

HTML_QuickForm para el procesamiento de Formularios en PHP.

CONAIS 2005.

Universidad Juarez Autonoma de Tabasco.

Septiembre 1, 2005. Villahermosa, Tab.

Farid Alfredo Bielma Lopez.

fbielma@fbielma.org

<http://fbielma.org/conferencia/conais/QuickForm.pdf>



Agenda.

- Una rápida introducción a PEAR.
- Funcionalidades provistas por PEAR.
- Que es HTML_QuickForm.
- Instalación.
- Usando HTML_QuickForm.
- Recursos en línea.
- Preguntas y Respuestas.



Que es PEAR.

PEAR is short for "PHP Extension and Application Repository". The purpose of PEAR is to provide:

- A structured library of open-sourced code for PHP users
- A system for code distribution and package maintenance
- A standard style for code written in PHP, specified here
- The PHP Extension Community Library (PECL).
- A web site, mailing lists and download mirrors to support the PHP/PEAR community.



Introducción a PEAR.

PEAR, es una colección de clases en PHP para agregar nuevas funcionalidades al lenguaje, como por ejemplo: conectar con un servidor SOAP, Abstracción de Base de Datos, Encriptación, manejo y validacion de formularios.

El Manejador de Paquetes de PEAR, es un programa en linea de comandos llamado pear, ayuda a instalar, actualizar y eliminar paquetes PEAR.

pear

build Build an Extension From C Source

list List Installed Package

install Install Packege

...

...



Funcionalidades.

Paquetes PEAR.

HTML_QuickForm

DB

ADODB

Smarty

Auth_HTTP

XML-RPC

Mail-mime

Net_SMTP



HTML_QuickForm.

HTML_QuickForm es un modulo PEAR. Con HTML_QuickForm se realiza el trabajo con formularios HTML de manera mas sencilla, en vez de mostrar los elementos uno por uno, el método define un contenedor de un formulario en donde se puede mostrar todos los elementos de una sola vez.

HTML_QuickForm automáticamente reserva los valores enviados del formulario, muestra mensajes de error, realiza checks de validación de datos y simplifica las capacidades de File Uploads.



Instalación.

Download HTML_QuickForm

http://pear.php.net/package/HTML_QuickForm

Unix.

```
#pear install -a HTML_QuickForm
```

Windows.

```
C:\php> pear install -a HTML_QuickForm
```



Usando QuickForm.

Pasos para usar HTML_QuickForm.

- Comprobar que QuickForm esta instalado.
- Cargar el codigo QuickForm: `require 'HTML/QuickForm.php';`
- Crear un nuevo objeto de Formulario.
- Agregar Elementos
- Validar/procesar/mostrar el Formulario



Un simple Form.

```
<?php
require 'HTML/QuickForm.php';

$form = new HTML_QuickForm();
$form->addElement('text','nombre','Nombre: ');
$form->addElement('submit','send','Enviar Saludos!!!');

if ($form->validate()) {
    $form->process('saludos');
} else {
    $form->display();
}

function saludos($data) {
    print 'Bienvenido, ' . $data['nombre']. ' a HTML_QuickForm' ;
}
?>
```



Agregando Elementos.

El método *addElement()*;

```
$form->addElement('text', 'nombre', 'Nombre:');  
$form->addElement('password', 'pwd', 'Clave:');  
$form->addElement('submit', 'send', 'Enviar');
```



Agregando Elementos.

Tipos de Elementos QuickForm:

Text	Textarea	Radio	Button
advcheckbox	Header	Autocomplete	Hidden
Submit	Reset	Select	Password
File	Checkbox	Link	image



El método validate().

La logica de la Aplicación:

```
if ($form->validate()) {  
    $form->process('saludos');  
} else {  
    $form->display();  
}
```

Mostrando el Formulario antes de procesar los datos:

```
$valid = $form->validate();  
    $form->display();  
if ($valid) {  
    $form->process('consulta_sql');  
}
```



Procesando el Form.

Usando un llamada a una funcion:

```
$form->process('saludos');
```

```
/* ... */
```

```
function saludos($data) {  
    print 'Bienvenido, ' . $data['nombre']. ' a HTML_QuickForm';  
}
```

La función saludos() es invocada por el método process(), pasando un array de los datos enviados del Formulario.



Validacion de Forms.

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','Tu nombre debe tener mas de 3  
letras.', 'minlength',3);
```



El método addRule().

```
<?php

require 'HTML/QuickForm.php';
$form = new HTML_QuickForm();
$form->addElement('text','nombre','Nombre: ');
$form->addElement('submit','send','Enviar Saludos!!!');
$form->addRule('nombre','Teclea tu nombre.', 'required');
$form->addRule('nombre',' Tu nombre debe tener mas de 3 letras ',
    'minlength',3);

if ($form->validate()) {
    $form->process('saludos');
} else {
    $form->display();
}

function saludos($data) {
    print 'Bienvenido, ' . $data['nombre']. ' a HTML_QuickForm' ;
}
?>
```



Reglas de Validación.

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



File Uploads.

```
<?php
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;
    $new_name = strstr($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.";
    } else {
        print "Error de procesamiento";
    }
}
?>
```



Referencias.

1. PEAR Manual: HTML_QuickForm.

<http://pear.php.net/manual/en/package.html.html-quickform.php>

2. Getting Started with PHP's HTML_QuickForm.

www.onlamp.com/pub/a/php/2004/07/22/html_quickform.html?page=2

3. PEAR HTML_QuickForm Getting Started Guide.

<http://www.thelinuxconsultancy.co.uk/quickform.php>

4. Using HTML_QuickForm to Manage WebForms.

<http://www.devarticles.com/c/a/Web-Design-Usability/Using-HTMLQuickForm-To-Manage-Web-Forms-Part-1/>



Preguntas y Respuestas.

<http://fbielma.org/conferencia/conais/QuickForm.pdf>