



CICLO: DAW
MÓDULO DE DESARROLLO WEB
ENTORNO SERVIDOR

Tarea N° 06

Alumno:
Edward-Ionut, Bunoaica
Y1963355C

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. RA06_D	2
2. RA06_E.....	5
3. RA06_F.....	7

(Una vez realizado el informe, no olvidar actualizar esta tabla del índice **(F9 + Actualizar toda la tabla)**, con el fin de que se actualicen todos los epígrafes y números de página)

1. RA06_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.

Crear una tabla Libro que contenga los siguientes campos:

- Una columna id de tipo entero.
- Una columna titulo 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.

```
1 <?php
2
3 $conn = new mysqli(hostname: "localhost", username: "root", password: "foc_formacion", database: "libros");
4
5 if ($conn->connect_error) {
6     die("Conexión fallida: " . $conn->connect_error);
7 }
8
9 $sql_autor = "CREATE TABLE Autor (
10     id INT AUTO_INCREMENT PRIMARY KEY,
11     nombre VARCHAR(15) NOT NULL,
12     apellidos VARCHAR(25) NOT NULL,
13     nacionalidad VARCHAR(10) NOT NULL
14 );";
15
16 if ($conn->query(query: $sql_autor) === TRUE) {
17     echo "Tabla Autor creada exitosamente";
18 } else {
19     echo "Error al crear la tabla Autor: " . $conn->error;
20 }
21
22 $sql_libro = "CREATE TABLE Libro (
23     id INT AUTO_INCREMENT PRIMARY KEY,
24     titulo VARCHAR(50) NOT NULL,
25     f_publicacion DATE NOT NULL,
26     id_autor INT,
27     FOREIGN KEY (id_autor) REFERENCES Autor(id) ON DELETE CASCADE
28 );";
29
30 if ($conn->query(query: $sql_libro) === TRUE) {
31     echo "Tabla Libro creada exitosamente";
32 } else {
33     echo "Error al crear la tabla Libro: " . $conn->error;
34 }
35
36 $conn->close();
37
```

Unamed\libros\ - HeidiSQL Portable 12.6.0.6765

Archivo Editar Buscar Consulta Herramientas Ira Ayuda

Filtro de ba Filtro de tal Host: 127.0.0.1 Base de datos: libros Consulta*

Nombre	Filas	Tamaño	Creado	Actualizado	Motor	Comentario	Tipo
autor	0	16,0 KiB	2025-01-13 20:26:01		InnoDB		Table
libro	0	32,0 KiB	2025-01-13 20:26:01		InnoDB		Table

Unamed

- base1
- informati...
- libros 48,0 KiB
 - autor 16,0 KiB
 - libro 32,0 KiB
- mydb
- mysql
- performa...
- phpmyad...
- test

EXPLORER

TAREA

- creacionTablas.php
- DWES_EDWARD_IONUT_BUNOAICA...
- insercionesLibros.php
- insercionesTablas.php

insercionesTablas.php

```

1 <?php
2
3 $conn = mysqli_connect(hostname: "localhost", username: "root", password: "foc_formacion", database: "libros");
4
5 if (!$conn) {
6     die("Error de conexión: " . mysqli_connect_error());
7 }
8
9 $query1 = "INSERT INTO libros (id, nombre, apellidos, nacionalidad) VALUES
10 (0, 'J. R. R.', 'Tolkien', 'Inglaterra')";
11
12 $query2 = "INSERT INTO libros (id, nombre, apellidos, nacionalidad) VALUES
13 (1, 'Isaac', 'Asimov', 'Rusia')";
14
15 if (mysqli_query(mysqli: $conn, query: $query1)) {
16     echo "Fila 1 insertada correctamente";
17 } else {
18     echo "Error al insertar fila 1: " . mysqli_error(mysqli: $conn);
19 }
20
21 if (mysqli_query(mysqli: $conn, query: $query2)) {
22     echo "Fila 2 insertada correctamente";
23 } else {
24     echo "Error al insertar fila 2: " . mysqli_error(mysqli: $conn);
25 }
26
27 mysqli_close(mysqli: $conn);
28
29
  
```

Unamed\libros\autor\ - HeidiSQL Portable 12.6.0.6765

Archivo Editar Buscar Consulta Herramientas Ira Ayuda

Filtro de ba Filtro de tal Host: 127.0.0.1 Base de datos: libros Tabla: autor Datos Consulta*

libros.autor: 2 filas en total (exact)

#	id	nombre	apellidos	nacionalidad
1	1	J. R. R.	Tolkien	Inglaterra
2	2	Isaac	Asimov	Rusia

Unamed

- base1
- informati...
- libros 48,0 KiB
 - autor 16,0 KiB
 - libro 32,0 KiB
- mydb
- mysql
- performa...
- phpmyad...
- test

```

1 <?php
2
3 $conn = new mysqli(hostname: "localhost", username: "usuario", password: "contraseña", database: "nombre_base_de_datos");
4
5 if ($conn->connect_error) {
6     die("Error de conexión: " . $conn->connect_error);
7 }
8
9 $query_libros = "
10 INSERT INTO libro (id, titulo, f_publicacion, id_autor) VALUES
11 (0, 'El Hobbit', '1937-09-21', 0),
12 (1, 'La Comunidad del Anillo', '1954-07-29', 0),
13 (2, 'Las dos torres', '1954-11-11', 0),
14 (3, 'El retorno del Rey', '1955-10-20', 0),
15 (4, 'Un guijarro en el cielo', '1950-01-19', 1),
16 (5, 'Fundación', '1951-06-01', 1),
17 (6, 'Yo, robot', '1950-12-02', 1);
18 ";
19
20 if ($conn->query($query_libros) === TRUE) {
21     echo "Datos insertados correctamente en la tabla libro.<br>";
22 } else {
23     echo "Error al insertar datos en la tabla libro: " . $conn->error . "<br>";
24 }
25
26 $conexion->close();
27
28

```

Unnamed\libros\libro - HeidiSQL Portable 12.6.0.6765

Archivo Editar Buscar Consulta Herramientas Ira Ayuda

Filtro de base de datos Filtro de tabla Host: 127.0.0.1 Base de datos: libros Tabla: libro Datos Consulta*

libros.libro: 7 filas en total (exact)

#	id	titulo	f_publicacion	id_autor
1	1	El Hobbit	1937-09-21	1
2	2	La Comunidad del Anillo	1954-07-29	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	1950-01-19	2
6	6	Fundación	1951-06-01	2
7	7	Yo, robot	1950-12-02	2

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.

```

1 <?php
2
3 class Libros {
4
5     public static function conexion($servidor, $base_de_datos, $usuario, $contraseña): mysqli|null {
6         $conexion = new mysqli(hostname: $servidor, username: $usuario, password: $contraseña, database: $base_de_datos);
7
8         if ($conexion->connect_error) {
9             return null;
10        }
11
12        return $conexion;
13    }
14 }

```

2. RA06_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.

```

class Libros {
    14
    15
    16
    17
    18
    19
    20
    21
    22
    23
    24
    25
    26
    27
    28
    29
    30
    31
    32
    33
    34
    35
    36
    37
    38
    39
    40
    41
    42
    43
    44
    45
    46
    47
    48
    49

    public function consultarAutores($conexion, $id_autor = null): array|null {
        $query = "SELECT * FROM autores";

        if ($id_autor) {
            $query .= " WHERE id_autor = ?";
        }

        $stmt = $conexion->prepare($query);

        if ($id_autor) {
            $stmt->bind_param("i", $id_autor);
        }

        if (!$stmt->execute()) {
            return null;
        }

        $result = $stmt->get_result();
        $autores = [];

        while ($row = $result->fetch_object()) {
            $autores[] = $row;
        }

        return empty($autores) ? null : $autores;
    }

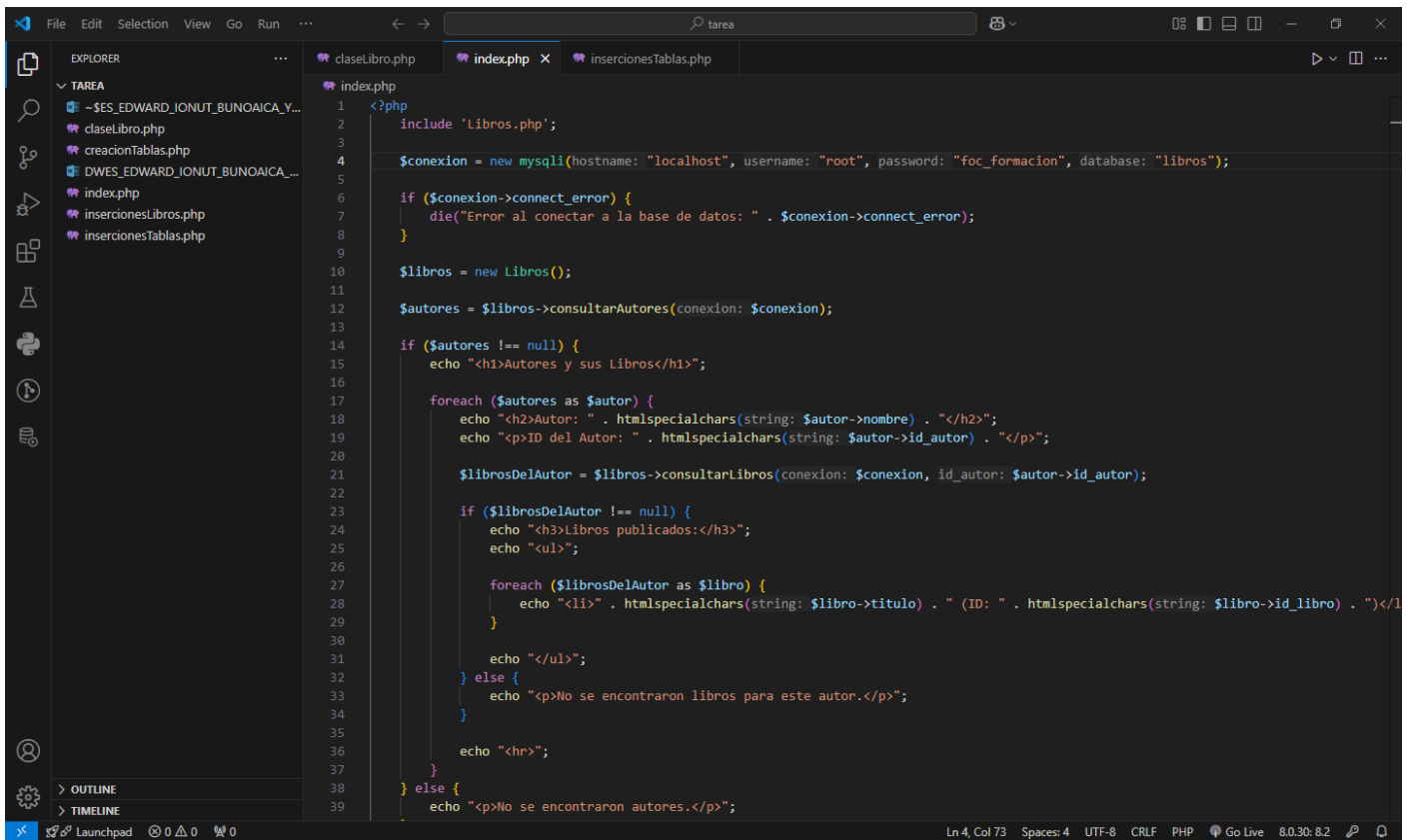
    public function consultarLibros($conexion, $id_autor = null): array|null {
        $query = "SELECT * FROM libros";

        if ($id_autor) {
            $query .= " WHERE id_autor = ?";
        }

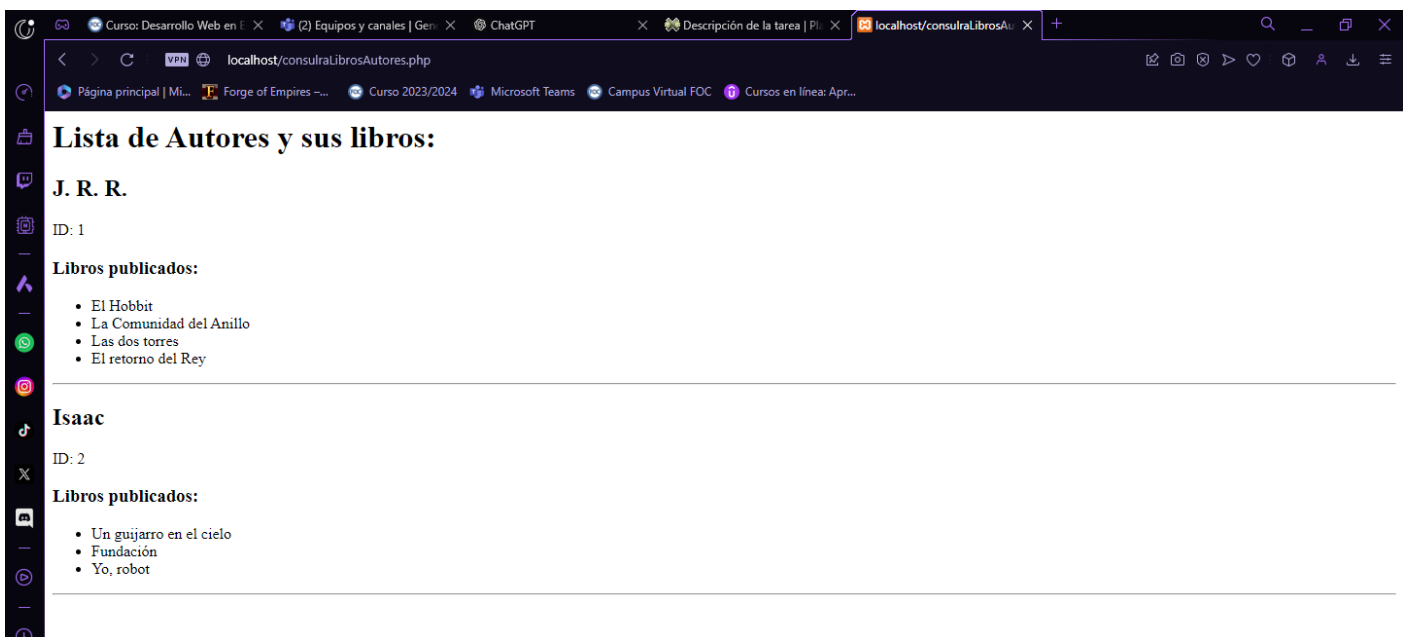
        $stmt = $conexion->prepare($query);
  
```

Crear una página web que muestre por pantalla para cada autor (utilizar la clase Libros):

- Los datos del autor, y los libros publicados.



```
1 <?php
2 include 'Libros.php';
3
4 $conexion = new mysqli(hostname: "localhost", username: "root", password: "foc_formacion", database: "libros");
5
6 if ($conexion->connect_error) {
7     die("Error al conectar a la base de datos: " . $conexion->connect_error);
8 }
9
10 $libros = new Libros();
11
12 $autores = $libros->consultarAutores(conexion: $conexion);
13
14 if ($autores != null) {
15     echo "<h1>Autores y sus Libros</h1>";
16
17     foreach ($autores as $autor) {
18         echo "<h2>Autor: " . htmlspecialchars(string: $autor->nombre) . "</h2>";
19         echo "<p>ID del Autor: " . htmlspecialchars(string: $autor->id_autor) . "</p>";
20
21         $librosDelAutor = $libros->consultarLibros(conexion: $conexion, id_autor: $autor->id_autor);
22
23         if ($librosDelAutor != null) {
24             echo "<h3>Libros publicados:</h3>";
25             echo "<ul>";
26
27             foreach ($librosDelAutor as $libro) {
28                 echo "<li>" . htmlspecialchars(string: $libro->titulo) . " (ID: " . htmlspecialchars(string: $libro->id_libro) . ")</li>";
29             }
30
31             echo "</ul>";
32         } else {
33             echo "<p>No se encontraron libros para este autor.</p>";
34         }
35
36         echo "<hr>";
37     }
38 } else {
39     echo "<p>No se encontraron autores.</p>";
40 }
```



3. RA06_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.



```

28     return empty($autores) ? null : $autores;
29 }
30
31 0 references | 0 overrides
32 public function borrarAutor($conexion, $id_autor): bool {
33     $queryLibros = "DELETE FROM libro WHERE id_autor = ?";
34     $stmtLibros = $conexion->prepare($queryLibros);
35     $stmtLibros->bind_param("i", $id_autor);
36
37     if (!$stmtLibros->execute()) {
38         return false;
39     }
40
41     $queryAutor = "DELETE FROM autor WHERE id = ?";
42     $stmtAutor = $conexion->prepare($queryAutor);
43     $stmtAutor->bind_param("i", $id_autor);
44
45     if ($stmtAutor->execute()) {
46         return true;
47     } else {
48         return false;
49     }
50 }
51
52 0 references | 0 overrides
53 public function borrarLibro($conexion, $id_libro): bool {
54     $query = "DELETE FROM libro WHERE id = ?";
55     $stmt = $conexion->prepare($query);
56     $stmt->bind_param("i", $id_libro);
57
58     if ($stmt->execute()) {
59         return true;
60     } else {
61         return false;
62     }
63 }

```

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.
 - Descargar el fichero `phpdocumentor.phar` de la página de PHPDocumentor
 - Crear un directorio `doc` fuera del directorio donde se esté creando la página web
 - Ejecutar `c:\xampp\php\bin\php.exe phpDocumentor.phar -d c:\directorio de trabajo -t c:\directorio de documentacion creado`. Estos directorios son de ejemplo..
 - Tras esto se debería haber creado una página web con el contenido de la documentación.

```

1  <?php
2  /**
3   * Clase que gestiona las operaciones relacionadas con los libros y autores
4   */
5  class Libros {
6
7      /**
8       * Consulta los autores registrados en la base de datos.
9       *
10      * Si se proporciona un id_autor, solo se consultará ese autor específico.
11      * Si no se proporciona, se consultarán todos los autores
12      *
13      * @param mysqli $conexion Conexión a la base de datos
14      * @param int|null $id_autor El ID del autor
15      *
16      * @return array|null Un arreglo con los autores encontrados o null si no hay resultados o ocurre un error
17      */
18      public static function consultarAutores($conexion, $id_autor = null): array|null {
19          $query = "SELECT * FROM autor";
20
21          if ($id_autor) {
22              $query .= " WHERE id = ?";
23          }
24
25          $stmt = $conexion->prepare(query: $query);
26
27          if ($id_autor) {
28              $stmt->bind_param(types: "i", var: &$id_autor);
29          }
30
31          if (!$stmt->execute()) {
32              return null;
33          }
34
35          $result = $stmt->get_result();
36          $autores = [];
37
38          while ($row = $result->fetch_object()) {

```

Documentation

Search (Press "/" to focus)

Packages

- Application

Reports

- Deprecated
- Errors
- Markers

Indices

- Files

Documentation

Table of Contents

Packages

- [Application](#)

Classes

- [Libros](#)

Clase que gestiona las operaciones relacionadas con los libros y autores

