Desarrollo Web en Entorno Servidor - Curso 2017/2018 - IES Leonardo Da Vinci - Alberto Ruiz

U3P03 - Acceso al catálogo con PHP

Entrega de: Sergio Guijarro Muñoz

---

1. Descripción:

El objetivo es practicar el acceso a bases de datos desde PHP, probando casos algo más avanzados como el filtrado o la ordenación de resultados.

2. Formato de entrega:

Incluye capturas del resultado en la Parte 2, o bien enseña el proyecto al profesor

3. Trabajo a realizar:

Parte 1: Preparación de la base de datos

Previamente preparaste una base de datos con una tabla con usuarios (para trabajar más adelante la autenticación) y otra de obras de alguna temática de tu interés. Asumiremos que esas tablas se llaman usuario y obra, y que nuestra temática son libros: pero puedes adaptar los nombres y contenido siempre que respetes la idea general de lo que vamos a hacer. Si tienes dudas sobre si tus cambios son excesivos, consulta al profesor.

- Añade, si no lo has hecho ya, una tabla autor con algunos datos sobre el autor de los libros. Esta tabla tendrá un campo de clave primaria que será un número autoincrementado, que en esta explicación tendrá el nombre id.

- La tabla obra incluirá un campo idAutor que contendra el identificador del autor (clave ajena).

- Para establecer la restricción de clave ajena desde phpMyAdmin debes seguir los siguientes pasos:

- Seleccionar la tabla obra > Estructura > Vista de relaciones

- Añadir un índice (la opción está en la parte inferior) llamado "libro-autor", de tipo INDEX para el campo idAutor

- Añadir una restricción de clave foránea con los siguientes datos:

- Nombre: libro-autor

- Columna: idAutor

- Restricción de clave foránea: catalogo > autor > id

- Comprobar la aplicación de la restricción es sencillo: inserta un nuevo libro en la tabla, o modifica uno existente. Verás que en el campo idAutor no podrás escribir libremente un valor, sino que tendrás que elegir de una lista desplegable alguno de los valores id existentes en la tabla autor.

Parte 2: Proyecto de acceso a la base de datos

Vamos a mostrar el contenido del catálogo. Iremos abordando el problema de forma incremental: no continúes con un paso hasta tener funcionando el anterior.

Puesto que vamos a trabajar con consultas SQL, recuerda que puedes probarlas en la sección SQL de phpMyAdmin antes de incluirlas en tu programa, para asegurarte de que son correctas y devuelven el resultado que esperas.

2.1 Preparación

- Crea un proyecto PHP llamado U3P03-PHP-Catalogo

- Crea una carpeta img en la estructura del proyecto y copia en ella las imágenes correspondientes a los elementos de tu catálogo

- Crea una carpeta doc en la estructura del proyecto y copia en ella este documento de enunciado

- Crea una carpeta sql en la estructura del proyecto y copia en ella un script SQL, resultado de exportar tu base de datos

2.2 Listado básico

- Codifica el archivo Obra.php (observa la mayúscula inicial) en el que definas la clase (sin constructor) correspondiente a las obras de tu base de datos. Puedes cambiar el nombre por algo más adecuado a tu temática (Libro,Comic,Disco,etc)

- Codifica un archivo mostrarCatalogo.php, que creará objetos por cada obra y los mostrará en una tabla: pero sólo mostrará los campos de nombre e identificador del autor. Observa que no se imprimirá el nombre del autor sino su identificador, puesto que es lo que tenemos almacenado en la tabla obra.

2.3 Nombre de autores

- Ahora debes conseguir que se muestre el nombre del autor. Vamos a estudiar dos posibilidades:

1. Dentro del while en el que procesas cada objeto obra, haz una segunda consulta SQL pidiendo el nombre del autor de esa obra. Recuerda liberar el resultado de esa consulta antes de cerrar el while

2. (Opcional) Es posible obtener toda la información con una sola consulta SQL multitabla. Adapta la siguiente consulta a tu base de datos y pruébala en la sección SQL de phpMyAdmin para comprender cómo funciona:

SELECT \* from libro,autor WHERE autor.idautor=libro.idAutor

El problema de esta solución es que nuestra clase Libro no tiene un atributo para el nombre del autor, sino para el identificador. Para solucionarlo podríamos añadir dicho atributo (y su correspondiente función get) a la clase.

2.4 Detalle de obras

- Codifica un archivo mostrarObra.php que mostrará todos los detalles de una obra (incluyendo el nombre del autor y la imagen, que debe visualizarse) cuyo identificador recibirá como parámetro. Asegúrate de que la página ofrezca un mensaje de error si no recibe el parámetro o este no existe en la base de datos

- Modifica mostrarCatalogo.php para que los nombres de las obras sean en realidad enlaces a la plantilla mostrarObra.php. En esta plantilla incluirás un enlace para volver.

2.5 Ordenación de resultados

Vamos a permitir la ordenación del listado de obras por nombre o autor, y en los dos sentidos (ascendente o descendente).

- Añade en mostrarCatalogo.php unos enlaces con forma de triángulo o flecha hacia arriba y hacia abajo en la cabecera de la tabla en la que muestras las obras: hazlo en las dos columnas. Puedes usar imágenes o entidades HTML.

- Cuando se pulse en la flecha hacia arriba, los resultados se mostrarán de la A a la Z, y cuando se pulse en la flecha hacia abajo, al contrario

- (Opcional): la flecha correspondiente al orden que se está utilizando en este momento, no será un enlace. Puedes usar otro color para el icono.

2.6 Filtro de obras por autor

- Modifica la plantilla mostrarCatalogo.php para que, si recibe como parámetro el identificador de un autor, se muestre:

- Los datos del autor (incluyendo su imagen si has añadido un campo de imagen a la tabla autor)

- El listado de obras, pero mostrando solo aquellas que son de ese autor.

- Un enlace Eliminar filtros que lleve a la misma plantilla, esta vez sin parámetros.

- (Opcional): Intenta que los enlaces de ordenación de resultados funcionen también para esta selección de obra

- Ahora modifica mostrarCatalogo.php para que los nombres de autores sean enlaces al propio archivo, pero enviando como parámetro el identificador del autor.

2.6 Filtro de búsqueda

- Incluye en mostrarCatalogo.php un formulario con un campo de texto para buscar obras por nombre. El formulario será procesado por la misma plantilla, y mostrará solo aquellas obras cuyo nombre contenga el texto buscado. Esto puede hacerse de dos formas:

- Más sencilla: obligar a que contenga todo el texto leído del campo en el mismo orden

- Más compleja y mejor: separar las palabras introducidas en el campo, y obligar a que las contenga todas

- Al final se incluirá el enlace Eliminar filtros para volver a la misma plantilla, sin parámetros.

- (Opcional): permitir la búsqueda también por autor

2.7 Navegabilidad y formato

- Asegúrate de que tu solución es navegable sin utilizar el botón Atrás del navegador: si no es así, incluye enlaces para solucionarlo

- Aporta el formato que desees a la presentación de resultados (márgenes, espaciado de celdas, colores...)