

# QuickStart: Un tutorial de HTML\_QuickForm.

**Encuentro Nacional de Linux 2005.  
Benemerita Universidad Autonoma de Puebla.**

**Octubre 21, 2005. Puebla, Pue.**

**Farid Alfredo Bielma Lopez.**

**[fbielma@fbielma.org](mailto:fbielma@fbielma.org)**

**[http://fbielma.org/conferencia/enli/HTML\\_QuickForm.pdf](http://fbielma.org/conferencia/enli/HTML_QuickForm.pdf)**

# PEAR :: HTML\_QuickForm

HTML\_QuickForm es una clase PEAR. El Repositorio PEAR es una colección para agregar librerías y módulos para PHP.

HTML\_QuickForm hace el trabajo con formularios de manera más fácil, como reservar valores enviados, muestra mensajes de error, agrega una vista consistente para los elementos del Formulario y simplifica las capacidades de File Uploads.

# Instalacion.

Las instrucciones generales de instalación puede encontrarla en el Sitio Oficial:

[http://pear.php.net/package/HTML\\_QuickForm](http://pear.php.net/package/HTML_QuickForm)

En Unix la instalación se realiza:

```
# pear install -a HTML_QuickForm.version.tgz
```

# Hello a HTML\_QuickForm.

```
<?
```

```
Require 'HTML/QuickForm.php';
```

```
//Se crea el contenedor llamado $form
```

```
$form=new HTML_QuickForm('nombre');
```

```
//Se agregan elementos al formulario:
```

```
$form-> addElement('text','nombre','Tu nombre:');
```

```
$form-> addElement('submit','enviar','Enviar');
```

# Hello a HTML\_QuickForm.

```
// Se llama a la Funcion process()
```

```
If($form-> validate()){  
    $form->process('enli');  
}  
Else{  
    $form->display();  
}
```

# Hello a HTML\_QuickForm.

```
// El metodo process() llama a la funcion enli, la cual recoge  
// los datos del formulario como array.
```

```
function enli($data){  
    echo $data['nombre'].', recibe un saludo de el enli';  
}
```

# Manipulacion de los Elementos.

Usando el metodo addElement()

```
$elemento = & form-> addElement('text','nombre','Tu nombre');
```

En el ejemplo se retorna un objeto por referencia para hacer llamadas al metodo.

```
$elemento = & form-> addElement('text','nombre','Tu nombre');
```

```
$elemento -> SetSize(10);
```

```
$elemento -> SetMaxlength(30);
```

# Elementos.

Los tipos de elementos soportados por HTML\_QuickForm son:

text	autocomplete	password
textarea	hidden	select
radio	submit	reset
button	image	file
header	link	html



# Elemento: password

El elemento password tiene una funcionalidad semejante al elemento text, solo que los datos ingresados son mostrados como simbolos o con asteriscos.

El elemento password produce la tag HTML:

```
<input type="password">
```

# Elemento: textarea

El elemento textarea produce entradas en multiples lineas en un area mostrada por la tag HTML `<textarea></textarea>`

Metodos:

`setRows()` El atributo controla cuantas lineas de texto estaran visible en el area.

`setCols()` El atributo controla el ancho de la textarea.

# Elemento: hidden

El elemento hidden produce un tag: `<input type="hidden">`. El navegador no muestra los datos ocultos en el formulario.

Ejemplo:

```
$form-> addElement('hidden', 'code',96546);
```

# Elemento: select

El elemento select produce una caja de selección, se pueden pasar datos como array:

```
$subjects = array('Math','Ice Fishing','Anatomy');  
$form->addElement('select','subject','Subject(s): ', $subjects);
```

# Elemento: radio

El elemento radio produce un tag HTML `<input type="radio">`. El navegador permite solamente que escojas un boton de radio en un tiempo.

```
$form->addElement('radio','when','Lenguaje:','Perl','perl');  
$form->addElement('radio','when',null,'PHP','php');  
$form->addElement('radio','when',null,'Java','java');
```

# Elemento: button

Este elemento es identico al submit, con la diferencia que el elemento button no tiene una accion por default asociada al elemento.

Ejemplo:

```
$form->addElement('button','check',  
Clic aqui,'onClick="checkPage();"');
```

# Elemento: file

El elemento produce un `<input type="file">`. Muestra una caja de texto y un botón que cuando lo clickeas, abre un cuadro de diálogo para la selección del archivo.

Ejemplo de uso:

```
$form->addElement('file', 'data', 'Tu archivo');
```

# Agregando reglas de validacion.

El método `addRule()` agrega reglas de validacion, para un elemento

especifico del Formulario.

```
$form->addElement('text','nombre','Nombre: ');
```

```
//Agrega la regla "required" al elemento nombre
```

```
$form->addRule('nombre','Teclea tu nombre.', 'required');
```

```
//Agrega la regla "minlength" al elemento nombre
```

```
$form->addRule('nombre','Mayor a 3 letras.', 'minlength',3);
```



# Agregando reglas de validacion.

required		La Entrada de datos es requerido para el elemento.
Maxlength	\$length	El dato puede tener \$length caracteres como máximo.
Minlength	\$length	Los caracteres que puede tener como mínimo el elemento.
Email		El dato debe tener sintacticamente una dirección de correo válida.
Alphanumeric		El elemento puede contener solamente letras y números.

# Agregando reglas de validacion.

Numeric		El elemento debe contener solamente un entero valido o un numero decimal.
Uploadedfile		El elemento debe contener un archivo exitosamente cargado al servidor.
maxfilesize	\$size	El archivo debe tener no mas de \$size tamaño permitido para subirlo al servidor.
filename	\$file_rx	El archivo cargado debe tener un nombre de acuerdo a la expresion: \$file_rx

# File Uploads.

```
require 'HTML/QuickForm.php';  
$form = new HTML_QuickForm();  
$form->addElement('file','uploaded_file','Tu archivo:');  
//Archivo menor o igual a 128kbytes  
$max_size = 131072;  
// Estar seguro que se subio al Servidor.  
$form->addRule('uploaded_file','Por favor, sube el archivo','uploadedfile');  
$form->addRule('uploaded_file','Demasiado grande','maxfilesize',$max_size);  
$form->addRule('uploaded_file','Enviar solamente HTML ','mimetype','text/html');  
$form->addElement('submit','save','Enviar archivo');  
  
//No esta permitido subir un archivo mayor que 128k  
$form->setMaxFileSize($max_size);
```

# File Uploads.

```
if ($form->validate()) {  
    $form->process('Mover');  
} else {  
    $form->display();  
}  
function Mover($data) {  
    global $form;  
    $new_name = strtr($data['uploaded_file']['name'],'\','');  
    $new_name = str_replace('.',",",$new_name);  
    $file =& $form->getElement('uploaded_file');  
    if ($file->moveUploadedFile('/tmp',$new_name)) {  
        print "El archivo ha sido cargado en /tmp/$new_name."  
    }  
    ?>
```

# Preguntas...

Farid Alfredo Bielma Lopez

Email: [fbielma@fbielma.org](mailto:fbielma@fbielma.org)

MSN: [fbielma@hotmail.com](mailto:fbielma@hotmail.com)

[http://fbielma.org/conferencia/enli/HTML\\_QuickForm.pdf](http://fbielma.org/conferencia/enli/HTML_QuickForm.pdf)