

---

## Actividad de los temas 1 al 5

---

La idea es construir una aplicación web que sea una **tienda**. Tienes que decidir que tipo de tienda quieres definir.

Diseñaréis la base de datos en la que al menos tiene que haber una tabla de **usuarios, Imágenes, familias y productos**.

Se debe configurar un archivo .env donde se encuentren definidas las variables DB\_DSN, DB\_USERNAME y DB\_PASSWORD para realizar la conexión a una base de datos. Utilizaremos la librería externa <https://github.com/vlucas/phpdotenv>

Utiliza Composer en el proyecto, dentro del directorio public figurará el archivo php de arranque de la aplicación y dentro del directorio src figuran las carpetas asociadas a la programación orientada a objetos.

Crea una **carpeta clases** en la que se crearán las siguientes clases:

**Imagen:** las instancias de esta clase representan las imágenes que están asociadas a los productos. Cada producto tendrá asociada una imagen.

**Producto:** las instancias de esta clase representan los productos que se venden en la tienda.

- Tendremos como atributos los campos que habéis creado en vuestra tabla de la base de datos, pero al menos debe tener los que figuran en las pantallas de ejemplos que figuran a continuación. Hay una relación
- Constructor
- Métodos get

La clase Producto es implementada siguiendo el patrón PDO que sigue la interface con los siguientes métodos:

- método crear como parámetro de entrada un objeto producto y como salida un booleano
- método listar devuelve un array de objetos de los productos
- método listarPorId, tiene como parámetro de entrada el id de un producto y devuelve un objeto producto
- método borrar, tiene como parámetro de entrada el id de un producto y como salida un booleano

**Familia:** las instancias de esta clase representan las familias a las que pertenecen los productos

- Tendremos como atributos los campos que habéis creado en vuestra tabla de base de datos
- Constructor
- Métodos get

Esta tabla podemos tenerla precargada en el script a generar de la BD que crees.

**BaseDatos:** va a ser la clase encargada de interactuar con la base de datos. Debe seguir el patrón singleton.

**Usuarios:** son las instancias de los usuarios que se registran en la web para realizar la compra.

**CestaCompra:** con esta clase vas a gestionar los productos que escoge el cliente de la tienda para comprar.

- Tendremos un array como atributo
- nuevoArticulo(\$id): introduce en la cesta el artículo indicado por su código.
- getProductos: devuelve un array con todos los productos de la cesta.
- getCoste: devuelve el coste de los productos que figuran en la cesta.
- estaVacía: devuelve true o false, según la cesta esté o no vacía.

También necesitamos dos funciones para guardar la cesta en la sesión del usuario, y para recuperarla. Y programaremos otra más para mostrar el contenido de la cesta en formato HTML.

- guardaCesta: guarda la cesta en la sesión del usuario.
- carga\_cesta: recupera el contenido de la cesta de la sesión del usuario.
- muestra: genera una salida en formato HTML con el contenido de la cesta.

Realizaremos un script de base de datos para crear las tablas. En el script de base de datos también cargaremos las familias a las que pertenecen los productos, pero el resto de información del producto se introduce a través de un formulario donde tendremos que validar la información.

Creamos un fichero denominado “helper.php”, donde tendremos unas funciones genéricas de validación:

- Función validar requerido, tiene un parámetro de entrada de tipo string y devolverá un booleano, es decir, que si existe valor devuelve un true.
- Función validar numérico, tiene un parámetro de entrada de tipo string y devolverá un booleano
- Función validar longitud, tiene dos parámetros de entrada (string campo, longitud int) y devolverá un booleano si tiene ese número de caracteres.
- Función validar subida fichero, tiene como parámetro de entrada un array, comprueba si ha sido subido el fichero de la imagen o no por lo que devuelve un booleano.
- Función validar formato Imagen, tiene como parámetro de entrada un tipo string. El único formato valido es jpg.
- Función limpiar texto tendrá un parámetro de entrada y devolverá un string, La cadena de salida solo puede contener las siguientes etiquetas de html: **<strong><em>**
- Función limpiar entrada, desinfecta todos los campos de entrada.
- Función redireccionar, tiene como parámetro de entrada un tipo string. Realiza una petición http get del path recibido como parámetro.

Tenemos dos apartados diferenciados:

- Primero realizamos la parte de administración de gestión de la información de los productos a crear:

Creamos un fichero “procesa.php”, donde se realizan las validaciones de los datos del formulario de alta del producto:

- Que todos los datos son obligatorios
- Que el precio es un dato numérico
- Que la subida de la imagen ha sido correcta
- Que el formato de la imagen es correcto, solo permitido jpg
- Las imágenes se guardarán en un subdirectorío “img” que cuelga del directorío “public” del proyecto, el nombre del archivo imagen será un identificador único seguido del nombre del archivo de la imagen a subir al servidor. En el campo nombre grabamos el nombre generado y en el campo url el directorío “img/nombre de la imagen generado con su extensión”.

Los mensajes de error que se pueden dar son:

1. Por favor, rellena todos los datos
2. No se puede procesar el archivo
3. El archivo no tiene una extensión válida
4. Por favor, introduce un precio válido
5. No se ha podido guardar el producto en base de datos

Si todo es correcto, mostramos el mensaje el producto ha sido dado de alta correctamente.

Para llevar esta aplicación también tendremos en cuenta el patrón repositorio que implica definir:

- Una interfaz repositorio del grupo, con el nombre “IntRepoProducto” que tendrá:
  - método crear como parámetro de entrada un objeto producto y como salida un booleano
  - método listar devuelve un array de objetos de los productos
  - método listarPorId, tiene como parámetro de entrada el id de un producto y devuelve un objeto producto
  - método borrar, tiene como parámetro de entrada el id de un producto y como salida un booleano
- Una clase denominada “PDOPProducto”, que implementa la interfaz repositorio creada que desarrolla los métodos.

Y también vamos a implementar principio SOLID open/close, que significa nos va a permitir con cualquier proveedor de datos (pdo, mysli,mysql,...) con solo cambiar la clase desarrollada en nuestro caso PDOPProducto funciona nuestra aplicación.

- Una clase denominada “Produ”, donde en el constructor pasamos la interfaz y desarrollamos las funciones que llamarán a la interfaz.

En este primer apartado tendremos:

Index.php – que muestra el listado de los productos y con las opciones de ver toda la información del producto, borrar un producto o crear un nuevo producto

## Listado de los productos

NOMBRE	PRECIO	ACCIONES
Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos	15	<a href="#">Más información</a> <a href="#">borrar</a>

[Crear Producto](#)

Mas información – detalle.php

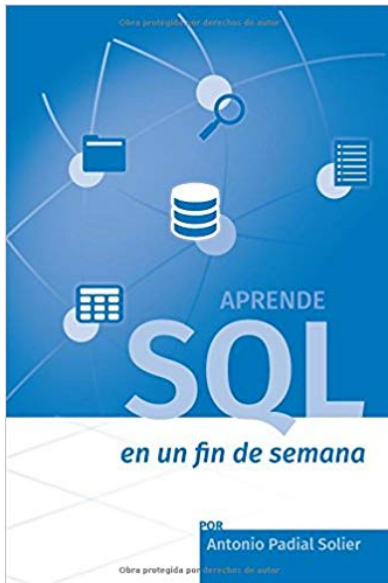
### Detalle del producto

Nombre :Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos

Precio :15

Descripción : Reseña del editor ¡Oferta de lanzamiento de la edición impresa! Sólo 4,99 € por tiempo limitado. El curso de SQL definitivo en español. Sin necesidad de conocimientos previos. Aprende a manipular y consultar bases de datos de forma rápida y sencilla. ¿Estás desarrollando una web y quieres utilizar MySQL para almacenar información? ¿estás estudiando y se te atraganta la asignatura de base de datos? ¿quieres aprender SQL para mejorar tu currículum o dar un giro a tu vida laboral? o ¿simplemente tienes curiosidad por conocer este lenguaje y sus posibilidades? A todos vosotros bienvenidos, habéis dado con el libro adecuado. Más de 100 sentencias de ejemplo, numerosos ejercicios y temas adicionales con los que aprenderás todo lo necesario para utilizar SQL en tus proyectos profesionales.

Familia : BBDD



[Volver listado](#)

Crear Producto -formularioAlta.php

## Alta de producto

Selecciona una familia: Bases de datos

No se ha seleccionado ningún archivo.

Una segunda parte que es la Tienda:

Vuestros métodos podrán ser similares a los siguientes:

- `Listar()`: devuelve un array con todos los productos de la base de datos.
- `ListarPorId($id)`: devuelve el producto que coincide con el código
- `verificarUsuario($usuario, $password)`: Verdadero o Falso dependiendo si son correctas las credenciales
- `registrarUsuario($usuario, $password)`: Verdadero o falso dependiendo si es correcta la inserción en la base de datos.

Debéis crear una página **registro.php** para que se registren usuarios (ya hecha).

### Registro de usuarios

Las contraseñas no coinciden

El usuario cic ya está registrado  
SQLSTATE[23000]: Integrity constraint violation: 1062 Duplicate entry 'cic' for key 'usuarios.PRIMARY'

© I.E.S. Miguel Herrero

La clave en base de datos tiene que estar encriptada.

En otra debéis mostrar un listado de los productos (**productos.php**) y para cada uno de ellos debe existir un botón “**Añadir a la cesta**”:

Ejemplo:

## Listado de productos

Cesta de la compra



Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos  
Bases de datos  
**4.99 €**

Descripción del producto Reseña del editor  
¡Oferta de lanzamiento de la edición impresa! Sólo 4,99 € por tiempo limitado.  
El curso de SQL definitiv...

Añadir a la cesta



Aprende Git: ... y, de camino, GitHub  
Programación y desarrollo de software  
**4.90 €**

git es un sistema de control de versiones distribuido, que dicho así suena geek y aburrido, pero que en la práctica es una forma de trabajar en equipo...

Añadir a la cesta



Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad  
Seguridad informática  
**29.93 €**

Aprende la profesión de pentester, y a dedicarte al hacking ético. Kali es una distribución de Linux que contiene centenares de herramientas para h...

Añadir a la cesta



Java para novatos  
Programación y desarrollo de software  
**15.26 €**

Todo lo que necesitas saber para empezar a programar en Java aplicando el paradigma de orientación a objetos desde el primer momento. ¿Te han dicho...

Añadir a la cesta



No me hagas pensar  
Medios digitales y diseño gráfico  
**18.95 €**

Cientos de miles de diseñadores y desarrolladores web se han basado en la guía del gurú de usabilidad Steve Krug para ayudarles a entender los princip...

Añadir a la cesta



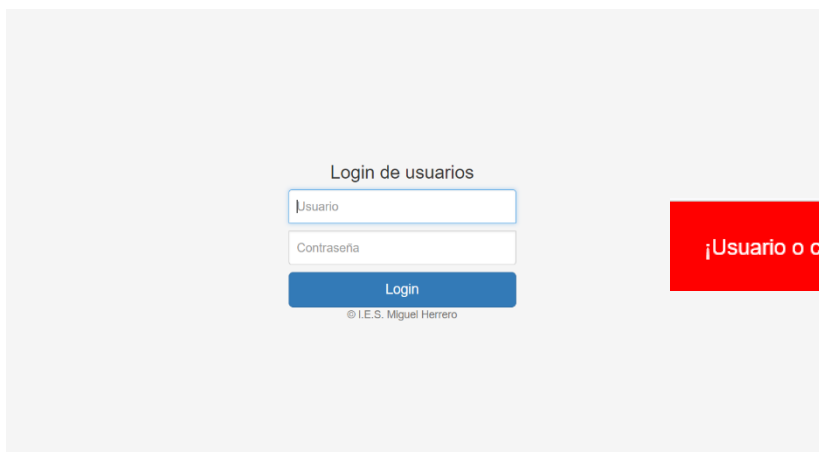
Planificación y Administración de Redes  
Redes y administración de sistemas  
**37.90 €**

La presente obra está dirigida a los estudiantes del Ciclo Formativo de Grado Superior de Administración de Sistemas Informáticos en Red, en concreto...

Añadir a la cesta

### Una página de login (**login.php**)

- Vas a crear una página **login.php** con un formulario con dos campos, uno de tipo text para el usuario, y otro de tipo password para la contraseña.
- Al pulsar el botón login, el formulario se enviará a esta misma página, donde se compararán las credenciales proporcionadas por el usuario con las almacenadas en la base de datos.
- Cuando un usuario proporciona unas credenciales de inicio de sesión correctas, se le redirige de forma automática a la página del listado de productos (**productos.php**)
- Esta página utiliza sesiones. Si los datos son correctos, se iniciará una nueva sesión y se almacenará en ella el nombre del usuario que se acaba de conectar.
- Si los datos son incorrectos sacamos un mensaje de aviso al usuario



### Modificaciones a la página de **Productos.php**

Tanto en esta página como en todas las demás, es necesario comprobar la variable de sesión `$_SESSION['usuario']` para verificar que el usuario se ha autenticado correctamente. Si el usuario no se ha autenticado, se muestra un mensaje de error junto con un enlace a la página **login.php**.



Error - debe identificarse

También debemos mostrar el **nombre de usuario** en la cabecera de la página. Este dato lo sacaremos de la sesión.

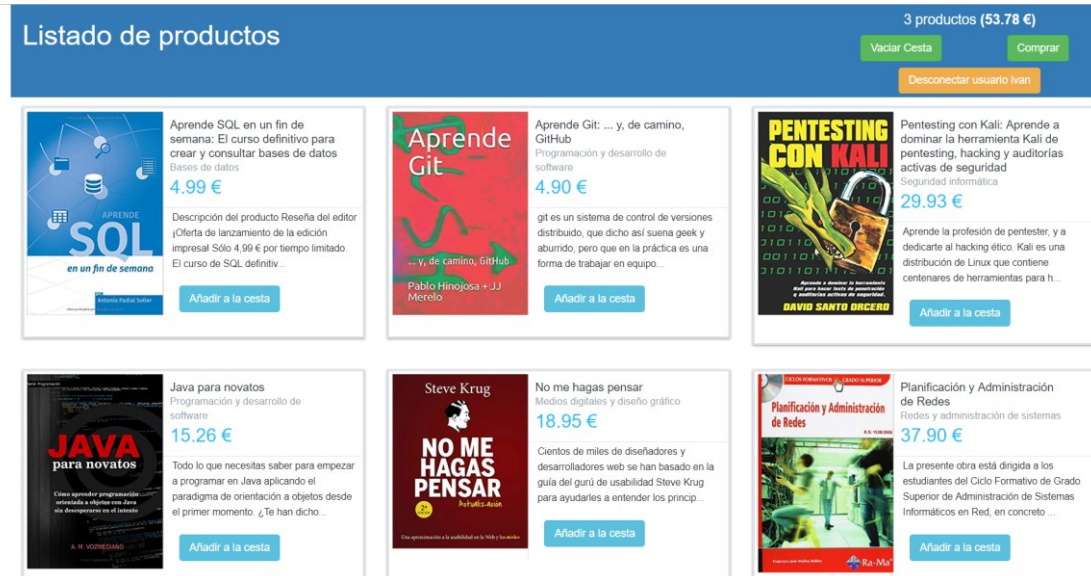
El botón "**Añadir**" envía a esta misma página los datos código, nombre y precio del producto.

Cuando se abre la página, se comprueba si se ha enviado este formulario, y si fuera así se añade un elemento al array asociativo `$_SESSION['cesta']` con los datos del nuevo producto. El array `$_SESSION['cesta']` es la variable de sesión en la que guardaremos los datos de todos los productos que va a comprar el usuario.

En esta página se mostrará el número de productos añadidos a la cesta y el importe total de ésta.

También habrá dos formularios: Uno para vaciar la cesta (botón "Vaciar Cesta"), dirigido a esta misma página, y otro para realizar la compra (botón "Comprar"), que dirige a la página **cesta.php**

Contendrá un botón para desconectar al usuario actual. Llama a la página **logoff.php**, que borrará la sesión actual.



**Listado de productos**

3 productos (53.78 €)

[Vaciar Cesta](#) [Comprar](#) [Desconectar usuario Ivan](#)

Producto	Descripción	Precio	Botón
Aprende SQL en un fin de semana: El curso definitivo para crear y consultar bases de datos	Reseña del editor ¡Oferta de lanzamiento de la edición impresa! Sólo 4,99 € por tiempo limitado. El curso de SQL definitiv...	4.99 €	<a href="#">Añadir a la cesta</a>
Aprende Git: ... y, de camino, GitHub	Programación y desarrollo de software git es un sistema de control de versiones distribuido, que dicho así suena geek y aburrido, pero que en la práctica es una forma de trabajar en equipo...	4.90 €	<a href="#">Añadir a la cesta</a>
PENTESTING CON KALI	Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad Seguridad informática	29.93 €	<a href="#">Añadir a la cesta</a>
JAVA para novatos	Programación y desarrollo de software Todo lo que necesitas saber para empezar a programar en Java aplicando el paradigma de orientación a objetos desde el primer momento. ¿Te han dicho...	15.26 €	<a href="#">Añadir a la cesta</a>
NO ME HAGAS PENSAR	Medios digitales y diseño gráfico Cientos de miles de diseñadores y desarrolladores web se han basado en la guía del gurú de usabilidad Steve Krug para ayudarlos a entender los princip...	18.95 €	<a href="#">Añadir a la cesta</a>
Planificación y Administración de Redes	Redes y administración de sistemas La presente obra está dirigida a los estudiantes del Ciclo Formativo de Grado Superior de Administración de Sistemas Informáticos en Red, en concreto ...	37.90 €	<a href="#">Añadir a la cesta</a>

## La página **Cesta.php**

Se muestra un resumen de los productos que ha seleccionado junto con el importe de los mismos.

Los datos que figuran en la página se obtienen todos de la información almacenada en la sesión del usuario. No es necesario establecer conexiones con la base de datos.

En esta página habrá dos formularios que simplemente redirigen a otras páginas: "Pagar", que redirige a la página **pagar.php**, que en nuestro caso lo único que debe hacer es eliminar la cesta del usuario. Y el que contiene el botón de desconexión, que es similar al que figuraba en la página **productos.php**, y dirige a la página **logoff.php**, que cierra la sesión del usuario



**Cesta de la compra**

3 productos (53.78 €)

Vaciar Cesta

Pagar

Desconectar usuario Ivan



Aprende Git: ... y, de camino, GitHub  
**4.90 €**



Pentesting con Kali: Aprende a dominar la herramienta Kali de pentesting, hacking y auditorías activas de seguridad  
**29.93 €**



No me hagas pensar  
**18.95 €**

Seguir comprando

Pagar

### Página de desconexión de la web **logoff.php**

Tanto desde la cesta como desde la página del listado de productos, se le ofrece al usuario la posibilidad de cerrar la sesión. Para ello se le dirige a la página **logoff.php**, que no muestra nada en pantalla y su única función es recuperar la sesión, eliminarla y redirigir a **login.php**

### Página de **Pagar.php**

Simplemente se recupera la información de la sesión y la elimina.

Se ha realizado su compra por importe de 53.78 €

Realizar otra compra