

Entorno Servidor

# INFORME PRÁCTICA 5

Para Jairo García

Carlos Plaza Y Jose Roberto Nuñez  
21-11-2021

Contenido

ARRAY PROVINCIAS ..... 2

FUNCIONES..... 2

FORMULARIO..... 4

VALIDACIÓN..... 5

## ARRAY PROVINCIAS

En primer lugar, deberemos crear nuestro array con las provincias (y su respectivo código postal).

```
$provincias = [];
```

Previamente nos habremos descargado un archivo txt que contenga dicha información, e introduciremos dicha información en nuestro array vacío.

```
$archivo = file_get_contents("archivo.txt");
$array_general = explode("\n", $archivo);
foreach ($array_general as $fila){
    //El limit es +2 porque tiene que ser positivo para que cuando la funcion
    //encuentre el primer espacio salte directamente y no lo haga por cada espacio que encuentre
    $item = explode(" ", $fila, +2);
    $provincias[] = [
        'numero' => $item[0],
        'provincia' => $item[1]
    ];
}
```

Leemos el archivo y lo vamos guardando por fila (cada provincia) y columna (nombre y número). Está hecho de manera que salte después de leer el campo entero y no cuando encuentre un espacio en blanco, como por ejemplo pasaría con Santa Cruz de Tenerife.

## FUNCIONES

A continuación, nos declararemos las funciones necesarias para validar los diferentes campos que tendrá nuestro formulario. Necesitaremos 6 funciones:

- Una función que compruebe si el campo introducido contiene un numero:

```
function no_numeros($string){
    $regex = "/^[A-Za-zÑñ ]/";
    return preg_match($regex, $string);
}
```

Esta función nos indicará si hay algún dígito que no sea una letra (tanto minúscula como mayúscula).

- Una función que compruebe si el campo introducido contiene una letra:

```
function no_letras($numero){
    $regex = "/^[0-9]/";
    return preg_match($regex, $numero);
}
```

Esta función nos indicará si hay algún dígito que no sea un número.

- Una función para validar una dirección de email:

```
function validar_email($string){
    $regex="/^[^0-9][a-zA-Z0-9_]+(.[a-zA-Z0-9_]+)*@[a-zA-Z0-9_]+(.[a-zA-Z0-9_]+)*.[a-zA-Z]{2,4}$/";
    //Si el preg_match devuelve un 1 significa que es valido.
    return preg_match($regex,$string);
}
```

Esta función nos indicará los requisitos necesarios de cualquier email. En caso de que devuelva 1 estará validada, y no será correcta en el caso de devolver 0.

- Una función para validar una web:

```
function validar_web($string){
    //https://didesweb.com/tutoriales-php/validar-url-php/
    $regex = "/\b(?:(:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?~_!:,.;]*[-a-z0-9+&@#\/%?~_](\.[a-z]{5}/i";
    //Si el preg_match devuelve un 1 significa que es valido.
    return preg_match($regex,$string);
}
```

Esta función nos indicará los requisitos necesarios de cualquier web. En caso de que devuelva 1 estará validada, y no será correcta en el caso de devolver 0.

- Una función para validar una contraseña con los siguientes requisitos (una mayúscula, una minúscula, un carácter especial y un número):

```
function validar_password($string){
    $regex = "/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[!@#$%^&*~_])[-a-zA-Z\d\W]{8,16}$/";
    //Si el preg_match devuelve un 1 significa que es valido.
    return preg_match($regex,$string);
}
```

Esta función nos indicará los requisitos necesarios requeridos por nuestra contraseña. En caso de que devuelva 1 estará validada, y no será correcta en el caso de devolver 0.

- Y, por último, una función que compruebe si el código postal introducido concuerda con la provincia seleccionada:

```
function cod_postal($numero){

    if(isset($_POST['guardar'])){
        $valor = $_POST['provincias'];
        //Si el value del option es distintos a los dos primeros digitos pasados pro la funcion dara un error.
        if( $valor !== substr($numero,0,2)){
            return false;
        }
        else{
            return true;
        }
    }
}
```

Esta función comprobará cuando se envíe el formulario si nuestros dos primeros dígitos del campo código postal concuerdan con el número asignado en nuestro array de provincias. En el caso de devolver true nuestro código postal será correcto.

## FORMULARIO

Después nuestro formulario con todos los campos requeridos:

```
<form action="index.php" method="POST">
<fieldset class="fieldset">
    <legend>Rellena tus datos personales</legend>
    <label for="">Nombre: </label> <input type="text" name="nombre" class="inputs_datos"/><br><br>
    <label for="">Primer Apellido: </label> <input type="text" name="apellido1" class="inputs_datos"/>
    <label for="">Segundo Apellido: </label> <input type="text" name="apellido2" class="inputs_datos"/><br><br>
    <label for="">Dirección: </label> <input type="text" name="direccion" class="inputs_datos" placeholder="c/,nº"/><br><br>
    <label for="">Ciudad: </label> <input type="text" class="inputs_datos" name="ciudad"/><br><br>
    <label for=""> Provincia: </label>
    <select class="inputs_datos" name="provincias" id="">
    <?php foreach ($provincias as $key => $row) { //Recorro el array de provincias creado anteriormente para coger los valores y la key y poder compararlo>
        <option value="<?php echo $row['numero']; ?>" ><?php echo $row['provincia']; ?></option>
    <?php } ?>
    </select>
    <label for="">Código Postal: </label> <input type="number" value="" name="cp" class="inputs_datos"/><br><br>
    <label for="">Teléfono: </label> <input type="text" name="tlf" class="inputs_datos"/><br><br>
    <label for="">Email: </label> <input type="text" name="email" class="inputs_datos">
    <label for="">Contraseña: </label> <input type="password" name="password" class="inputs_datos"><br><br>
    <label for="">Web: </label> <input type="text" name="web" class="inputs_datos"><br><br>
    <input type="submit" value="Guardar" name="guardar" class="botones"/>
<?php
```

En el input que recoge el nombre de nuestra provincia hacemos un foreach que recorre todo nuestro array provincias y nos las muestra mediante el input tipo option.

## VALIDACIÓN

En primer lugar, creamos un contador para guardar los campos que estén correctos y hacemos un isset del formulario para recoger sus valores y guardarlos.

```
$cont_aciertos = 0;
if (isset($_POST['guardar'])){//recojo los valores del formulario y los guardo en variables
    $nombre = filter_input(INPUT_POST, 'nombre', FILTER_SANITIZE_STRING);
    $apellido1 = filter_input(INPUT_POST, 'apellido1', FILTER_SANITIZE_STRING);
    $apellido2 = filter_input(INPUT_POST, 'apellido2', FILTER_SANITIZE_STRING);
    $direccion = filter_input(INPUT_POST, 'direccion', FILTER_SANITIZE_STRING);
    $ciudad = filter_input(INPUT_POST, 'ciudad', FILTER_SANITIZE_STRING);
    $provincia = filter_input(INPUT_POST, 'provincia', FILTER_SANITIZE_STRING);
    $cp = filter_input(INPUT_POST, 'cp', FILTER_SANITIZE_STRING);
    $tlf = filter_input(INPUT_POST, 'tlf', FILTER_SANITIZE_STRING);
    $email = filter_input(INPUT_POST, 'email', FILTER_SANITIZE_STRING);
    $password = filter_input(INPUT_POST, 'password', FILTER_SANITIZE_STRING);
    $web = filter_input(INPUT_POST, 'web', FILTER_SANITIZE_STRING);
```

A continuación, validaremos nuestro formulario mediante las funciones que hemos realizado:

```
if((no_numeros($nombre) == 0) && !empty($nombre)) {//compruebo si nombre lleva numeros
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu nombre es correcto</p>";
    //Sumamos en uno los aciertos
    $cont_aciertos = $cont_aciertos +1;
}else{
    echo "<p class='errores'> <img class='img' src='img/cerrar.png' alt='error'>Error el nombre solo puede llevar letras<br/></p>";
}
if((no_numeros($apellido1) == 0) && !empty($apellido1)) {//compruebo si apellido lleva numeros
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tus apellidos es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}else{
    echo "<p class='errores'><img class='img' src='img/cerrar.png' alt='error'>Error el primer apellido solo puede llevar letras<br/></p>";
}
if(no_numeros($apellido2) == 0) {//compruebo si apellido lleva numeros
    //echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tus apellidos son correcto</p>";
}else{
    echo "<p class='errores'> <img class='img' src='img/cerrar.png' alt='error'>Error el segundo apellido solo puede llevar letras<br/></p>";
}
if((no_numeros($ciudad)==0) && !empty($ciudad)) {//compruebo si ciudad lleva numeros
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu ciudad es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}else{
    echo "<p class='errores'> <img class='img' src='img/cerrar.png' alt='error'>Error la ciudad no puede llevar numeros<br/></div>";
}
}
```

Comprobamos los campos que únicamente deben tener letras, todos funcionan de la misma manera. Si nos devuelve 0, es decir no contiene algo que no sea una letra, y su valor no es vacío hará un echo de la imagen que es un check verde y nos sumará uno a nuestro contador de aciertos. En el caso contrario hará un echo de la imagen que es una x roja dando a entender que el campo introducido no es correcto.

Ahora pasamos a validar los campos que no deben llevar letras:

```
if(no_letras($tlf)==0 && !empty($tlf) && strlen($tlf) == 9) { //compruebo si telefono lleva letras
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu numero de teléfono es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}else{
    echo "<p class='errores'> <img class='img' src='img/errar.png' alt='error'>El numero de telefono es incorrecto<br/></div>";
}

if(no_letras($cp) == 0 && !empty($cp) && strlen($cp) == 5 && cod_postal($cp)){ //compruebo si codigo postal lleva letras

    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu codigo postal es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}else{
    echo "<p class = 'errores'> <img class='img' src='img/errar.png' alt='check'>Tu código postal es incorrecto</p>";
}
}
```

Estos campos funcionan exactamente igual que los anteriores con la diferencia que en ambos tenemos la condición de que deben llevar un número de dígitos concreto. Esto se soluciona añadiendo la condición de que `strlen(campo) == número de dígitos requerido`.

Los campos de web, email y contraseña funcionan igual:

```
if((validar_email($email) == 0) && empty($email) ) { //compruebo si la estructura del email es correcta
    echo "<p class='errores'> <img class='img' src='img/errar.png' alt='error'>Error el email no es válido<br/></p>";

}else{
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu Email es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}

if((validar_password($password) == 0 && empty($password) ) ) { //compruebo si contraseña es correcta(mayuscula,minuscula,numero,caracter especial,longitud de 8-16
    echo "<p class='errores'> <img class='img' src='img/errar.png' alt='error'> La contraseña debe tener: entre 8 y 16 caracteres, Al menos un número, Al menos

}else{
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu contraseña es correcto</p>";
    $cont_aciertos = $cont_aciertos +1;
}

if(validar_web($web) == 0 && empty($web) ) { //comprueba que la estructura de la web sea correcta
    echo "<p class='errores'> <img class='img' src='img/errar.png' alt='error'>Error la web no es válida</p>";

}else{
    echo "<p class = 'errores'> <img class='img' src='img/cheque.png' alt='check'>Tu web es correcta</p>";
    $cont_aciertos = $cont_aciertos +1;
}
}
```

Lo último que quedaría por realizar es lo siguiente:

```
<div class="pop" style="z-index:<?php if($cont_aciertos == 8){ //Si el contador de aciertos es 8 significa que estan todos los inputs validados
    echo 2;} //y al z-index de este div lo modificamos en 2 para que se muestre por delante del form
    if (isset($_GET['volver'])) { echo -1; } //Si el usuario pulsa volver el z-index volverá a -1 para que se oculte?>>">
    
    <p>Formulario enviado correctamente.</p>
    <input type="submit" value="Volver" name="volver" class="botones">
</div>
<?php
    if(isset($_GET['volver'])){
        //Si el usuario pulsa volver se vuelve a cargar la pagina
        header("Location:index.php");
    }
?>
```

En caso de que el contador sea 8 (todos los campos están introducidos correctamente) nos aparecerá un div con la imagen del check verde en grande diciendo que nuestro formulario se ha enviado correctamente. Esto sucede ya que el div está oculto y cambiamos su valor para que aparezca delante y sea visible.

En el caso de darle a volver el div volvería a estar oculto.