

Práctica Tema 1: Enciclopedia de Animales

Introducción

El objetivo es construir una pequeña aplicación web dinámica utilizando PHP que servirá como una enciclopedia de animales. A través de este proyecto, se aplicarán algunos de los conceptos más importantes de PHP en un entorno práctico y realista.

Se trabajará con datos predefinidos para generar páginas dinámicamente, se gestionará la entrada de datos a través de formularios, se procesarán peticiones GET y POST y se gestionará la subida de archivos al servidor.

El objetivo no es crear una base de datos, sino simular el comportamiento de una aplicación web leyendo y procesando los datos desde un array de PHP.

Archivos Suministrados

Se proporcionará una estructura de archivos inicial:

- **datos.php**: Este archivo contiene toda la información que necesitará tu aplicación. Incluye dos arrays principales: ***\$categorias*** y ***\$animales***.
- **template.html**: Un archivo HTML base que contiene la estructura completa de una página. Será dividido en partes reutilizables.
- **estilos.css**: Una hoja de estilos básica para que la aplicación tenga un aspecto limpio y ordenado. No hay que modificar dicho archivo.

Ejercicios: Pasos a Seguir

Paso 0: Configuración Inicial

1. Crea el repositorio en Github

- Uno de los miembros del equipo debe crear el repositorio.
 - Agrega como colaborador al otro miembro del equipo (Settings -> Collaborators)
 - Da visibilidad también al profesor (acrerosj).
 - Cada uno de los miembros del equipo debe clonar el repositorio en local para trabajar.
 - Uno de los dos debe copiar la carpeta .devcontainer dentro del proyecto para que se genere el contenedor de desarrollo.
2. Crea una estructura de carpetas para tu proyecto:
- **public/**: Aquí se crearán las páginas principales.
 - **data/**: Aquí se almacenan los archivos de datos. En nuestro caso **datos.php**
 - **includes/**: Aquí se guardan los archivos PHP reutilizables (cabecera, pie, funciones).
 - **uploads/**: Contendrá los archivos que suban los usuarios. Dentro, crea dos subcarpetas:
 - **uploads/images/**
 - **uploads/pdfs/**
3. Toma el archivo **template.html** y divídelo en dos archivos PHP:
- **includes/header.php**: Con la cabecera.
 - **includes/footer.php**: Con el pie de página.
4. En **header.php**, asegúrate de que el **<title>** pueda ser dinámico. Puedes usar una variable PHP como **\$titulo_pagina**.

Paso 1: Página de Inicio (**index.php**)

- Crea el archivo **index.php**.
- Incluye la cabecera (**header.php**) y el pie de página (**footer.php**).
- Importa los datos desde **datos.php**.

- Recorre el array `$categorias` y muestra un listado (usando `` y ``) con el nombre de cada categoría.
- Cada nombre de categoría debe ser un enlace que apunte a `category.php`, pasando el ID de la categoría por la URL. Ejemplo: `category.php?id=1`.

Paso 2: Página de Categoría (`category.php`)

- Crea el archivo `category.php`.
- Incluye la cabecera y el pie de página.
- Recupera el ID de la categoría de la URL usando `$_GET`. **Valida que el ID exista.**
- Busca en el array `$categorias` la información de la categoría correspondiente y muestra su nombre y descripción.
- Recorre el array `$animales` y muestra un listado con los nombres de los animales que pertenecen a esa categoría.
- Cada nombre de animal debe ser un enlace que apunte a `animal.php`, pasando el ID del animal por la URL. Ejemplo: `animal.php?id=47293`

Paso 3: Página del Animal (`animal.php`)

- Crea el archivo `animal.php`.
- Incluye la cabecera y el pie de página.
- Recupera el ID del animal de la URL usando `$_GET`. **Valida que el animal exista.**
- Busca en el array `$animales` toda la información del animal y muéstrala de forma clara:
 - Nombre, hábitat.
 - La imagen del animal (usando la URL del array).
 - Un listado de sus características (recorriendo el sub-array `caracteristicas`).

- Un enlace al PDF con su descripción.
- Un enlace a la página de actualización:
`update-animal.php?id=ID_DEL_ANIMAL.`

Paso 4: Formulario de Registro (**register-animal.php**)

- Crea el archivo `register-animal.php`.
- Incluye la cabecera y el pie de página.
- Crea un formulario HTML que envíe los datos por **POST** a `process-animal.php`. No olvides el atributo `enctype="multipart/form-data"` para permitir la subida de archivos.
- El formulario debe contener los siguientes campos:
 - Un `select` (desplegable) para la **categoría**. Este desplegable debe generarse dinámicamente recorriendo el array `$categorias`. El `value` de cada `option` debe ser el ID de la categoría.
 - Un `input` de tipo `number` con un **ID** para el animal.
 - Un `input` de tipo `text` para el **nombre** del animal.
 - Un `input` de tipo `text` para el **hábitat**.
 - Un `textarea` para las **características**, indicando al usuario que las separe por comas (ej: "Característica 1, Característica 2").
 - Un `input` de tipo `file` para subir la **imagen**.
 - Un `input` de tipo `file` para subir el **PDF** con la descripción.
 - Un botón de `submit`.

Paso 5: Procesamiento del Formulario (**process-animal.php**)

- Crea el archivo `process-animal.php`.
- Esta página no guardará los datos en el array, solo los procesará y mostrará.
- Recoge todos los datos enviados desde el formulario usando `$_POST` y `$_FILES`.

- **Validación:** Comprueba que los campos de texto no estén vacíos.
- **Procesamiento de archivos:**
 - Para la imagen y el PDF, comprueba si se subieron correctamente (`$_FILES['nombre_del_campo']['error'] === UPLOAD_ERR_OK`).
 - Mueve los archivos subidos desde su ubicación temporal a las carpetas correspondientes (`uploads/images/` y `uploads/pdfs/`) usando `move_uploaded_file()`.
 - Como nombre de archivo para almacenar tanto la imagen como el pdf debes usar el id del animal. Por ejemplo, para el león será `47293.jpg` o `47293.pdf`.
- **Visualización:** Muestra en pantalla todos los datos recibidos del formulario, con un formato similar al de `animal.php`: nombre, hábitat, la imagen que se acaba de subir, las características (procesa la cadena de texto para mostrarla como una lista) y un enlace al PDF subido.

Paso 6: Actualización de Archivos (`update-animal.php`)

- Crea el archivo `update-animal.php`.
- Recupera el ID del animal de la URL usando `$_GET`. **Valida que el animal exista.**
- Muestra el nombre del animal que se está modificando.
- Crea un formulario simple (que se envíe a sí mismo por `POST`) con `enctype="multipart/form-data"` que solo permita:
 - Subir una nueva imagen (`input` tipo `file`).
 - Subir un nuevo PDF (`input` tipo `file`).
- **Lógica de procesamiento (dentro del mismo archivo):**
 - Comprueba si la petición es de tipo `POST`.
 - Comprueba que existe un animal con ese ID. Si no lo hay muestra un mensaje de “animal no encontrado”. En este ejercicio, como no

guardamos los nombres de los nuevos animales en el array, sólo serán válidos los animales del archivo `datos.php` original.

- Si lo hay, procesa los archivos subidos, sobrescribiendo los archivos existentes para ese animal, simplemente los guardaremos con el ID del animal, ej: `uploads/images/47293.jpg`).
- Muestra un mensaje de éxito tras la subida.

BONUS: Funciones Comunes (`funciones.php`)

- Si notas que repites código (por ejemplo, la lógica para buscar un animal por su ID o una categoría por su ID), crea un archivo `includes/funciones.php`.
- Define funciones como `getAnimalById($id)` o `getCategoryById($id)`.
- Incluye este archivo en las páginas que lo necesiten usando `require_once`.