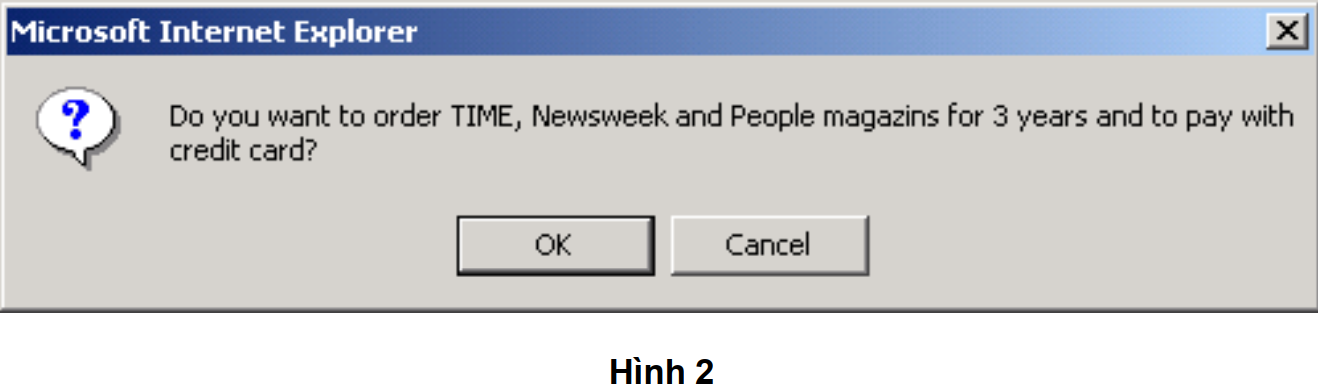
- Nếu sau đó nút Cancel được bấm thì trường last name sẽ nhận tiêu điểm (focus) và

người sử dụng có thể thay đổi sự lựa chọn của họ.

- Khi bấm nút Reset, tất cả các trường phải được xóa trắng.

Sử dụng bootstrap để format form.



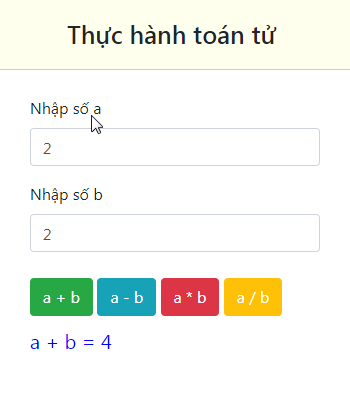




Gợi ý: 

## Bài tập 5: Sử dụng PHP để hoàn thành bài tập sau

Mục đích của bài tập này dùng để sử dụng 4 nút submit khác nhau



Yêu cầu xử lý validate với PHP cho các trường hợp sau:

* Không cho phép dữ liệu trống, hiển thị thông báo lỗi màu đỏ sau “Không được để trống số a hoặc số b” In ra “Không được phép tính toán” ở trên đầu trang và không thực hiện tính toán
* Chỉ cho phép nhập số, nếu không phải số hiển thị thông báo lỗi màu đỏ sau “Chi cho phép nhập số”.

In ra “Không được phép tính toán” ở trên đầu trang và không thực hiện tính toán

Sau khi validate thành công dữ liệu, click các phép tính và thực hiện tính toán, hiển thị kết quả như trong hình

Gợi ý:

* 4 phép tính tương đương với 4 input submit, sẽ được phân biệt qua name để biết input nào sẽ được submit

Để kiểm tra chuỗi nhập vào có phải là số hay không thì sử dụng hàm

* is\_numeric($inputToTest)

Ví dụ như sau:

* is\_numeric(‘123’) => TRUE
* is\_numeric(123) => TRUE
* is\_numeric(‘abc’) => FALSE

Gợi ý như sau



## Bài tập 6: Sử dụng Bootstrap và PHP để thực hành bài về chuỗi như sau

Sử dụng PHP để in ra số lượng nguyên âm trong một chuỗi PHP cho trước, được nhập vào từ input trong form. Nguyên âm bao gồm các kí tự “a,e,i,o,u”. Trong trường hợp chuỗi ở dưới dạng không được format, ví dụ như “hEllO TheRe”, biến đổi chuỗi thành dạng chữ thường và in ra số lượng nguyên âm trong đó.

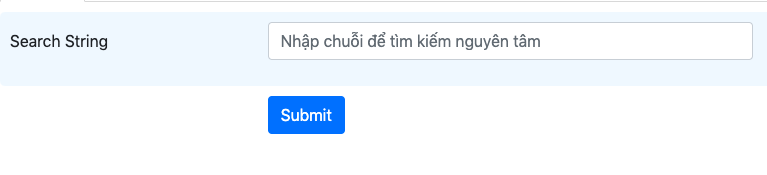


Figure 1: Mẫu giao diện

Gợi ý: Để biền đổi thành chữ thường, dùng hàm strtolower($string)

* strtolower(“hEllO TheRe”) => hello there