

	RAMA:	Informática	CICLO:	DAM			
	MÓDULO	Lenguajes de marcas y sistemas de gestión empresarial				CURSO:	1
	PROTOCOLO:	Apuntes	AVAL:	2	DATA:		
	UNIDAD COMPETENCIA		PHP. Formularios				

## Índice

1.	Crear un formulario simple .....	2
2.	Utilizar GET o POST .....	3
2.1.	Get .....	3
2.2.	Post .....	3
3.	Recibir datos de un formulario .....	3

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM
	MÓDULO	Lenguajes de marcas y sistemas de gestión empresarial		CURSO: 1
	PROTOCOLO:	Apuntes	AVAL: 2	DATA:
	UNIDAD COMPETENCIA	PHP. Formularios		

## 1. Crear un formulario simple

Para el formulario HTML de ejemplo, se creará una página de comentarios y sugerencias que contiene el saludo del usuario, el nombre, la dirección de correo electrónico, respuestas y comentarios.

A pesar de que el código es de fácil creación, se añade a continuación. Se recomienda repasar los apuntes de HTML si hubiese alguna duda en la realización de formularios.

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Formulario</title>
</head>
<body>
  <form action="manejar_formulario.php" method="post">
    <p> Nombre: <select name="tratamiento">
      <option value="Sr.">Sr.</option>
      <option value="Sra.">Sra.</option>
      <option value="Srta.">Srta.</option>
    </select>
    <input type="text" name="nombre" size="20"/> </p>
    <p>Dirección de correo electrónico <input type="text" name="email" size="20"/></p>
    <p>Valoración</p>
    <input type="radio" name="respuesta" value="excelente"/>Excelente
    <input type="radio" name="respuesta" value="normal"/>Normal
    <input type="radio" name="respuesta" value="malo"/>Malo
    <p>Comentarios <textarea name="comentarios" rows="4" cols="40"></textarea></p>
    <input type="submit" name="enviar" value="Enviar mi respuesta"/>
  </form>
</body>
</html>
```

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM	
	MÓDULO	Lenguajes de marcas y sistemas de gestión empresarial			CURSO: 1
	PROTOCOLO:	Apuntes	AVAL:	2	DATA:
	UNIDAD COMPETENCIA	PHP. Formularios			

Al guardar este formulario podríamos utilizar la extensión .html, ya que es una página HTML estándar (no contiene código PHP), aunque también se puede utilizar la extensión .php sin ningún problema ya que como ya hemos visto, en una página PHP cualquier cosa que no se encuentre dentro de las etiquetas PHP `<?php y ?>` se trata como HTML.

El atributo action de la etiqueta form, debe apuntar a un archivo que exista en el servidor, o el formulario no se procesará correctamente. En este caso crearemos un script manejar\_formulario.php ubicado en el mismo directorio que nuestro formulario.

## 2. Utilizar GET o POST

Al utilizar formulario deberemos rellenar el atributo **method**. Este atributo le dice al servidor cómo transmitir los datos desde el formulario hasta el script que los manejará. Existen dos opciones GET y POST.

### 2.1. Get

El método GET envía toda la información recopilada como parte de la URL. Ejemplo:

```
http://www.ejemplo.com/pagina.php?variables=variables$edad=20...
```

Desventajas:

Con el método GET se puede pasar una cantidad limitada de información, además GET envía la información como parte de la URL por lo que si enviamos una contraseña en un formulario, esta será visible para cualquier persona que se encuentre viendo el navegador web lo que puede entrañar un problema de seguridad.

### 2.2. Post

El método POST transmite la información de tal modo que sólo pueda verla el usuario final. Ejemplo:

```
http://www.ejemplo.com/pagina.php
```

Desventajas:

Si el acceso a una página depende del envío de datos al servidor mediante el método post, cada vez que queremos volver a ingresar a esta página debemos enviar nuevamente los datos del formulario, y la mayoría de navegadores advertirá esto con un mensaje de alerta al usuario, que no siempre comprenderá que es lo que está sucediendo.

## 3. Recibir datos de un formulario

Ahora que tenemos el formulario en HTML necesitamos escribir el script PHP que recibirá y procesará los datos del formulario.

En este primer paso solamente se repetirán los datos que el usuario ha introducido en el formulario.

Para acceder a los datos enviados por el formulario, necesitamos hacer referencia a las variables predefinidas \$\_GET o \$\_POST. Si un formulario utiliza el método GET los datos que se envían se encontrarán en la variable \$\_GET y si el formulario utiliza el método POST los datos se encontrarán en la variable \$\_POST.

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM
	MÓDULO	Lenguajes de marcas y sistemas de gestión empresarial		
	PROTOCOLO:	Apuntes	AVAL:	2
	UNIDAD COMPETENCIA	PHP. Formularios		
			DATA:	CURSO: 1

\$\_GET y \$\_POST además de ser variables predefinidas son matrices, esto significa que cada variable puede contener muchos valores. Para acceder a un valor específico, deberemos hacer referencia a la clave o índice de la matriz.

Ejemplo:

```
<input type="text" name="texto"/>
```

Si suponemos que hemos utilizado el método POST el valor introducido por el método form se podría mostrar por pantalla

```
print_r $_POST['texto'];
```

Vamos a crear el script PHP para el formulario del ejemplo anterior.

```
<?php
//Esta página recibe los datos título, nombre, email, respuesta, comentarios y enviar en $_POST
$tratamiento = $_POST['tratamiento'];
$nombre = $_POST['nombre'];
$respuesta = $_POST['respuesta'];
$comentarios = $_POST['comentarios'];
//Mostramos los datos recibidos
echo "<p>Gracias, $tratamiento $nombre por sus comentarios</p>
      <p>Su valoración $respuesta y sus comentarios $comentarios serán tenidos en cuenta</p>"
?>
```

Si queremos comprobar que los campos no están vacíos, podemos hacerlo de la siguiente forma, con isset comprobaremos que el campo está creado y con empty que no está vacío.

```
<?php
//Esta página recibe los datos título, nombre, email, respuesta, comentarios y enviar en $_POST
$flag = FALSE;
$tratamiento = $_POST['tratamiento'];
if (!empty($_POST['nombre'])) {
    $nombre = $_POST['nombre'];
} else {
    $nombre = "";
    $flag = True;
}
if (isset($_POST['respuesta'])) {
    $respuesta = $_POST['respuesta'];
} else {
    $respuesta = "";
    $flag = True;
}
if (!empty($_POST['comentarios'])) {
    $comentarios = $_POST['comentarios'];
} else {
    $comentarios = "";
    $flag = True;
}
```

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM
	MÓDULO	Lenguajes de marcas y sistemas de gestión empresarial		
	PROTOCOLO:	Apuntes	AVAL:	2
	UNIDAD COMPETENCIA	PHP. Formularios		
			DATA:	
			CURSO:	1

```
//Mostramos los datos recibidos dependiendo de si tenemos todos o no
if ($flag == True) {
    echo 'Es necesario rellenar todos los datos del formulario';
    echo "<a href='form.html'>Volver al formulario</a>";
    //lo ideal sería decir cual no esta
} else {
    echo "<p>Gracias, $tratamiento $nombre por sus comentarios</p>";
    echo "<p>Su valoración $respuesta y sus comentarios $comentarios serán tenidos en cuenta</p>";
}
?>
```