



**CICLO: D.A.W.
MÓDULO DE DESARROLLO WEB
ENTORNO SERVIDOR**

Tarea N° 06

**Alumno:
Armando Herrero Silva
76442135W**

Los documentos, elementos gráficos, vídeos, transparencias y otros recursos didácticos incluidos en este contenido pueden contener imprecisiones técnicas o errores tipográficos. Periódicamente se realizan cambios en el contenido. Fomento Ocupacional FOC SL puede realizar en cualquier momento, sin previo aviso, mejoras y/o cambios en el contenido.

Es responsabilidad del usuario el cumplimiento de todas las leyes de derechos de autor aplicables. Ningún elemento de este contenido (documentos, elementos gráficos, vídeos, transparencias y otros recursos didácticos asociados), ni parte de este contenido puede ser reproducida, almacenada o introducida en un sistema de recuperación, ni transmitida de ninguna forma ni por ningún medio (ya sea electrónico, mecánico, por fotocopia, grabación o de otra manera), ni con ningún propósito, sin la previa autorización por escrito de Fomento Ocupacional FOC SL.

Este contenido está protegido por la ley de propiedad intelectual e industrial. Pertenecen a Fomento Ocupacional FOC SL los derechos de autor y los demás derechos de propiedad intelectual e industrial sobre este contenido.

Sin perjuicio de los casos en que la ley aplicable prohíbe la exclusión de la responsabilidad por daños, Fomento Ocupacional FOC SL no se responsabiliza en ningún caso de daños indirectos, sean cuales fueren su naturaleza u origen, que se deriven o de otro modo estén relacionados con el uso de este contenido.

© 2022 Fomento Ocupacional FOC SL todos los derechos reservados.

Contenido

1. RA6_d.....2

2. RA6_e.....6

3. RA6_f.....10

1. RA6_d.

Crear una base de datos Libros. sobre esta base de datos, crear dos tablas.

- **Tabla Autor que contenga los siguientes campos:**
 - Una columna id de tipo entero.
 - Una columna nombre de tipo cadena con una longitud máxima de 15 caracteres.
 - Una columna apellidos de tipo cadena con una longitud máxima de 25 caracteres.
 - Una columna nacionalidad de tipo alfanumérica con una longitud máxima de 10 caracteres.

A esta tabla se le debe insertar el contenido mostrado a continuación:

id	nombre	apellidos	nacionalidad
0	J. R. R.	Tolkien	Inglaterra
1	Isaac	Asimov	Rusia

- **Crear una tabla Libro que contenga los siguientes campos:**
 - Una columna id de tipo entero.
 - Una columna título de tipo cadena con una longitud máxima de 50 caracteres.
 - Una columna f_publicacion de tipo fecha.
 - Una columna id_autor como clave externa del id autor.
- A esta tabla se le debe insertar el contenido mostrado a continuación:

id	título	f_publicacion	id_autor
0	El Hobbit	21/09/1937	0
1	La Comunidad del Anillo	29/07/1954	0
2	Las dos torres	11/11/1954	0
3	El retorno del Rey	20/10/1955	0
4	Un guijarro en el cielo	19/01/1950	1
5	Fundación	01/06/1951	1
6	Yo, robot	02/12/1950	1

- **Crear una clase llamada Libros.**
 - Añadir un método conexion(servidor, base de datos, usuario, contraseña), que establece la conexión y retorna un objeto de conexión con la base de datos, si hay algún error retorna null.
 - En el foro de la tarea se suministra las pruebas de PHPUnit que debe satisfacer el código.

Lo primero que debemos hacer es crearla base de datos y sus correspondientes tablas. Hay que tener en cuenta que cuando nos conectamos a Heidi por primera vez debemos comprobar si se ha conectado correctamente. En nuestro caso como vamos a crear la base de datos, en la conexión no debemos poner el nombre de la base que vamos a crear, pues no existe aún:

```
<h2>CREACION DE BASE DE DATOS Y TABLAS</h2>
<?php
    $hostname = "localhost";
    $usuario = "root";
    $contrasena = "";

    $conn = new mysqli($hostname, $usuario, $contrasena);

    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }

    $sql = "CREATE DATABASE IF NOT EXISTS Libros";

    $conn->query($sql);

    $conn = new mysqli($hostname, $usuario, $contrasena, "Libros");

    $sql = "CREATE TABLE Autor(
        Id INT AUTO_INCREMENT PRIMARY KEY,
        Nombre VARCHAR(15) NOT NULL,
        Apellidos VARCHAR(25) NOT NULL,
        Nacionalidad VARCHAR(20) NOT NULL
    )";

    if(($conn->query($sql)) === TRUE) {
        echo "<p>La tabla Autor se ha creado correctamente</p>";
    }else{
        echo "<p>Error al crear la tabla Autor: ".$conn->error."</p>";
    }

    $sql = "CREATE TABLE Libro(
        Id INT AUTO_INCREMENT PRIMARY KEY,
        Titulo VARCHAR(50) NOT NULL,
        Fecha_publicacion DATE NOT NULL,
        Nacionalidad VARCHAR(20) NOT NULL,
        Id_Autor INT
```

A continuación, procedemos a crear las sentencias de para rellenar dichas tablas que acabamos de crear:

```
// Ahora se rellenan las tablas

$sql = "INSERT INTO Autor (Nombre, Apellidos, Nacionalidad) VALUES ('J.R.R.', 'Tolkien', 'Inglesa'), ('Isaac', 'Asimov', 'Rusa');";

if(($conn->query($sql)) === TRUE) {
    echo "<p>Se han insertado los datos en la tabla Autor correctamente.</p>";
}else{
    echo "<p>Error al insertar los datos en la tabla Autor: ".$conn->error."</p>";
}

$result = $conn->query("SELECT Id FROM Autor WHERE Nombre = 'J.R.R.' AND Apellidos = 'Tolkien'");
$id_Tolkien = $result->fetch_assoc()['Id'];

$result = $conn->query("SELECT Id FROM Autor WHERE Nombre = 'Isaac' AND Apellidos = 'Asimov'");
$id_Asimov = $result->fetch_assoc()['Id'];

$sql = "INSERT INTO Libro (Titulo, Fecha_publicacion, Id_Autor) VALUES
('El Hobbit', '1937-09-21', $id_Tolkien),
('La comunidad del anillo', '1954-07-20', $id_Tolkien),
('Las dos torres', '1954-11-11', $id_Tolkien),
('El retorno del rey', '1955-10-20', $id_Tolkien),
('Un guijarro en el cielo', '1950-01-15', $id_Asimov),
('Fundación', '1951-06-01', $id_Asimov),
('Yo, Robot', '1950-12-02', $id_Asimov)
";

if(($conn->query($sql)) === TRUE) {
    echo "<p>Se han insertado los datos en la tabla Libro correctamente.</p>";
}else{
    echo "<p>Error al insertar los datos en la tabla Libro: ".$conn->error."</p>";
}

$conn->close();
```

Activar Windows

Si todo ha salido bien deberíamos obtener lo siguiente en el navegador y en HeidiSQL:

CREACION DE BASE DE DATOS Y TABLAS

La tabla Autor se ha creado correctamente

La tabla Libro se ha creado correctamente

Se han insertado los datos en la tabla Autor correctamente.

Se han insertado los datos en la tabla Libro correctamente.

Filtro de bases de c Filtro de tablas

Unnamed Base de datos: libros Tabla: autor Datos

libros.autor: 2 filas en total (exacto)

#	Id	Nombre	Apellidos	Nacionalidad
1	1	J.R.R.	Tolkien	Inglesa
2	2	Isaac	Asimov	Rusa

base_datos_segundo
information_schema
libros 48,0 KiB
autor 16,0 KiB
libro 32,0 KiB
mi_base

Filtro de bases de c Filtro de tablas

Unnamed Base de datos: libros Tabla: libro Datos Consulta

libros.libro: 7 filas en total (exacto)

#	Id	Título	Fecha_publicacion	Id_Autor
1	1	El Hobbit	1937-09-21	1
2	2	La comunidad del anillo	1954-07-20	1
3	3	Las dos torres	1954-11-11	1
4	4	El retorno del rey	1955-10-20	1
5	5	Un guijarro en el cielo	1950-01-15	2
6	6	Fundación	1951-06-01	2
7	7	Yo, Robot	1950-12-02	2

base_datos_segundo
information_schema
libros 48,0 KiB
autor 16,0 KiB
libro 32,0 KiB
mi_base
mysql
performance_schema
phpmyadmin
test

2. RA6_e.

- **Añadir a la clase Libros los siguientes métodos:**
 - **consultarAutores(conexion, id autor):** Retorna un objeto con el id de autor solicitado. Se muestran todos los autores si no se pasa el id del autor. NULL si hay algún error.
 - **consultarLibros(conexion, id autor):** Retorna un objeto con los libros del id de autor solicitado. Se muestran todos los libros si no se pasa el id del autor. NULL si hay algún error.
 - **consultarDatosLibro(conexion, id libro):** Retorna un objeto con los datos del id del libro solicitado. NULL si hay algún error.
 - En el foro de la tarea se suministra las pruebas de phpUnit que debe satisfacer el código.
- **Crear una página web que muestre por pantalla para cada autor (utilizar la clase Libros):**
 - Los datos del autor, y los libros publicados.

Hemos creado un formulario para poder hacer las diferentes búsquedas que nos pide el enunciado del ejercicio.

```
<form action="" method="post">
  <ul>
    <li>
      <label for="ID_busqueda">Buscar un autor por su id: </label>
      <input type="number" id="ID_busqueda" name="ID_busqueda">
    </li>
    <li>
      <label for="ID_libros">Buscar un libro por el id de su autor: </label>
      <input type="number" id="ID_libros" name="ID_libros">
    </li>
    <li>
      <label for="ID_unico_libro">Buscar un libros por su id: </label>
      <input type="number" id="ID_unico_libro" name="ID_unico_libro">
    </li>
  </ul>
  <input type="submit" value="Buscar">
</form>
<br><br>
```


Para consultar los autores hemos escrito el siguiente código:

```
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (!empty($_POST['ID_busqueda'])) {
        $id_solicitado = $_POST['ID_busqueda'];
        $result = $conn->query("SELECT * FROM Autor WHERE Id = $id_solicitado");
        if ($result->num_rows > 0) {
            $autor = $result->fetch_assoc();
            echo "<p>El autor con ID $id_solicitado es: $autor[Nombre] $autor[Apellidos], de nacionalidad $autor[Nacionalidad]</p>";
        } else {
            echo "<h3 id='nulo'><strong>NULL</strong></h3>";
        }
    }
}
```

Para consultar libros:

```
if (!empty($_POST['ID_libros'])) {
    $id_autor = $_POST['ID_libros'];
    $result = $conn->query("SELECT * FROM Libro WHERE Id_Autor = $id_autor");
    if ($result->num_rows > 0) {
        echo "<p>Los libros del autor con ID $id_autor son:</p><ul>";
        while ($libro = $result->fetch_assoc()) {
            echo "<li>$libro[Titulo] (Fecha de publicación: $libro[Fecha_publicacion])</li>";
        }
        echo "</ul>";
    } else {
        echo "<h3 id='nulo'><strong>NULL</strong></h3>";
    }
}
```

Y para los datos de un libro específico:

```
if (!empty($_POST['ID_unico_libro'])) {
    $id_unico_libro = $_POST['ID_unico_libro'];
    $result = $conn->query("SELECT * FROM Libro WHERE Id = $id_unico_libro");
    if ($result->num_rows > 0) {
        $libro = $result->fetch_assoc();
        echo "<p>El libro solicitado es:</p>";
        echo "<p>Título: $libro[Titulo] | Fecha de publicación: $libro[Fecha_publicacion]</p>";
    } else {
        echo "<h3 id='nulo'><strong>NULL</strong></h3>";
    }
} else {
    echo "<h3><strong>Introduzca una consulta</strong></h3>";
}
```

Cuando se produce un error:

Consultas de libros y autores

Consultas

- Buscar un autor por su id:
- Buscar un libro por el id de su autor:
- Buscar un libros por su id:

NULL

A continuación, vamos a poner capturas para cada una de las búsquedas:

Consultas de libros y autores

Consultas

- Buscar un autor por su id:
- Buscar un libro por el id de su autor:
- Buscar un libros por su id:

El autor con ID 1 es: J.R.R. Tolkien, de nacionalidad Inglesa

Consultas de libros y autores

Consultas

- Buscar un autor por su id:
- Buscar un libro por el id de su autor:
- Buscar un libros por su id:

Los libros del autor con ID 2 son:

- Un guijarro en el cielo (Fecha de publicación: 1950-01-15)
- Fundación (Fecha de publicación: 1951-06-01)
- Yo, Robot (Fecha de publicación: 1950-12-02)

Consultas de libros y autores

Consultas

- Buscar un autor por su id:
- Buscar un libro por el id de su autor:
- Buscar un libros por su id:

El libro solicitado es:

Título: Fundación | Fecha de publicación: 1951-06-01

3. RA6_f.

- **Añadir a la clase Libros los siguientes métodos:**
 - ***borrarAutor(conexion, id autor):*** Elimina el autor solicitado por id. Retorna verdadero si se ha tenido éxito, falso si hay algún error.
 - ***borrarLibro(conexion, id libro):*** Elimina el libro solicitado por id. Retorna verdadero si se ha tenido éxito, falso si hay algún error.
 - ***En el foro de la tarea se suministra las pruebas de phpUnit que debe satisfacer el código.***
- **Documentación del proyecto:** Utilizando la sintaxis básica de PHPDoc, comentar todos los métodos creados. Crear un comentario ***/** */*** en cada función que incluya un comentario sobre la función
 - ***Utilizar @param para describir los parámetros que recibe la función y @return para comentar el valor resuelto***
 - ***Se puede verificar el correcto funcionamiento de los comentarios creados usando PHPDocumentor.***

```

if (!empty($_POST['Borrar_autor'])) {
    $Borrar_autor = $_POST['Borrar_autor'];
    $conn->query("DELETE FROM Autor WHERE Id = $Borrar_autor");
    if ($conn->affected_rows > 0) {
        echo "<p>El autor ha sido borrado.</p>";
    } else {
        echo "<h3 id='nulo'><strong>NULL</strong></h3>";
    }
}

if (!empty($_POST['Borrar_libro'])) {
    $Borrar_libro = $_POST['Borrar_libro'];
    $conn->query("DELETE FROM Libro WHERE Id = $Borrar_libro");
    if ($conn->affected_rows > 0) {
        echo "<p>El libro ha sido borrado.</p>";
    } else {
        echo "<h3 id='nulo'><strong>NULL</strong></h3>";
    }
}
} else {
    echo "<h3><strong>Introduzca una consulta</strong></h3>";
}

```

```
if ($_SERVER['REQUEST_METHOD'] === 'POST') {  
  
    /**  
    * Busca un autor por su ID y muestra su información.  
    *  
    * @param int $id_solicitado El ID del autor a buscar.  
    * @return void  
    */  
  
    if (!empty($_POST['ID_busqueda'])) {  
        $id_solicitado = $_POST['ID_busqueda'];  
        $result = $conn->query("SELECT * FROM Autor WHERE Id = $id_solicitado");  
        if ($result->num_rows > 0) {  
            $autor = $result->fetch_assoc();  
            echo "<p>El autor con ID $id_solicitado es: $autor[Nombre] $autor[Apellido]</p>";  
        } else {  
            echo "<h3 id='nulo'><strong>NULL</strong></h3>";  
        }  
    }  
}  
  
/**  
* Busca todos los libros de un autor por su ID.  
*  
* @param int $id_autor El ID del autor cuyos libros se buscan.  
* @return void  
*/
```