

# Actividad Práctica.

## Procesamiento de Archivos JSON en PHP

### Introducción

El formato JSON (JavaScript Object Notation) **es un estándar de intercambio de datos ligero y fácil de leer**, utilizado comúnmente para representar datos estructurados. A diferencia del formato CSV, **JSON permite representar datos complejos, como objetos anidados y arreglos**. Debido a su compatibilidad con JavaScript y su capacidad para ser utilizado en aplicaciones web, JSON se ha convertido en uno de los formatos más populares para el intercambio de datos en APIs y aplicaciones web.

En esta actividad, vas a aprender cómo procesar archivos JSON en PHP.

JSON se basa en dos estructuras principales:

1. **Objetos:** Son colecciones de **pares clave-valor**, similares a los diccionarios en Python o a los objetos en JavaScript. Se representan con llaves {}.
2. **Arrays:** Son listas ordenadas de valores, que pueden ser de cualquier tipo. Se representan con corchetes [].

### Ejemplo

```
{
  "libros": [
    {
      "titulo": "Clean Code",
      "autor": ["Robert C. Martin"],
      "anio": 2008,
      "genero": "Programming"
    },
    {
      "titulo": "The Pragmatic Programmer",
      "autor": ["Andrew Hunt", "David Thomas"],
```

```

        "anio": 1999,
        "genero": "Programming"
    },
    {
        "titulo": "Design Patterns",
        "autor": ["Erich Gamma", "Richard Helm", "Ralph Johnson", "John Vlissides"],
        "anio": 1994,
        "genero": "Software Engineering"
    }
]
}

```

Importante

- **Raíz del objeto JSON:** Todo el contenido está dentro de un objeto JSON, que comienza y termina con llaves {}.
- **Clave "libros":** Es un par clave-valor, donde la clave es "libros" y su valor es un array de objetos que representan cada libro.
- **Arreglo de libros:** El valor de "libros" es un array [] que contiene múltiples objetos (libros), cada uno de los cuales está representado entre llaves {}.

Cada libro tiene varias propiedades (pares clave-valor):

- "titulo": el título del libro.
- "autor": el autor del libro ,etc

## Funciones relacionadas con la lectura de archivos JSON

### file\_get\_contents()

file\_get\_contents es una función de PHP que se utiliza para leer el contenido de un archivo como una cadena. Esta función es útil para obtener el contenido completo de archivos, incluidos los archivos JSON.

```
$jsonData = file_get_contents('./libros.json');
```

Devuelve el contenido del archivo en caso de éxito, o false si falla.

## json\_decode()

json\_decode **convierte una cadena JSON en un objeto o un array de PHP.**

Esta función es fundamental para procesar datos JSON, ya que transforma los datos en un formato que puede ser fácilmente manipulado en PHP.

Devuelve un objeto si se utiliza el valor predeterminado, **o un array si se pasa true como segundo argumento.**

### Ejemplo tratar los datos como un array

```
// Obtener el objeto o el array json
$data = json_decode($jsonArchivo, true);

// Procesar los datos JSON
foreach ($data as $libro) {

    // Procesar cada libro
    //$libro['title'], $libro['author']...

}
}
```

### En este otro ejemplo vamos a procesar la decodificación como un objeto JSON

```
<?php
// Cadena JSON
$jsonString = '{
    "titulo": "Clean Code",
    "autor": "Robert C. Martin",
    "anio": 2008,
    "genero": "Programming"
}';

// Decodificar JSON como objeto
$libro = json_decode($jsonString, false);

// Acceder a los campos del objeto en forma de objeto json
```

```
echo "Título: " . $libro->titulo . "\n";  
echo "Autor: " . $libro->autor . "\n";  
echo "Año: " . $libro->anio . "\n";  
echo "Género: " . $libro->genero . "\n";  
?>
```

## Desarrollo de la actividad práctica

Implementa y desarrolla cada uno de los módulos que se te muestran a continuación.

Haz que index.html envíe el archivo al módulo procesar\_json.php

### index.html

Formulario HTML para subir archivos JSON.

NOTA: usa el siguiente tipo de input en HTML

```
<input type="file" name="file" id="file" accept=".json" required>
```

#### **accept=".json":**

- Este atributo indica al navegador que solo debe mostrar archivos con la extensión .json en el cuadro de diálogo de selección de archivos. Esto ayuda a guiar al usuario a seleccionar solo archivos JSON, mejorando la experiencia del usuario y reduciendo la posibilidad de errores al subir un archivo incorrecto.

#### **required:**

- Este atributo indica que el campo es obligatorio. Si el usuario intenta enviar el formulario sin seleccionar un archivo, el navegador mostrará un mensaje de advertencia y no permitirá el envío del formulario.

### Módulo procesar\_json.php

Procesa el archivo json mostrando al cliente de la solicitud sólo la siguiente información: el título y el autor

Completa el siguiente módulo con las instrucciones que faltan.

NOTA : SIGUE Estrictamente el orden que se muestra a continuación

```
<?php
// Verificar si se ha subido un archivo y guardarlo en carpeta
temporal ./archivo/ dentro del proyecto actual
'-----'
```

```

    $ruta_archivo_subido= -----
// Leer el contenido del archivo subido al servidor y procesar cada
línea

    //Obtener el contenido como cadena
    $jsonData = file_get_contents($ruta_archivo);

    // Decodificar el JSON
    $datos_json = json_decode($jsonData, true);

    // Procesar los datos JSON de forma individualizada
    foreach ($datos_json as $libro) {
        // Acceder a cada campo (título y nombre )
        //y enviarlo a la salida
        '-----'
    }

```

## Cuestiones

- Los datos JSON se procesar inicialmente como un array en esta actividad. Cambia el módulo **procesar\_json.php** para que los datos JSON se procesen como un objeto