

Trabajo Práctico N° 1

En este primer trabajo práctico buscaremos afianzar los conocimientos adquiridos sobre los conceptos básicos del lenguaje de programación PHP durante el primer capítulo de nuestro curso.

Objetivo: Comprender y profundizar los conocimientos sobre el funcionamiento de PHP como lenguaje de programación utilizando variables, funciones y estructuras de control.

Introducción: En nuestra actualidad vemos que el desarrollo de Internet ha sido inminente y con ello las aplicaciones Web, por lo tanto, se hace indispensable el uso de un lenguaje que permita desarrollar aplicaciones Web como PHP, entre otros. Teniendo en cuenta la situación anterior es que veremos la como desarrollar aplicaciones Web que se ejecutarán del lado del servidor utilizando uno de los mejores lenguajes del ambiente del Software Libre.

Puntos a realizar en la entrega:

1. Función para validar si un año es bisiesto.
2. Función para validar si una fecha es correcta.
3. Función para calcular la edad de una persona.
4. Función para procesar la información personal de un usuario.

Desarrollo

1. Función para validar si un año es bisiesto.

Se deberá crear una función para determinar si el año recibido como parámetro corresponde a un año bisiesto o no retornando un valor de tipo “boolean” determinando la respuesta. Se debe tener en consideración la siguiente regla para calcularlo:

Un año es bisiesto si es divisible por 4, a menos que sea divisible por 100. Sin embargo, si un año es divisible por 100 y además es divisible por 400, también resulta bisiesto.

2. Función para validar si una fecha es correcta.

La función que se creará recibirá como parámetro la fecha dentro de un tipo de dato “string” y retornará un valor del tipo “boolean” determinando la respuesta. Se determina si una fecha es correcta partiendo del formato dd/mm/aaaa (ej. 03/04/2014) y teniendo en cuenta las siguientes reglas:

- Deberá contener 10 caracteres.
- Debe estar compuesta por 3 partes separadas por “/”.
- Ninguno de los campos que la componen puede ser menor a 1.
- Los días no pueden exceder el mayor valor para el correspondiente mes, teniendo también en cuenta si un año es o no bisiesto.
- El mes no puede ser menor a 1 y mayor a 12.

3. Función para calcular la edad de una persona.

Partiendo de que la fecha recibida por parámetro debe ser una fecha válida, y que no puede ser una fecha posterior a la fecha actual, se deberá calcular la edad de una persona en base a la fecha recibida como parámetro de la función.

4. Función para procesar la información personal de un usuario.

Esta función recibirá un arreglo y deberá generar un valor del tipo “string” detallando la la información del mismo. El formato que deberá tener el string será el siguiente:

[Apellido], [Nombre] de [edad] años de edad y documento ([TIPO DE DOCUMENTO]) [N° Documento] nacido en la fecha [fecha de nacimiento].

Cada uno de los campos (determinados entre corchetes) deberá ser reemplazado por su correspondiente valor obtenido del array recibido como parámetro cuya estructura será:

```
array(  
    'apellido' => 'Matusalen',  
    'nombre' => 'Jonas',  
    'fecha_nacimiento' => '29/02/1900',  
    'tipo_documento' => 'le',  
    'numero_documento' => '1.369.851',  
)
```

Las consideraciones que se deberán tener en cuenta al momento de completar los campos del valor a retornar son:

- El nombre y apellido no deberán tener espacios en blanco extras ni al principio ni al final, y deberán estar todo en minúsculas excepto la primera letra.
- La edad se deberá calcular en base a la fecha de nacimiento obtenida. En el caso de no ser válida el valor que se deberá mostrar será dos guiones "--".
- El tipo de documento deberá estar todo en mayúsculas.
- La fecha de nacimiento en el caso de no ser correcta, deberá ser reemplazada por el texto **"(Fecha no válida)"**.

Prueba del Proceso

Para probar si el proceso podremos utilizar el siguiente arreglo como ejemplo práctico:

```
array(
  [0] => array('apellido' => 'Romero', 'nombre' => 'Fernando',
    'tipo_documento' => 'dni',
    'numero_documento' => 42601837,
    'fecha_nacimiento' =>
      '13/07/2000'),
  [1] => array('apellido' => 'Febre ', 'nombre' => 'Yolanda Rosa ',
    'tipo_documento' => 'dni',
    'numero_documento' =>
      '10587664',
    'fecha_nacimiento' => '15/11/1948'),
  [2] => array('apellido' => 'Gomez', 'nombre' => 'IRMA',
    'tipo_documento' => 'lc',
    'numero_documento' => '5362812',
    'fecha_nacimiento' => '22/05/1918'),
  [3] => array('apellido' => 'Lissa ', 'nombre' => 'Romina ',
    'tipo_documento' => 'dni',
    'numero_documento' => '12385035',
    'fecha_nacimiento' => '13/05/1958'),
  [4] => array('apellido' => 'Friz', 'nombre' => 'Carlos Rolando',
    'tipo_documento' => 'dni',
    'numero_documento' => '14128874',
    'fecha_nacimiento' => '01/05/1961'),
  [5] => array('apellido' => 'Garzia', 'nombre' => 'Marta Camila',
    'tipo_documento' => 'dni',
    'numero_documento' => '17121841',
    'fecha_nacimiento' => '23/11/1964'),
  [6] => array('apellido' => 'Soñez', 'nombre' => 'Jorge Gonzalo',
    'tipo_documento' => 'dni',
    'numero_documento' => '13695121',
    'fecha_nacimiento' => '09/08/1959'),
  [7] => array('apellido' => 'PEREZ', 'nombre' => 'GISELA YANINA',
    'tipo_documento' => 'dni',
    'numero_documento' => '31486546',
    'fecha_nacimiento' => '25/03/1985'),
  [8] => array('apellido' => 'YNZA', 'nombre' => 'CARLA',
    'tipo_documento' => 'dni',
    'numero_documento' => '28302680',
    'fecha_nacimiento' => '30/01/1981'),
  [9] => array('apellido' => 'Martinez', 'nombre' => 'Antonia',
    'tipo_documento' => 'dni',
    'numero_documento' => '38054607',
    'fecha_nacimiento' => '27/04/1994')
);
```



Entrega

La primer entrega deberá ser en un solo script y deberá ser cargado a la plataforma como un archivo comprimido en formato *.rar o *.tar.gz.