

Repaso

¿Como persistimos datos en PHP?

Usando: <?php echo \$var; ?>

De alguna otro manera?

- Con archivos:
 - Abrimos: fopen('archivo', modo);
 - Obtenemos info:
 - Cierta info: fread(<u>recurso</u> , <u>tamaño</u>);
 - Toda la info: file_get_content('archivo');
 - La siguiente linea: fgets(<u>recurso</u>);
 - Escribimos info:
 - fwrite(<u>recurso</u>, 'contenido');
 - file_put_contents('archivo', 'contenido', modo);
- Con JSON:
 - Decode y encode.



Objetivos

- ♦ Subir archivos al servidor usando PHP.
- Mostrar los archivos como las imágenes ó enlaces de descarga.





Uploads

PHP nos da ciertas facilidades para poder manejar la carga de archivos en formularios





Formularios

```
<form action="registrar.php" method="post">
    Nombre: <input type="text" name="nombre">
    Apellido: <input type="text" name="apellido">
        <input type="submit" value="Enviar">
    </form>
```

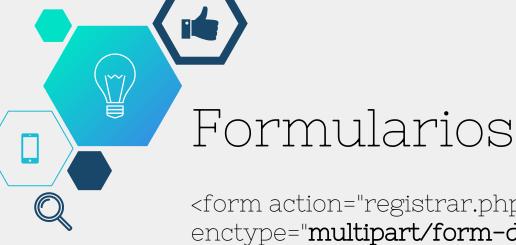




Formularios

```
<form action="registrar.php" method="post">
    Nombre: <input type="text" name="nombre">
    Apellido: <input type="text" name="apellido">
    Avatar:<input type="file" name="imgPerfil">
    <input type="submit" value="Enviar">
    </form>
```





```
<form action="registrar.php" method="post"</pre>
enctype="multipart/form-data">
   <!-- opcional --!>
    <input type="hidden"</pre>
name="MAX_FILE_SIZE" value="30000">
    Nombre: <input type="text" name="nombre">
   Apellido: <input type="text" name="apellido">
   Avatar:<input type="file" name="imgPerfil">
    <input type="submit" value="Enviar">
</form>
```



Es un array asociativo de elementos subidos a través del método POST

En el caso del formulario anterior va a tener una clave llamada **imgPerfil**.



Por cada archivo, contará con un array asociativo.

En este ejemplo, **\$_FILES["imgPerfil"]** es un array asociativo que contiene al menos las siguientes claves interesantes:

- error \$_FILES["imgPerfil"]["error"]
- name \$_FILES["imgPerfil"]["name"]
- tmp_name \$_FILES["imgPerfil"]["tmp_name"]





Indica si la imágen se subió exitosamente.

Habitualmente se compara con la constante UPLOAD_ERR_OK que indica que no hubo errores.

Se pueden ver más errores de subida aquí.





Name indica el nombre del archivo que se subió. Combinándolo con **pathinfo** podemos obtener la extensión del archivo, en caso de necesitarla.

```
<?php
    $nombre = $_FILES["imgPerfil"]["name"];
    pathinfo($nombre, PATHINFO_EXTENSION);
?>
```



\$_FILES - tmp_name

Representa un puntero al **contenido** del archivo subido completo. Combinándolo con **move_uploaded_file** podemos subir un archivo.

```
<?php
    $nombre=$_FILES["imgPerfil"]["name"];
    $archivo=$_FILES["imgPerfil"]["tmp_name"];
    move_uploaded_file($archivo, $nombre);
?>
```





Obtenemos la ruta del archivo que está siendo ejecutado.

```
<?php
    dirname(__FILE__);
?>
```



```
function guardarImagen() {
   if ($_FILES["imgPerfil"]["error"] == UPLOAD_ERR_OK)
       $nombre=$_FILES["imgPerfil"]["name"];
       $archivo=$_FILES["imgPerfil"]["tmp_name"];
       $ext = pathinfo($nombre, PATHINFO_EXTENSION);
       $miArchivo = dirname( FILE ):
       $miArchivo = $miArchivo . "/img/";
       $miArchivo = $miArchivo . "imagenNueva." . $ext;
       move_uploaded_file($archivo, $miArchivo);
guardarImagen();
```



¡A practicar!

```
<?php
   echo "Hora de practicar!";
?>
```