

INTRODUCCIÓN A LA PROGRAMACIÓN CON PHP

PRÁCTICA Nº 5

Información extraída de:

<http://ar.php.net/manual/es/langref.php>

Envío de Emails

Para el envío de correos electrónicos utilizando PHP se dispone de una función bastante potente, incluida en PHP, sin necesidad de instalar ningún añadido.

En PHP disponemos de una función llamada **mail()** que permite configurar y enviar el mensaje de correo.

```
bool mail ( string $to , string $subject , string $message [, string  
    $additional_headers [, string $additional_parameters ] ] )
```

Envío de Emails

Parámetros necesarios en todos los casos

Destinatario: la dirección de correo o direcciones de correo que han de recibir el mensaje. Si incluimos varias direcciones debemos separarlas por una coma.

Asunto: para indicar una cadena de caracteres que queremos que sea el asunto del correo electrónico a enviar.

Cuerpo: el cuerpo del mensaje, lo que queremos que tenga escrito el correo.

Envío de Emails

Ejemplo de envío de un mail sencillo

```
<?  
mail("pepito@desarrolloweb.com,  
    maria@guiartemultimedia.com",  
    "asunto",  
    "Este es el cuerpo del mensaje")  
?>
```

Envío de Emails

Parámetros opcionales del envío de correo

Headers: Cabeceras del correo. Datos como la dirección de respuesta, las posibles direcciones que recibirán copia del mensaje, las direcciones que recibirán copia oculta, si el correo está en formato HTML, etc.

Additional_parameters: esta opción no suele utilizarse y, además, sólo está disponible a partir de la versión PHP 4.0.5 y desde PHP 4.2.3 está deshabilitado en modo seguro. Puede usarse para pasar parámetros adicionales al programa configurado para enviar el correo.

Ejemplo

- Ejemplo

```
<?php
```

```
$destinatario = "xx@xx.com ";
```

```
$asunto = "Prueba";
```

```
$cuerpo = "Esto es una prueba de envío de  
correo a través del servidor";
```

```
mail($destinatario,$asunto,$cuerpo)
```

```
?>
```

Ejercicio :

Escribir un script en PHP para poder enviar un correo electrónico, con formato HTML, a través del servidor

Explicando algunas características del código...

- ✓ Cuando el servidor envíe un documento a un agente de usuario (por ejemplo, un explorador), también enviará información en el campo Content-Type del encabezado adjunto de HTTP sobre el tipo de formato de datos del que se trata. Esta información se expresa mediante una etiqueta del tipo MIME.
- ✓ Parámetro CHARSET: los documentos transmitidos con HTTP que son de tipo texto, como text/html pueden enviar un **parámetro CHARSET** en el encabezado de HTTP para especificar la codificación de caracteres del documento.
- ✓ HTTP 1.1 especifica que el **charset** predeterminado es ISO-8859-1.

✓ **MIME**

(Multipurpose Internet Mail Extensions – “Extensiones de Correo Internet Multipropósito”).

Serie de especificaciones dirigidas al intercambio transparente de todo tipo de archivos a través de Internet.

MIME se empleó originalmente para describir el contenido de los e-mails, pero luego se extendió para describir cualquier tipo de contenido.

Permite mejorar la posibilidad de transferencia de texto entre distintos idiomas y alfabetos en el envío de emails, y también sirve para el envío de archivos a través del protocolo HTTP.

SMTP: es un protocolo de red utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

SMTP: Simple Mail Transfer Protocol, «*Protocolo para la transferencia simple de correo electrónico*».

Es un protocolo de la capa de aplicación basado en texto, utilizados para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

QK SMTP Server es un smtp server, es un software potente pero fácil de usar, que permite enviar correos electrónicos desde el host local a los buzones de los destinatarios.

Llegue o no el correo se obtendrá retroalimentación instantáneamente

```

<?php
$destinatario = "xx@xx.com ";
$asunto = "Prueba";
$scuerpo = '
<html>
<head>
    <title>Envio de mail</title>
</head>
<body>
<h1>Hola!</h1>
<p>
<b>Esto es una prueba</b>.
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
x
</p>
</body>
</html>
';
// Para enviar un correo HTML
//debe establecerse la cabe-
//cera Content-type
$headers = "MIME-Version:
1.0\r\n";

$headers .= "Content-type:
text/html; charset=iso-8859-
1\r\n";
//dirección del remitente concatenada con
//otras direcciones
$headers .= "From: NN
<nn@nn.com>\r\n";
//dirección de respuesta distinta que la del
//remitente
$headers .= "Reply-To:
ss@ss.com\r\n";
//direcciones que reciben copia
$headers .= "Cc:
yy@yy.com\r\n";

//direcciones que reciben copia oculta
$headers .= "Bcc: zz@zz.com,
pp@pp.com\r\n";

mail($destinatario,$asunto,$scue
rpo,$headers)
?>

```

Formulario.html

```
<body>
```

```
  <form action="enviar.php" method="post">
```

```
    <fieldset>
```

```
    <p><label> Nombre <input type="text" name="nombre"
size="25" /> </label></p>
```

```
    <p><label> Email <input type="text" name="email"
size="25" /> </label></p>
```

```
    Comentario
```

```
    <p><label><textarea name="texto" cols="32"
rows="6"></textarea></label></p>
```

```
    <input type="submit" value="Enviar"/>
```

```
  </fieldset>
```

```
</form>
```

```
</body>
```

enviar.php

<body>

<?php

```
$fecha=date("d-m-Y");
```

```
$hora= date("H :i:s");
```

```
$destino="danyelisabet@gmail.com"; /*"micorreo@dominio.com";*/
```

```
$asunto="Comentario";
```

```
$desde='From:' .$_POST['email'];
```

```
$comentario= "
```

```
\n
```

```
Nombre: $_POST[nombre]\n
```

```
Email: $_POST[email]\n
```

```
Consulta: $_POST[texto]\n
```

```
Enviado: $fecha a las $hora\n
```

```
\n
```

```
".  
;
```

```
mail($destino,$asunto,$comentario,$desde);
```

```
echo "Su consulta ha sido enviada, en breve recibirá nuestra  
respuesta.";
```

?>

Alojamiento Gratis

Alojamiento web (en inglés: web **hosting**) es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web.

Un Hosting gratuito puede ser el 000webhost
Accede a:

<https://ar.000webhost.com/>

User Rated
4.9/5
★★★★★
read more..
2208 reviews

Cloud Hosting Gratis

Hosting coste **Cero** con PHP, MySQL, Cpanel & ¡sin publicidad!

¡Regístrate GRATIS!

Sí, ¡es absolutamente gratis!



Let's chat

Looking for help?

Dirección de tutorial: <https://www.youtube.com/watch?v=4RaU7nTUU6M>

En **HOSTINGSAURIO** puede encontrarse un análisis exhaustivos y comparativas de los mejores alojamientos web para que tomar la mejor decisión para tu proyecto digital.

<https://hostingsaurio.com/hosting-gratuito/>

Variables de Sesión

Las sesiones son un método seguro y eficaz de guardar y mantener datos del usuario durante toda su visita.

Podemos guardar por ejemplo una variable que diga si está identificado en nuestro sistema o no y si lo está, también podemos guardar sus datos.

`session_start ()`

Inicia una sesión para el usuario o continúa la sesión abierta en otras páginas.

Al hacer `session_start()` PHP internamente recibe el identificador de sesión almacenado en la cookie o el que se envíe a través de la URL. Si no existe tal identificador de sesión, simplemente lo crea.

Variables de Sesión

La sesión se tiene que inicializar antes de escribir cualquier texto en la página.

Una vez inicializada la sesión con **session_start()**, se pueden utilizar variables de sesión, es decir, almacenar datos para ese usuario, que se conserven durante toda su visita o recuperar datos almacenados en páginas que haya visitado.

Se pueden utilizar variables de sesión a través de `$_SESSION`, que es un array asociativo, donde se accede a cada variable a partir de su nombre, de este modo:

`$_SESSION["nombre _variable"]`

Ejemplo: Contar las páginas visitadas por un usuario durante su sesión

Pasos:

- Inicializar la sesión
- Si no tenemos el contador de páginas creado, lo inicializamos al valor 1
- Si tenemos el contador de páginas creado, lo incrementamos en 1.

```
<?php session_start();  
if (!isset($_SESSION["contador"])){  
    $_SESSION["contador "] = 1;  
}else{  
    $_SESSION["contador "]+=;  
}  
?>
```

Cada página en el sitio, tiene que incrementar en 1 el contador de páginas, o sea, tener este código.

Ejemplo (continuación)

Para mostrar el número de páginas visitadas en un momento dado:

```
<?php echo "Has visitado " .  
$_SESSION["contador "] . " paginas"; ?>
```

EJERCICIO

cuenta.php

```
<?php
session_start();
?>
<html>
<body>
<?php
if (!isset($_SESSION["contador"])){
    $_SESSION["contador"] = 1;
}else{
    $_SESSION["contador"]++;
} ?>
<a href= "cant_visitas.php">Otra pagina</a>
</body>
</html>
```

EJERCICIO

cant_visitas.php

```
<?php
session_start();
?>
<html>
<body>
<a href="cuenta.php"></a>
<?php
echo "Has visitado " . ($_SESSION["contador"]) . " páginas";
?>
<br>
<br>
<a href="cuenta.php">Otra página</a>
</body>
</html>
```

session_destroy

En PHP 4, PHP 5

Destruye toda la información registrada de una sesión.

Sintaxis: bool **session_destroy** (void)

session_destroy() destruye toda la información asociada con la sesión actual. No destruye ninguna de las variables globales asociadas con la sesión, ni destruye la cookie de sesión. Para volver a utilizar las variables de sesión se debe llamar a [session_start\(\)](#).

session_destroy

Para destruir la sesión completamente, como desconectar al usuario, el id de sesión también debe ser destruido. Si se usa una cookie para propagar el id de sesión (comportamiento por defecto), entonces la cookie de sesión se debe borrar.

