

## EJERCICIOS PARA PRACTICAR LA UNIDAD 1

1. Crea una página PHP que muestre la fecha y hora actual.
2. Divide el ejercicio 1 en varios archivos PHP usando ``include`` y ``require``, separando la lógica de la presentación.
3. Implementa un sistema simple de *login* sin usar sesiones en PHP.
4. Diseña una página que utilice funciones para realizar diferentes operaciones matemáticas y muestre los resultados.
5. Escribe un *script* PHP que utilice ``readfile`` para mostrar el contenido de un archivo de texto.
6. Emplea ``urlencode`` para enviar datos a través de una URL y recogerlos en otra página PHP.
7. Escribe un *script* que incluya un archivo PHP usando ``require_once`` y maneje errores si el archivo no se encuentra.
8. Desarrolla una aplicación que pase datos codificados con ``urlencode`` a través de un formulario y los recupere y decodifique en otra página.
9. Crea un formulario HTML que envíe datos a un *script* PHP y muestre esos datos en la misma página o en una diferente.
10. Genera un script PHP que lea datos de un archivo CSV y los muestre en una tabla HTML.
11. Desarrolla un script PHP que tome datos de un formulario y los guarde en un archivo CSV.
12. Implementa una funcionalidad para importar un archivo CSV y actualizar luego dicho CSV y guardarlo.
13. Crea un *array* asociativo en PHP y muestra su contenido en una estructura de tabla HTML.
14. Desarrolla un *script* que convierta un *array* multidimensional en un *array* asociativo y viceversa.
15. Crea un script PHP para filtrar, ordenar y realizar operaciones sobre *arrays* asociativos multidimensionales.
16. Implementa el algoritmo de ordenamiento burbuja en PHP.
17. Resuelve un problema clásico como la búsqueda binaria o la generación de números Fibonacci.
18. Escribe una función PHP para resolver el problema de las Torres de Hanoi.
19. Implementa el algoritmo de búsqueda en profundidad (DFS) para recorrer un árbol o un grafo.
20. Implementa un sistema de paginación para mostrar una lista de productos de un CSV, con opciones para ordenar y filtrar los resultados.

## PROYECTO PHP 8: APLICACIÓN WEB DE ALQUILER DE BICICLETAS

### Descripción del Proyecto:

Crea una aplicación web para administrar un sistema de alquiler de bicicletas. La aplicación permitirá a los usuarios registrar alquileres, subir y procesar datos relacionados con los alquileres (en formato texto o CSV), y visualizar informes.

### Funcionalidades :

#### 1. Interfaz de Usuario:

- Diseña una interfaz de usuario atractiva utilizando HTML y CSS para navegar entre las diferentes secciones de la aplicación, como registro de alquiler, visualización de alquileres, y generación de informes.

#### 2. Registro y Visualización de Alquileres:

- Implementa un formulario para registrar nuevos alquileres, incluyendo detalles como nombre del cliente, fecha, duración del alquiler y tipo de bicicleta.
- Muestra una lista de todos los alquileres registrados con opciones para ver detalles o editar registros.

#### 3. Procesamiento de Archivos de Alquiler:

- Permite subir archivos de texto o CSV con datos de alquileres.
- Lee y muestra el contenido de estos archivos, presentando los datos en formato legible.

#### 4. Manipulación de Archivos CSV:

- Al seleccionar un archivo CSV, lee su contenido y muestra los datos de alquiler en una tabla HTML.
- Permite realizar operaciones como ordenar y filtrar los registros de alquiler en la tabla.

#### 5. Edición y Análisis de Datos de Alquiler:

- Ofrece una opción para editar el contenido del archivo CSV directamente en la página y guardar los cambios.
- Realiza análisis simples, como calcular el ingreso total o el tiempo promedio de alquiler.

#### 6. Generación de Informes:

- Genera informes basados en los datos de alquiler, incluyendo análisis y estadísticas relevantes.
- Los informes se pueden descargar en formato de texto.

#### 7. Uso de Arrays y Funciones:

- Utiliza arrays asociativos para manejar los datos de alquiler y las operaciones realizadas.
- Crea funciones personalizadas para diferentes operaciones, como análisis de datos y generación de informes.

#### 8. Funciones Avanzadas:

- Implementa `readfile` y `urlencode` para gestionar la lectura de archivos y la transferencia segura de datos. Luego, organiza el código en varios archivos PHP usando `include` y `require`.

## **EJERCICIOS PARA PRACTICAR LA UNIDAD 2**

- 1.** Realiza una consulta simple para obtener todos los registros de una tabla.
- 2.** Crea un formulario HTML para ingresar datos y un script PHP que procese e inserte un nuevo registro en la base de datos.
- 3.** Desarrolla un formulario y script PHP para seleccionar un registro de la base de datos y actualizar uno o más de sus campos.
- 4.** Implementa un sistema que muestre registros con un botón de eliminar, pidiendo confirmación antes de proceder.
- 5.** Muestra los datos de una tabla en un formato de lista paginada.
- 6.** Crea un formulario de búsqueda que permita filtrar los registros por varios criterios.
- 7.** Implementa validación de datos en PHP para un formulario de registro de usuarios.
- 8.** Subida y Almacenamiento de Archivos: Desarrolla un formulario para subir archivos, guardándolos en el servidor y registrando su ubicación en la base de datos.
- 9.** Implementa un sistema de inicio de sesión con nombre de usuario y contraseña.
- 10.** Recupera datos de una base de datos y muéstralos en una tabla HTML.
- 11.** Escribe un script PHP que se conecte a una base de datos MySQL y maneje excepciones.
- 12.** Desarrolla un sistema CRUD completo para la gestión de usuarios.
- 13.** Implementa un sistema de paginación con funcionalidades adicionales, como ordenamiento o búsqueda integrada.
- 14.** Desarrolla un sistema de autenticación más completo con registro, inicio de sesión y cierre de sesión.
- 15.** Escribe un script que maneje los errores de base de datos de forma segura y amigable para el usuario.
- 16.** Crea un sistema que gestione relaciones entre distintas tablas, como usuarios y sus publicaciones.
- 17.** Implementa un sistema de autenticación que incluya hashing de contraseñas para mayor seguridad.
- 18.** Utiliza consultas preparadas para proteger tu aplicación de inyecciones SQL.
- 19.** Permite a los usuarios calificar y comentar eventos, y muestra estas calificaciones y comentarios.
- 20.** Integra una API externa para obtener información adicional, como en el caso del sistema de gestión de eventos.

# PROYECTO PHP 8 Y BASES DE DATOS: SISTEMA DE GESTIÓN DE EVENTOS

## Descripción del Proyecto:

Desarrolla un sistema de gestión de eventos utilizando PHP 8 y MySQL. Este sistema permitirá a los usuarios crear, buscar, inscribirse y gestionar eventos. También manejará aspectos como la gestión de asistentes, programación de eventos y comentarios de los usuarios.

## Funcionalidades :

### 1. Registro e Inicio de Sesión de Usuarios:

Implementa un sistema de registro e inicio de sesión para usuarios y organizadores de eventos. El inicio de sesión lo puedes hacer con sesiones o sin sesiones (pidiendo una contraseña para cada operación que haga el usuario).

### 2. Creación y Gestión de Eventos:

Facilita a los organizadores la creación y gestión de eventos, incluyendo detalles como fecha, hora, ubicación y descripción. Implementa opciones para editar o cancelar eventos.

### 3. Búsqueda y Filtrado de Eventos:

Desarrolla una interfaz para que los usuarios busquen eventos por nombre, ubicación, fecha o categoría. Implementa filtros avanzados para mejorar la experiencia de búsqueda.

### 4. Inscripción y Gestión de Asistentes:

Permite a los usuarios inscribirse en eventos y cancelar su inscripción. Gestiona la lista de asistentes y la capacidad máxima de eventos.

### 5. Sistema de Calificaciones y Comentarios:

Implementa una funcionalidad para que los usuarios puedan calificar eventos y dejar comentarios. Muestra las calificaciones y comentarios en la página de detalles del evento.

### 6. Panel de Administración de Eventos:

Crea un panel de administración para que los organizadores gestionen sus eventos, asistentes y reciban *feedback*. Incluye estadísticas y reportes sobre la participación en los eventos.

### 7. Notificaciones y Recordatorios:

Envía notificaciones por correo electrónico a los usuarios para confirmaciones de inscripción, cambios en los eventos y recordatorios. Permite a los organizadores enviar mensajes a todos los inscritos en un evento.

### EJERCICIOS PARA PRACTICAR LA UNIDAD 3

1. Escribe un *script* PHP que inicie una sesión y guarde algunos datos de sesión como el nombre del usuario.
2. Implementa un sistema que use *cookies* para guardar las preferencias del usuario, como el idioma o el tema de la página.
3. Desarrolla un simple sistema de *login* que almacene la información del usuario en la sesión después de un inicio de sesión exitoso.
4. Crea un contador de visitas que utilice sesiones para recordar cuántas veces un usuario ha visitado la página.
5. Escribe un *script* que use *cookies* para almacenar los artículos en un carrito de compras.
6. Usa *cookies* y sesiones para transferir datos de forma segura entre diferentes páginas de un sitio web.
7. Escribe un *script* que gestione errores relacionados con sesiones y *cookies*, como la expiración de una sesión o una *cookie*.
8. Implementa un sistema de comentarios donde las *cookies* recuerden los detalles del usuario (como nombre y correo electrónico) para futuras visitas.
9. Crea un formulario de ingreso que use *cookies* para recordar el nombre de usuario y ofrecer un inicio de sesión más rápido.
10. Genera un *script* PHP que proteja el acceso a una página, permitiendo el ingreso solo a usuarios con una sesión activa.
11. Desarrolla un *script* PHP que guarde y recupere *arrays* asociativos en la sesión.
12. Modifica un proyecto PHP existente para que utilice sesiones y mejore la gestión de usuarios.
13. Crea un sistema de control de acceso basado en roles de usuario utilizando sesiones.
14. Desarrolla un *script* que cree y destruya *cookies* de forma segura, asegurando la privacidad del usuario.
15. Implementa sesiones para seguir las actividades de un usuario en un sitio web, como las páginas visitadas o el tiempo de permanencia.
16. Analiza y mejora el rendimiento de un sitio web al manejar sesiones y *cookies* de manera eficiente.
17. Escribe un *script* que maneje conflictos de sesiones cuando se ejecutan varias aplicaciones PHP en paralelo.
18. Implementa la personalización del contenido de un sitio web basado en las preferencias almacenadas en *cookies*.
19. Crea un formulario HTML que utilice *cookies* para pre-cargar información basada en visitas anteriores.
20. Desarrolla un sistema de notificaciones que utilice sesiones para mostrar mensajes personalizados a los usuarios.

## PROYECTO PHP 8: SITIO WEB CON CAMBIO DE TEMAS Y ESTILOS CSS CON SESIONES Y COOKIES

### Descripción del Proyecto:

Desarrolla un sitio web dinámico en PHP 8 que permita a los usuarios cambiar y personalizar temas y estilos CSS. Utiliza sesiones y cookies para recordar las preferencias de estilo del usuario, proporcionando una experiencia de usuario personalizada y coherente a lo largo de las visitas.

### Funcionalidades :

#### 1. Interfaz de Usuario para Selección de Temas:

- Diseña una interfaz de usuario intuitiva con HTML y CSS que permita a los usuarios seleccionar entre diferentes temas y estilos.
- Incluye una variedad de temas predefinidos y la opción de personalizar colores, fuentes y otros elementos de estilo.

#### 2. Uso de Sesiones para Temas Activos:

- Implementa un sistema de inicio de sesión que guarde las preferencias de estilo del usuario en una sesión PHP.
- Asegura que el tema seleccionado permanezca constante durante la sesión del usuario, incluso al navegar por diferentes páginas del sitio.

#### 3. Gestión de Cookies para Recordar Preferencias:

- Utiliza cookies para almacenar las preferencias de estilo del usuario, de manera que se recuerden en visitas futuras.
- Permite a los usuarios cambiar sus preferencias en cualquier momento, actualizando las cookies correspondientes.

#### 4. Personalización Dinámica del Sitio:

- Carga dinámicamente los estilos CSS basados en las preferencias del usuario almacenadas en la sesión o en las cookies.
- Implementa un mecanismo para que los cambios de estilo se reflejen instantáneamente en toda la interfaz del sitio.

#### 5. Seguridad y Validación de Datos:

- Implementa medidas de seguridad para validar las entradas de los usuarios y proteger contra vulnerabilidades comunes en la gestión de sesiones y cookies.
- Asegura que solo se apliquen estilos válidos y previene la inyección de código malicioso a través de la personalización de estilos.

#### 6. Panel de Configuración de Usuario:

- Ofrece un panel de configuración donde los usuarios pueden ver y modificar sus preferencias de estilo actuales.
- Incluye opciones para restablecer a los estilos predeterminados o para crear un conjunto personalizado de preferencias.