

Ambientes no propietarios

Ejercicios

Edwin Salvador

27 de octubre de 2015

Sesión 5

Contenido I

1 Instrucciones

2 Ejercicios

- Los ejercicios planteados a continuación deben ser realizados de manera individual.
- Deben ser subidos en el repositorio Github **hasta el día 6 de noviembre de 2015 23:59.**
- Deben estar dentro de la carpeta deberes/deber5. Organizados en carpetas ejercicio1, ejercicio2, ejercicio3, etc.
- Se recomienda ejecutar cada uno de los ejemplos propuestos para lograr entender el funcionamiento del código para luego pasar a implementar los ejercicios.
- Cualquier duda deben escribir a mi correo y de ser necesario podemos acordar una cita en mi oficina para resolverla.

Contenido I

1 Instrucciones

2 Ejercicios

Ejercicio 1

Formularios

Escribir un formulario que permita realizar la conversión de dólares a euros o a pesos colombianos. El usuario puede elegir solo una moneda a la que desea convertir. El formulario debe contener:

- Un Campo de texto para ingresar el valor en dólares que se desea convertir.
- Un elemento select o dos radios que liste las dos opciones de monedas a las que se puede convertir.
- Un botón que envíe los datos al servidor para realizar la conversión y luego se presente el resultado bajo el formulario.
- La siguiente figura muestra un mockup del ejercicio.

Mockup Ejercicio 1

Ejercicio1 - Conversor de Monedas

← → ✕ 🏠 🔍

Conversor de monedas

Valor en USD: Convertir a:

Seleccione... ▼
Euros
Pesos Colombiano

Resultado:

1 USD equivale a 0.91 euros

⏏

Ejercicio 2

Arrays

Hemos hablado brevemente sobre los arrays en PHP. Los hemos utilizado en el manejo de formularios a través de la variable superglobal `$_POST` y cuando recuperamos información desde MySQL. La sintaxis para definir un array es la siguiente:

```
$array = array(  
    "indice1" => "valor1",  
    "indice2" => "valor2",  
);
```

Los índices en el ejemplo son: "indice1" e "indice2". Los índices de un array en PHP pueden ser strings o enteros. Los valores pueden ser de cualquier de cualquier tipo, incluyendo otro array y objetos. Para mayor referencias sobre los arrays pueden ir a

<http://php.net/manual/es/language.types.array.php>

Ejercicio 2

Arrays Multidimensionales

```
$array = array(  
    "indice1" => array(  
        "subindice1" => "sub valor 1",  
        "subindice2" => 2  
    ),  
    "indice2" => true,  
);
```

El ejemplo muestra un array bidimensional ya que el valor del “indice1” es otro array (dos dimensiones de índices).

También podemos observar la flexibilidad que ofrecen los arrays al poder combinar varios tipos de valores en diferentes índices.

Arrays

foreach

PHP ofrece una estructura de control para iterar fácilmente sobre cualquier arreglo llamada `foreach`. Tiene la siguiente sintaxis:

```
// sintaxis 1  
foreach (expresion_array as $valor)  
    sentencias  
  
// sintaxis 2  
foreach (expresion_array as $indice => $valor)  
    sentencias
```

Donde `expresion_array` puede ser una variable o directamente un array. La `sintaxis 1` va recorriendo el array y va asignando el valor correspondiente a `$valor`.

La `sintaxis 2` además asigna el índice correspondiente a `$indice`. A continuación se muestra un ejemplo del uso del `foreach`.

Arrays

Ejemplo foreach

Imprimir los valores del vector asociativo siguiente usando la estructura de control foreach:

```
$v[1]=90;  
$v[30]=7;  
$v['e']=99;  
$v['hola']=43;
```

Solución:

```
<html>  
  <head>  
    <title>Ejemplo</title>  
  </head>  
  <body>  
    <?php  
      $v[1]=90;  
      $v[30]=7;  
      $v['e']=99;  
      $v['hola']=43;  
      foreach ($v as $indice => $valor) {  
        echo "El elemento de indice $indice vale $valor <br>";  
      }  
    ?>  
  </body>  
</html>
```

Ejercicio 2

Arreglos Multidimensionales

El archivo `provincias_cantones_parroquias.php` contiene un arreglo que contiene todas las provincias del Ecuador con sus respectivos cantones y cada cantón con sus respectivas parroquias. La estructura de este arreglo es la siguiente:

```
$provincias_cantones_parroquias = array (
    1 =>
        array (
            'provincia' => 'AZUAY',
            'cantones' =>
                array (
                    101 =>
                        array (
                            'cantón' => 'CUENCA',
                            'parroquias' =>
                                array (
                                    10101 => 'BELLAVISTA',
                                    10102 => 'CANARIBAMBA',
                                    10103 => 'EL BATAN',
                                    10104 => 'EL SAGRARIO',
                                    10105 => 'EL VECINO',
                                    ...
                                )
                            )
                        )
                )
        )
    )
```

Donde:

- Los índices del primer nivel del array corresponden al código de las provincias. Ej: El índice 1 es el código para la provincia 'AZUAY'.
- Cada provincia tiene un item 'cantones' que corresponde a la lista de sus cantones. Cada índice del array 'cantones' corresponde al código del cantón. Ej: el índice 101 es el código para el cantón 'CUENCA'.
- Cada cantón tiene un item 'parroquias' que corresponde a la lista de sus parroquias. Cada índice del array 'parroquias' corresponde al código de la parroquia. Ej: el índice 10101 es el código para la parroquia 'BELLAVISTA'.

Ejercicio 2

Arreglos Multidimensionales

- Este ejercicio consiste en implementar un formulario web que simule el comportamiento del siguiente: http://130.56.248.115/anp/sesion5/ejercicio2/provincias_cantones_parroquias.php
- En este formulario, las listas de cantones y parroquias son dinámicos, es decir, la lista de cantones se carga dependiendo de la provincia que se seleccione y la lista de parroquias se carga dependiendo del cantón que se seleccione. **Las listas son cargadas únicamente al presionar el botón enviar.** En las siguientes clases veremos como cargar estas listas sin necesidad de recargar la página (utilizando Ajax).

Ejercicio 2

Arreglos Multidimensionales

- **PISTA** Las opciones (`<option>`) de los elementos `select` deben tener como atributo `value` el código de la provincia, cantón o parroquia según corresponda.
- **PISTA** Se debe enviar los datos del formulario y mediante la variable `$_POST` determinar si se ha seleccionado alguna provincia y determinar su código. Tener en cuenta que cuando el formulario se recarga, los elementos del formulario siguen seleccionados.
- **PISTA** Para facilitar la realización de este ejercicio se recomienda utilizar la sintaxis 2 del `foreach`, así podemos obtener el código y el nombre de las provincias, cantones o parroquias según sea el caso.

Ejercicio 3

MySQL

En este ejercicio se debe hacer que los datos del formulario anterior se guarden en una BDD llamada s5e3 en una tabla llamada datos. Los datos deben guardarse únicamente cuando se ha seleccionado un item de las tres listas.

Se debe subir el código del ejercicio incluyendo un archivo SQL con la BDD.