

1 INTRODUCCIÓN

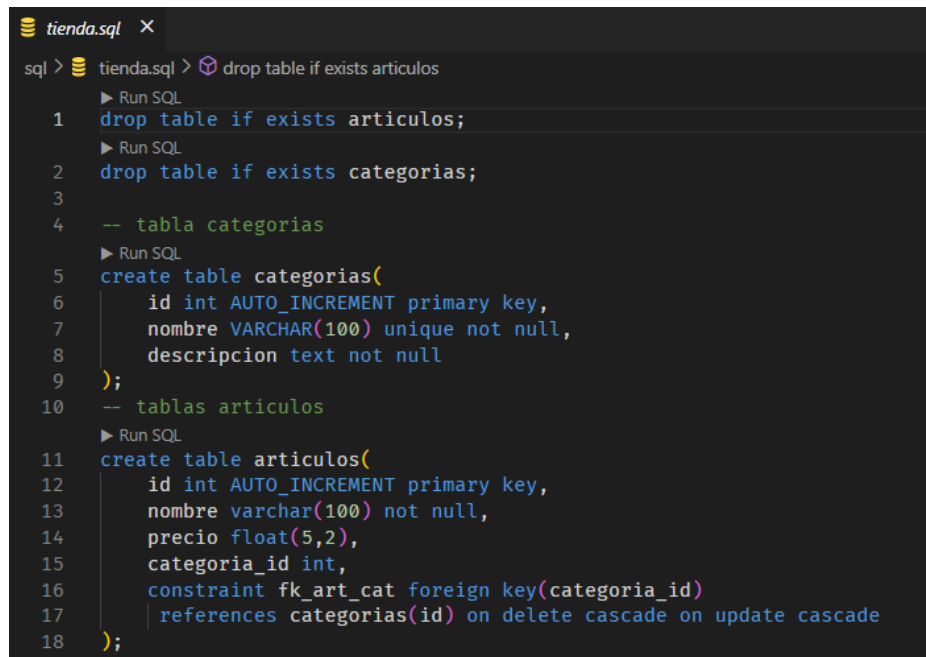
En este proyecto he elaborado una web programada en PHP para una tienda. En dicha web podemos crear Artículos con distintos atributos, además de Categorías en base a estos dos elementos girará nuestra página.

Podremos crear, actualizar, borrar, y leer Artículos y/o categorías que se encuentren en la base de datos de nuestra tienda. Además, para los artículos podremos ver un detalle mayor de cada artículo, así como filtrar por distintas características del mismo.

2 CÓDIGO FUENTE

2.1 BASE DE DATOS

Tenemos la siguiente estructura de tablas en la BD.



```
tienda.sql x
sql > tienda.sql > drop table if exists articulos
  ▶ Run SQL
1 drop table if exists articulos;
  ▶ Run SQL
2 drop table if exists categorias;
3
4 -- tabla categorias
  ▶ Run SQL
5 create table categorias(
6     id int AUTO_INCREMENT primary key,
7     nombre VARCHAR(100) unique not null,
8     descripcion text not null
9 );
10 -- tablas articulos
  ▶ Run SQL
11 create table articulos(
12     id int AUTO_INCREMENT primary key,
13     nombre varchar(100) not null,
14     precio float(5,2),
15     categoria_id int,
16     constraint fk_art_cat foreign key(categoria_id)
17     references categorias(id) on delete cascade on update cascade
18 );
```

2.2 CLASES

2.2.1 Conexión

Contiene el código necesario para conectarse a la Base de Datos, previamente se deber haber rellenado el archivo **conf.sample** y renombrarlo a .config con los datos de sesion de la base de datos.

```

1  <?php
2
3  namespace Src;
4  use PDO;
5  use PDOException;
6
7  class Conexion
8  {
9      //Todos los objetos que se instancien van a tener este atributo
10     protected static $conexion;
11
12
13     public function __construct()
14     {
15         //Con esto garantizamos que solo tenemos una conexion a la BD
16         if (self::$conexion == null) {
17             self::crearConexion();
18         }
19     }
20
21     public static function crearConexion()
22     {
23         $fichero=dirname(__DIR__,1)."\.config";
24         $opciones=parse_ini_file($fichero);
25         $dbname=$opciones['dbname'];
26         $host=$opciones['host'];
27         $usuario=$opciones['user'];
28         $pass=$opciones['pass'];
29         //Descriptor del nombre de servicio
30         $dns="mysql:host=$host;dbname=$dbname;charset=utf8mb4";
31         //Creamos la conexion
32
33         try{
34             self::$conexion=new PDO($dns,$usuario,$pass);
35             //Solo en el desarrollo
36             self::$conexion->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
37         }catch(PDOException $ex){
38             die("Error en la conexión: ".$ex->getMessage());
39         }
40     }
41 }
42

```

2.2.2 Artículos

En esta clase tenemos funciones que desempeñan distintos trabajos como crear, actualizar, leer...

- Con esta función borramos un registro

```

//Borrar un registro en funcion del ID pasado por parámetro
// *****
public function borra($id)
{
    $sql = "DELETE FROM articulos WHERE id=$id";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute([]);
    } catch (PDOException $ex) {
        die("Error al borrar registro... " . $ex->getMessage());
    }
    parent::$conexion = null;
}
// *****

```

- Con esta función creamos un registro

```
//Esta funcion inserta elementos en nuestra BD
// *****
public function crear()
{
    $sql = "INSERT INTO articulos (nombre, precio, categoria_id) VALUES (:n,:p,:c)";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute([
            ':n' => $this->nombre,
            ':p' => $this->precio,
            ':c' => $this->categoria_id
        ]);
    } catch (PDOException $ex) {
        die("Error al insertar registro ... " . $ex->getMessage());
    }
    parent::$conexion = null;
}
// *****
```

- Con esta función leemos todos los registros de la base de datos

```
//Esta funcion lee todos los registros de nuestra BD
// *****
public function leer_todos()
{
    $sql = "SELECT * FROM articulos";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error al devolver todos los registros" . $ex->getMessage());
    }
    parent::$conexion = null;
    return $stmt;
}
//***** */
```

- Esta función actualiza un elemento de nuestra base de datos

```
//Esta funcion inserta elementos en nuestra BD
// *****
public function actualizar($id)
{
    $sql = "UPDATE articulos SET nombre=:n, precio=:p, categoria_id=:c WHERE id={$id}";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute([
            ':n' => $this->nombre,
            ':p' => $this->precio,
            ':c' => $this->categoria_id
        ]);
    } catch (PDOException $ex) {
        die("Error al actualizar registro ... " . $ex->getMessage());
    }
    parent::$conexion = null;
}
}
```

- Esta función devuelve el próximo registro (que aun no existe) para un nuevo artículo

```
//Traer un registro
public function proximoid()
{
    $sql = "SELECT * FROM articulos";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
        while ($ids = $stmt->fetch(PDO::FETCH_OBJ)) {
            $proxid = $ids->id;
        }
        $proxid = $proxid + 1;
    } catch (PDOException $ex) {
        die("Error en la devolución del ID" . $ex->getMessage());
    }
    return $proxid;
}
```

- Generamos artículos usando la librería Faker

```
//Usando la libreria Faker creamos datos que insertaremos en la BD
public function generame($num)
{
    $categoria = (new Categorías)->devuelvecategorias();

    for ($i = 0; $i < $num; $i++) {
        $nombre = Faker\Factory::create('es_ES')->sentence(1);
        $precio = Faker\Factory::create('es_ES')->randomFloat($nbMaxDecimals = 2, $min = 1, $max = 300);
        $id_categoria = $categoria[array_rand($categoria, 1)];
        (new Artículos)->setNombre($nombre)->setPrecio($precio)->setCategoria_id($id_categoria)->crear();
    }
}
```

- Función que devuelve la cantidad de registros existente en nuestra base de datos

```
//Esta función devuelve el número de registros que existen en nuestra BD
public function contarregistros()
{
    $sql = "SELECT * FROM articulos";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
        $num = $stmt->rowCount();
    } catch (PDOException $ex) {
        die("Error al devolver el número de registros ... " . $ex->getMessage());
    }
    return $num;
}
```

- Función que devuelve para un id todos sus valores

```
//Traer un registro
public function leeelemento($id)
{
    $sql = "SELECT * FROM articulos WHERE id=$id";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error en la lectura del ID: $id" . $ex->getMessage());
    }
    return $stmt->fetch(PDO::FETCH_OBJ);
}
```

- Función que comprueba si existe algún registro con ese ID

```
//Con esta función devolvemos true si existe ese valor pasado
//o false sino existe
public function existeid($id)
{
    $sql = "SELECT * FROM articulos WHERE id=$id";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error en la lectura del ID: $id" . $ex->getMessage());
    }
    return ($stmt->rowCount() == 1);
}
```

- Funciones que filtran con el valor de un campo; estas dos funciones (una depende de la otra) filtrarán los elementos que deseemos

```
//Con el ID del artículo y la columna a filtrar nos devolver el valor a partir del que filtraremos
public function valorafiltrar($categoria, $id)
{
    $sql = "SELECT $categoria FROM articulos WHERE id=$id";
    $stmt = parent::$conexion->prepare($sql);
    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error al devolver el filtro ... " . $ex->getMessage());
    }
    parent::$conexion = null;
    return $stmt->fetch(PDO::FETCH_OBJ);
}

// Usando el valor que retorna la función anterior y la columna para la que queremos filtrar
// devolvemos el STMT para el que posteriormente haremos un fetch(PDO::FETCH_OBJ)
public function filtrar($campo, $valor)
{
    $sql = "SELECT * FROM articulos WHERE {$campo}=:'v'";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute([
            ':v' => $valor
        ]);
    } catch (PDOException $ex) {
        die("Error al filtrar " . $ex->getMessage());
    }
    parent::$conexion = null;
    return $stmt;
}
```

2.2.3 Categorías

- Las funciones de crear, actualizar, leer y borrar son iguales a las anteriores, únicamente modificando el nombre de la tabla en la consulta (CRUD)
- Función que nos devuelve todos los nombres e ids

```
//Funcion que nos devuelve el nombre y el id de las categorías (para poder usarlos en creacion de artículos)
public function devuelveNombre()
{
    $sql = "SELECT id, nombre FROM categorias";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error al comprobar si la base de datos está vacía");
    }
    parent::$conexion = null;
    return $stmt;
}
```

- Función que Genera elementos

```
//Con esta función generamos en nuestra base de datos elementos para tabla categorías
public function categorias_generator($number)
{
    //Si la base de datos esta vacía, generamos con Faker datos
    if ($this->bdvacía()) {
        for ($i = 0; $i < $number; $i++) {
            $nombre = Faker\Factory::create('es_ES')->unique()->sentence(1);
            $description = Faker\Factory::create('es_ES')->text(100);
            (new Categorías)->setDescription($description)->setNombre($nombre)->crear();
        }
    }
}
```

- Función que devuelve si la BD está vacía, o no

```
//Esta función nos devolverá true (si la base de datos esta vacía) o falso si ya contiene datos.
public function bdvacía()
{
    $sql = "SELECT * FROM categorías";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
        $filas = $stmt->rowCount();
        if ($filas == 0) {
            $bool = true;
        } else {
            $bool = false;
        }
    } catch (PDOException $ex) {
        die("Error al comprobar si la base de datos está vacía");
    }
    parent::$conexion = null;
    return $bool;
}
```

- Función que comprueba que el nombre introducido es único

```
public function uniquenombre($nombre)
{
    if (!isset($this->id)) {
        $sql = "SELECT * FROM categorías WHERE id≠{$this->id} AND nombre=:n";
    } else {
        $sql = "SELECT * FROM categorías WHERE nombre=:n";
    }
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute([
            ':n' => $nombre
        ]);
        $filas = $stmt->rowCount();
    } catch (PDOException $ex) {
        die("Error al comprobar Nombre unico ... " . $ex->getMessage());
    }
    parent::$conexion = null;
    return $filas;
}
```

- Funcion que devuelve todos los ids de categorias en un array

```
//Esta funcion la usaremos en el Faker de Articulos para generar arrays ids aleatorios con los valores ya creados
public function devuelvecategorias()
{
    $sql = "SELECT id FROM categorias";
    $stmt = parent::$conexion->prepare($sql);
    try {
        $stmt->execute([]);
    } catch (PDOException $ex) {
        die("Error al comprobar Nombre unico ... " . $ex->getMessage());
    }
    $id = [];
    while ($fila = $stmt->fetch(PDO::FETCH_OBJ)) {
        $id[] = $fila->id;
    }
    parent::$conexion = null;
    return $id;
}
```

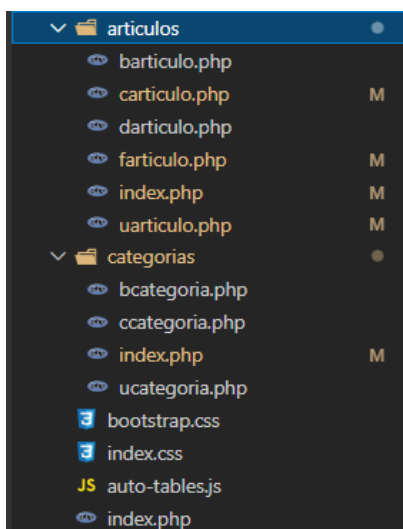
- Funcion que nos devuelve unicamente el nombre y el id de una categoria para la que hemos pasado el id

```
//Funcion que nos devuelve el nombre y el id de las categorias (para poder usarlos en articulos)
public function nombre($id)
{
    $sql = "SELECT nombre FROM categorias WHERE id=$id";
    $stmt = parent::$conexion->prepare($sql);

    try {
        $stmt->execute();
    } catch (PDOException $ex) {
        die("Error al comprobar si la base de datos está vacía");
    }
    parent::$conexion = null;
    return $stmt;
}
```

2.3 PÁGINAS PHP

La estructura de archivos de nuestra web es la siguiente:



- **Index.php** (Página principal que nos da a elegir si queremos ver Articulos o Categorías)
- **Carpeta articulos**
 - Index.php -> Muestra todos los artículos en una tabla
 - barticulo.php -> Borra un articulo y redirige al index
 - carticulo.php -> Muestra un formulario para crear un artículo
 - darticulo.php -> Muestra detalles del artículo
 - farticulo.php -> Muestra una tabla con los artículos filtrados
 - uarticulo.php -> Muestra formulario para actualizar un articulo
- **Carpeta categorias**
 - Index.php -> Muestra todas las categorias en una tabla
 - bcategoria.php -> Borra una categoria y redirige al index
 - ccategoria.php -> Muestra un formulario para crear una categoría
 - ucategoria.php -> Muestra formulario para actualizar una categoria

3 FUNCIONAMIENTO PÁGINA WEB

3.1 INDEX PRINCIPAL



3.2 ARTICULOS

ARTICULOS					
Barra Navegación Inicio Crear Artículo Categorías					
TABLA ARTICULOS					
Show 10 entries		Search:			
ID	Nombre artículo	Precio	Categoría	Acciones	
2	Veritatis sit.	274.32 €	19		
3	Hic sint.	42.65 €	13		
4	Quas.	46.04 €	16		
5	Numquam nihil.	69.59 €	17		
6	Ut.	269.03 €	12		
7	Ut nobis.	223.73 €	12		
8	Aut.	182.70 €	6		

3.2.1 Crear artículo

CREAR ARTÍCULO

ID

NOMBRE ARTÍCULO

PRECIO ARTÍCULO

CATEGORÍA

ARTICULOS

Barra Navegación Inicio Crear Artículo Categorías

Artículo creado

TABLA ARTICULOS

Show entries
 Search:

ID	Nombre artículo	Precio	Categoría	Acciones
103	Prueba	150.00 €	10	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>

Showing 1 to 1 of 1 entries (filtered from 102 total entries)

3.2.2 Editar Artículo

Actualizar Artículo 103

ID

103

Nombre Artículo

Prueba

Precio Artículo

150.00

ID Categoría

Alias ducimus.

Borrar

Actualizar



ARTICULOS

Barra Navegación Inicio Crear Artículo Categorías

TABLA ARTICULOS

Show 10 entries

Search: prueb

ID	Nombre artículo	Precio	Categoría	Acciones
103	Prueba Actualizada	150.00 €	2	 

Showing 1 to 1 of 1 entries (filtered from 102 total entries)

Previous 1 Next

3.2.3 Borrar Artículo

Barra Navegación Inicio Crear Artículo Categorías

localhost dice
Borrar Artículo 103



Aceptar

Cancelar

TABLA ARTICULOS

Show 10 entries

Search: prueb

ID	Nombre artículo	Precio	Categoría	Acciones
103	Prueba Actualizada	150.00 €	2	 

Showing 1 to 1 of 1 entries (filtered from 102 total entries)

Previous 1 Next

ARTICULOS

Barra NavegaciónInicioCrear ArtículoCategorías

ARTÍCULO 103 BORRADO CON ÉXITO

TABLA ARTICULOS

Show10▼entriesSearch:prueba✕

ID	Nombre artículo	Precio	Categoría	Acciones
No matching records found				

Showing 0 to 0 of 0 entries (filtered from 101 total entries)PreviousNext

3.2.4 Detalle Artículo

Barra NavegaciónVolver a ArtículosCrear artículo

Detalle Artículo 5



NOMBRE » Numquam nihil.

PRECIO » 69.59€

CATEGORÍA » Occaecati.

FILTRAR IGUALES POR:

Nombre Artículo▼Filtrar

3.2.5 Filtrado

Artículos con misma Categoría

FILTRO

Barra Navegación Inicio Volver a detalles Crear artículo

FILTRANDO VALORES POR "CATEGORIA_ID"

Show 10 entries Search:

ID	Nombre artículo	Precio	Categoría	Acciones
5	Numquam nihil.	69.59 €	17	 
28	Deleniti.	158.98 €	17	 
61	Sit.	236.00 €	17	 
66	Quisquam quibusdam.	7.52 €	17	 
77	Rerum.	43.92 €	17	 
92	Temporibus dolor.	163.82 €	17	 
98	Provident.	275.58 €	17	 

Showing 1 to 7 of 7 entries Previous 1 Next



















3.3 CATEGORÍAS

CATEGORÍAS

Barra Navegación Inicio Crear Categoría Artículos

TABLA CATEGORIAS

Show 10 entries Search:

ID	Nombre	Descripcion	Acciones
1	Modi.	Molestias eveniet voluptatem ullam id. Et deleniti quia sit vel. Et eligendi et nulla vel.	 
2	Alias ducimus.	Vitae minima qui quam rerum omnis laudantium ipsum. Sed in voluptatum aut ea.	 
3	Earum quo.	Qui aliquid quod ipsam ut provident cupiditate. Sit dolor repellendus quidem minima sequi aut.	 
4	Cum vel.	Aut excepturi laborum quia aut tempore. Nihil porro ea dolores et.	 
5	Aliquam.	Rerum nihil ut ullam sapiente dolorum nihil. Id qui odio blanditiis eos facilis.	 
6	Sequi.	Qui voluptatum dicta debitis quo et. Vel alias maiores sequi qui dignissimos sunt et.	 
7	Dolor delectus.	Sint ducimus distinctio animi laborum ipsam et. Voluptas voluptatem est adipisci ut dolores.	 
8	Aut esse.	Voluptatem eos ratione rerum fugiat commodi. Rem et omnis et ea fugit ad.	 
9	Omnis.	Quis ut sit deserunt hic. Doloremque et reprehenderit a. Quos facilis sit eos ea qui aperiam.	 

3.3.1 Crear Categoría

CREAR CATEGORÍA

NOMBRE CATEGORÍA

Prueba2

DESCRIPCIÓN CATEGORÍA

Esto es una prueba para la documentacion

Crear

Borrar



CATEGORÍAS

Barra Navegación Inicio Crear Categoría Artículos

Categoría creada

TABLA CATEGORIAS

Showing 10 entries Search: docume x

ID	Nombre	Descripción	Acciones
21	Prueba2	Esto Es Una Prueba Para La Documentacion	 

Showing 1 to 1 of 1 entries (filtered from 20 total entries) Previous 1 Next

3.3.2 Actualizar Categoría

ACTUALIZAR CATEGORÍA → 21,PRUEBA2

ID

21

NOMBRE CATEGORÍA

Prueba2

DESCRIPCION CATEGORÍA

Esto Es Una Prueba Para La Documentacion

Borrar Actualizar

ACTUALIZAR CATEGORÍA → 21,PRUEBA2

ID

21

NOMBRE CATEGORÍA

Prueba2 Modificada

DESCRIPCION CATEGORÍA

Esto Es Una Prueba Para La Documentacion Modificada

Borrar Actualizar



CATEGORÍAS

Barra Navegación Inicio Crear Categoría Artículos

CATEGORÍA " " MODIFICADA

TABLA CATEGORIAS

Show 10 entries Search:

ID	Nombre	Descripcion	Acciones
21	Prueba2 Modificada	Esto Es Una Prueba Para La Documentacion Modificada	 

Showing 1 to 1 of 1 entries (filtered from 20 total entries) Previous 1 Next

3.3.3 Borrar Categoría



localhost dice
Borrar Categoría 21

Barra Navegación Inicio Crear Categoría Artículos

CATEGORÍA " " MODIFICADA

TABLA CATEGORIAS

Show 10 entries Search:

ID	Nombre	Descripcion	Acciones
21	Prueba2 Modificada	Esto Es Una Prueba Para La Documentacion Modificada	 

Showing 1 to 1 of 1 entries (filtered from 20 total entries) Previous 1 Next

CATEGORÍAS

Barra Navegación Inicio Crear Categoría Artículos

CATEGORÍA "(21) Prueba2 Modificada" BORRADA

TABLA CATEGORIAS

Show 10 entries

Search: modif x

ID	Nombre	Descripcion	Acciones
No matching records found			

Showing 0 to 0 of 0 entries (filtered from 19 total entries)

Previous Next