GUÍA PRÁCTICA GRUPAL

LM - SI

13/02/2020

Contenido

[1. Instalación de programas 1](#_Toc32482113)

[1.1 Xampp 1](#_Toc32482114)

[1.2 Oracle 11g Express Edition 6](#_Toc32482115)

[1.3 SQL Developer 6](#_Toc32482116)

[1.4 Comprobación 12](#_Toc32482117)

[2. Funcionamiento PL/SQL (para la semana del 17 al 23) 13](#_Toc32482118)

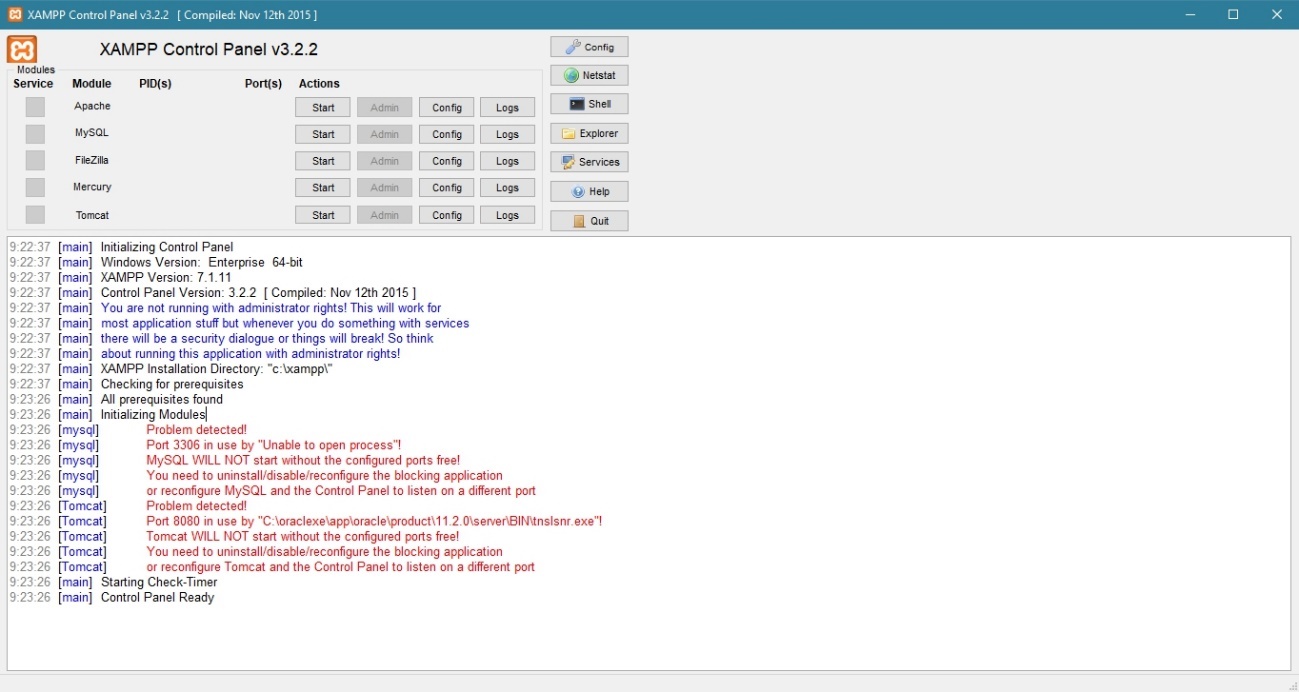
[3. Funcionamiento PHP (para la semana del 17 al 23) 14](#_Toc32482119)

# Instalación de programas

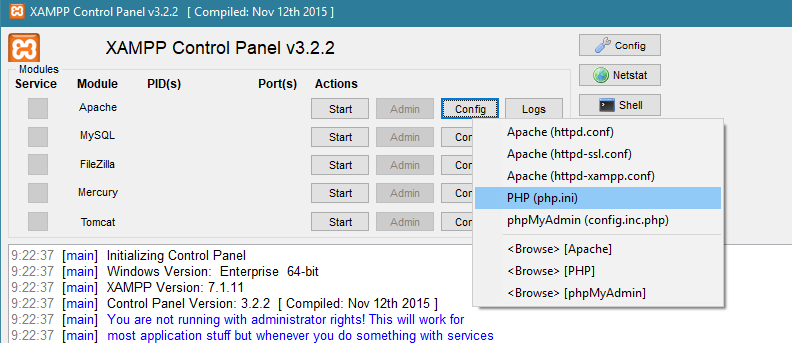
## Xampp

Instalamos la última versión que haya en su web.

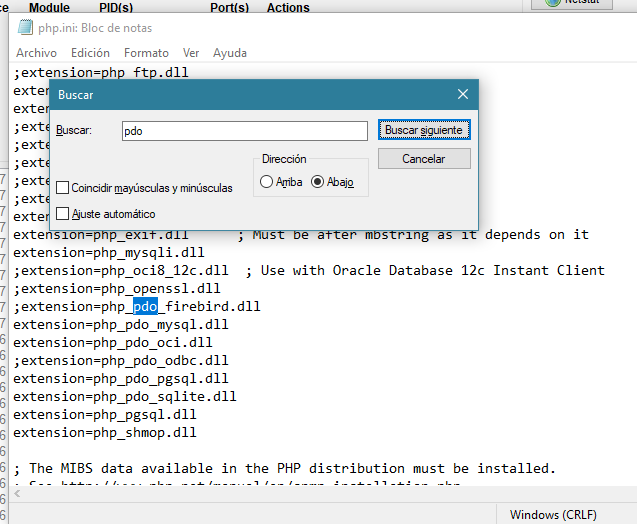
Una vez instalado debería aparecer esto:



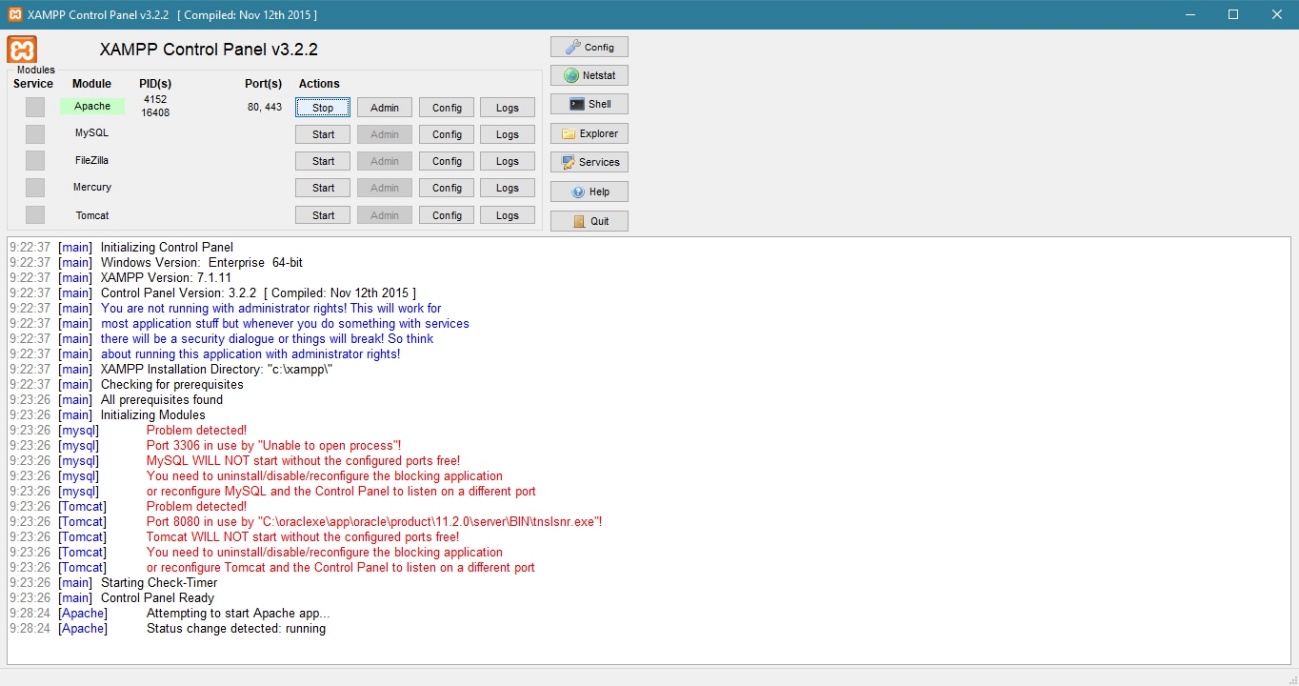
Ahora le damos a “config” y a “php.ini”



Buscamos “pdo” (ctrl+B) y ponemos los “;” de delante como está el siguiente

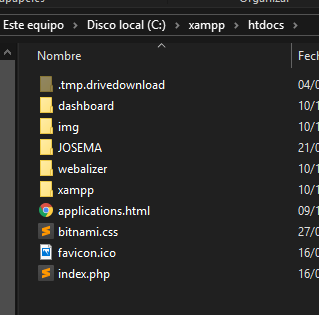


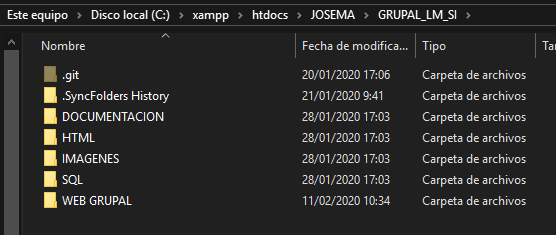
Por último, iniciamos apache. **Esto es necesario cada vez que queramos ver cosas de la base de datos en la web.** Debería quedar algo así:



Ahora vamos a añadirle las páginas de HTML, PHP, CSS o JavaScript a XAMPP.

Si nos vamos a “C:” deberíamos tener una carpeta “xampp”. Dentro de “xampp” debemos tener una carpeta “htdocs”. Pues dentro de “htdocs” copiamos las carpetas que tengamos en el repositorio de GitKraken. **Cada vez que modifiquemos algo aquí hay que acordarse de copiarlo a la carpeta de GitKraken y hacer el PUSH (Primero hacedme el PULL mamones, que si no podemos machacar lo de otra persona).**





Fijaos en la ruta.

**Consejo: Crearos un acceso directo desde el escritorio a esa carpeta antes de que queráis matar a alguien por tener que entrar cada vez hasta ahí.**

Ahora en Chrome abrimos la dirección “http://localhost/…” donde “…” es la dirección de la carpeta una vez entrados en “htdocs”, para que os hagáis una idea, mi localhost es “http://localhost/josema/GRUPAL\_LM\_SI/WEB%20GRUPAL/”.

Si preferís, una vez que entras en la primera carpeta puedes ir navegando por las carpetas que tengas ahí guardadas.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Imagen que contiene captura de pantalla

Descripción generada automáticamente

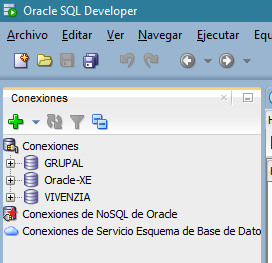
## Oracle 11g Express Edition

Descargamos del enlace de Mega que hay en Discord.

De Oracle no tengo pantallazos (y no me voy a poner a desinstalarlo por lo que pueda pasar XD). Lo que si hay que tener claro al instalar es que **HAY QUE GUARDAR LA CONTRASEÑA QUE PONGAMOS**. Si la olvidas hay que desinstalar (se necesita para SQL Developer).

## SQL Developer

En principio es un ejecutable, así que pasamos directamente a configurar.



Debería apareceros esta pantalla, pero con la sección de “Conexiones” vacía.

Le damos al y configuramos la siguiente pestaña que nos aparece (puede variar un poco según la versión. De nuevo deberíais tener vacía la columna de la izquierda. Si os aparece alguna opción que aquí no esté dejadla por defecto. **La contraseña debe ser la que se ha puesto al instalar Oracle 11g.**

Imagen que contiene captura de pantalla

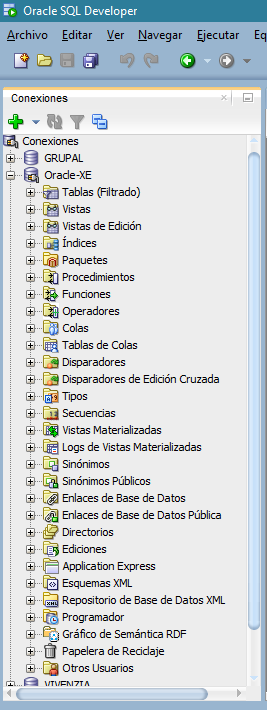
Descripción generada automáticamente

Antes de darle a nada mas le dais a probar.

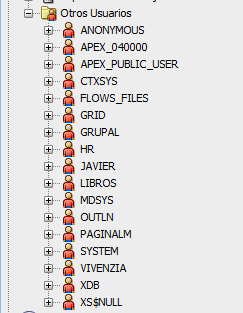
Imagen que contiene captura de pantalla

Descripción generada automáticamente

Si aparece “Estado: Correcto” perfecto, si no hay que mirar uno a uno (Que algo habréis hecho mal XD). Una vez correcto pulsamos en conectar. Y debería aparecernos la sección “Oracle-XE” dentro de conexiones (Sólo esa, paciencia).

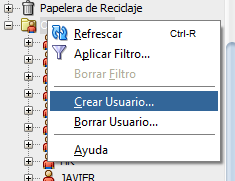


Ahora abrimos la sección de “Otros Usuarios” de dentro de “Oracle-XE”.

Esto habrá que hacerlo cada vez que quieras crear una base de datos nueva.

Si os aparecen mas o menos que a mi es por la versión. Si os fijáis tengo varias creadas por mi (“VIVENZIA”, “LIBROS”, “JAVIER”, “PAGINALM”, “GRUPAL”). Pues vamos a crear “GRUPAL”.

Hacemos click derecho sobre “Otros Usuarios” y a “Crear Usuario…”



En usuario y en los 2 campos de contraseña ponemos “GRUPAL”, para no dar por culo con el PHP (así todos tenemos la misma contraseña y no hay que estar cambiándola con cada PULL que hagamos).

Imagen que contiene captura de pantalla

Descripción generada automáticamente

**Antes de darle a “Aplicar”** os vais a “Roles Otorgados” y se marca “Connect” y “Resource”. Una vez marcados le damos a “Aplicar”.

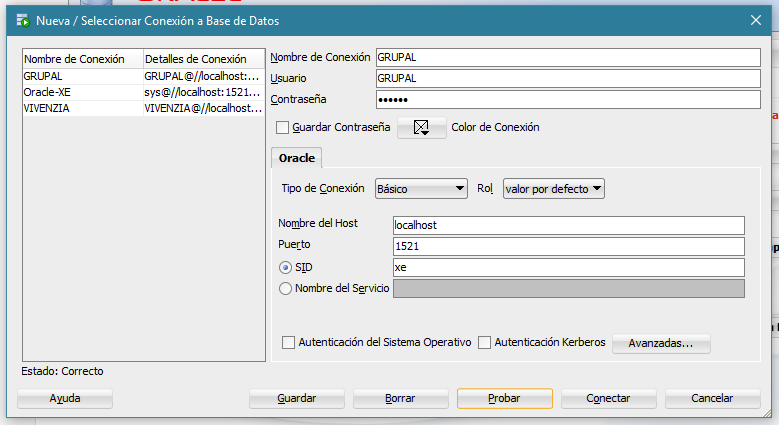
Imagen que contiene captura de pantalla

Descripción generada automáticamente

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Le damos a  de nuevo. Rellenamos la pestaña, ahora con GRUPAL, GRUPAL, GRUPAL y dejamos los demás valores por defecto. Le damos a probar, y si aparece “Estado: Correcto” le damos a conectar.

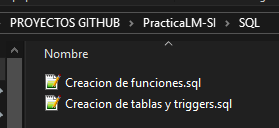


Ahora sí debería apareceros la sección conexiones con “Oracle-XE” y con “GRUPAL”.

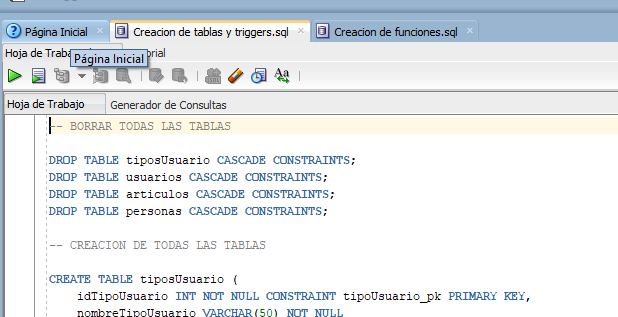
Imagen que contiene captura de pantalla

Descripción generada automáticamente

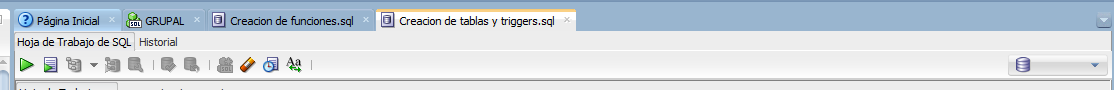
Ahora debemos irnos a la carpeta donde tengamos los archivos, a la carpeta SQL y arrastramos los archivos hasta SQL Developer.

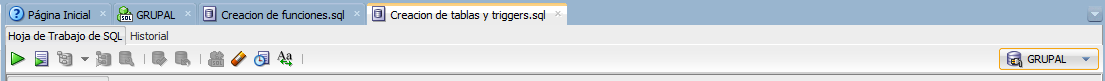


Debería quedar así.



Antes de ejecutar nada debemos asegurarnos de que lo hacemos sobre la base de datos GRUPAL. Fijaos en el cambio de la opción de la derecha de la imagen.





Para ejecutar miramos estos botones . El play verde ejecuta solo la línea que tengamos marcada, la hoja con el miniplay ejecuta toda la página **(Al reves que en workbench)**.

**Primero ejecutamos 2 veces el de Creación de tablas y triggers, y luego 2 veces el de creación de funciones. Esto solo se hace la 1ª vez. Si no borraremos lo que insertemos desde la web. Si obtenéis fallos la segunda vez que le deis fijaos que en los mensajes de error debería haber mensajes escritos en los triggers (excepciones).**

**Es muy importante que si insertáis datos desde la web guardéis en un .txt todo lo que vayáis a meter, todos los campos del formulario, incluidos posibles enlaces o fotos con su estructura html (las fotos se van a guardar como parte del varchar del contenido, así que las podéis coger de internet sin descargar).**

## Comprobación

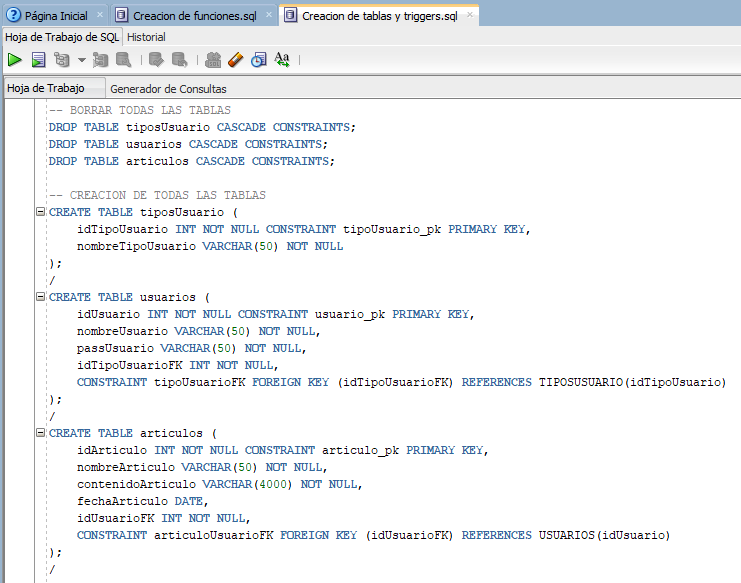
Si todo ha funcionado bien ahora debéis iros al localhost de XAMPP y entrar en “consultaArticulos.php" y que os aparezca esto. Si no sabes llegar vuelve al final de la instalación de XAMPP para ver cómo llegar.

Imagen que contiene captura de pantalla

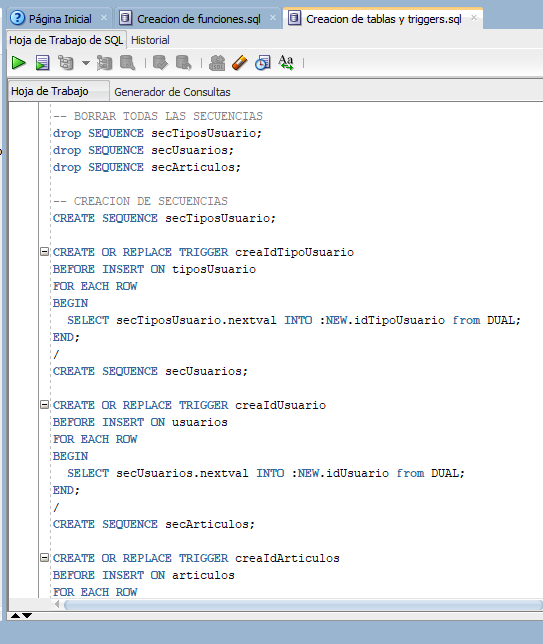
Descripción generada automáticamente

# Funcionamiento PL/SQL

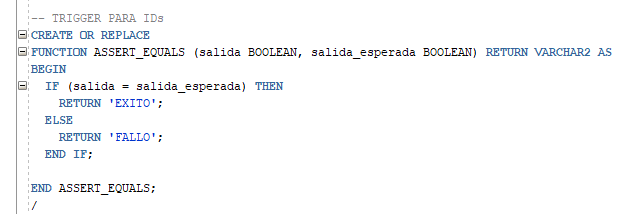
## Creación de tablas y triggers.sql



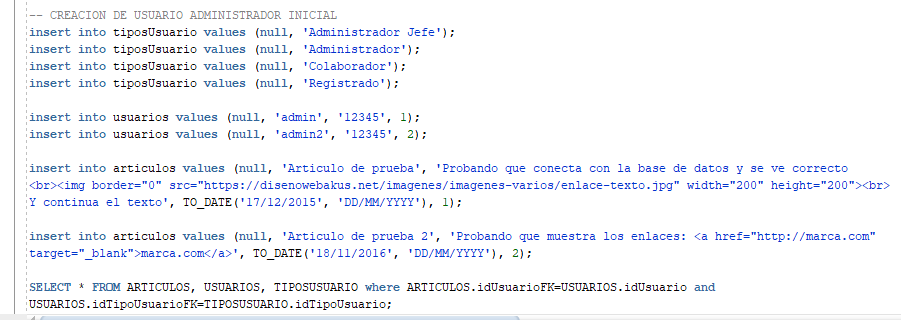
Si os fijáis se parece mucho a lo que hacemos en SQL Server, la diferencia es que a las primary key y las foreing key se les da un nombre.



Las secuencias son necesarias para que los “id…” sean autonuméricos. Así se evita que alguien con malas intenciones te machaque un “id” que ya estuviese creado.

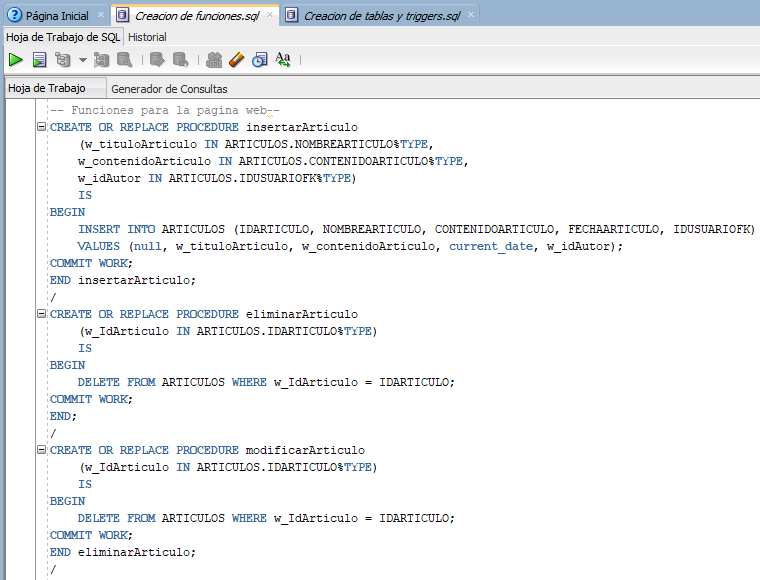


Este otro trozo de código es para lo mismo que las secuencias. Evitamos duplicidad de id.



De los insert solo fijaros en que los de artículos incluyen todo el tema de los enlaces para cuando se muestren en el localhost.

## Creación de funciones.sql



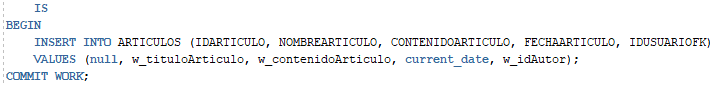
Las funciones y procedimientos se parecen a los métodos de JAVA. Se declara un nombre del procedimiento.



Se declaran las variables que vamos a insertar al llamar al procedimiento.



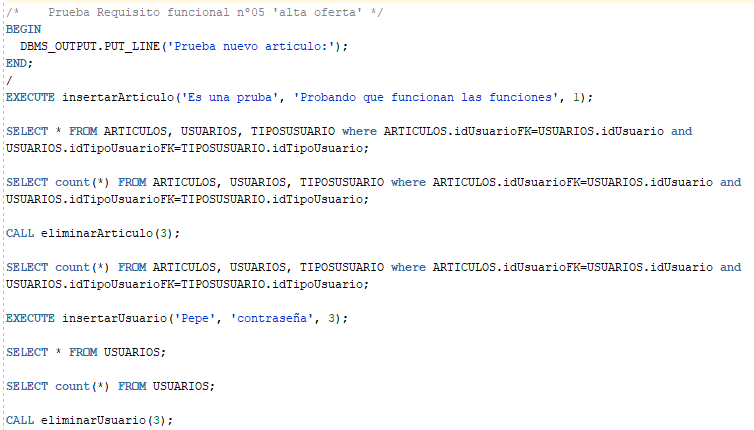
Y luego se escribe que va a hacer en concreto.



Y se cierra.



Cada uno según lo que haga es diferente, pero manteniendo esta estructura.



Las 3 primeras líneas de “Begin…” es lo que en JAVA sería el “sysout”.

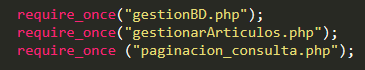
EXECUTE y CALL son llamadas a procedimientos y funciones. Mejor usar CALL siempre que podáis, en principio deberían funcionar igual de bien en SQL pero en PHP puede dar problemas EXECUTE.

Este código son pruebas de los procedimientos creados.

# Funcionamiento PHP

PHP lo que va a hacer es mostrar por el HTML la información de SQL, también vamos a poder llamar a procedimientos (insertar, borrar, modificar). Va a estar “mezclado” con el HTML, así que atentos.

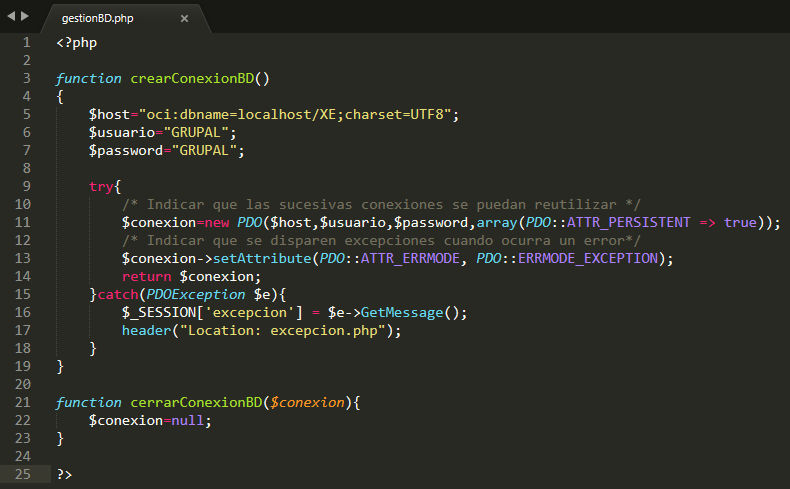
El orden no tiene que ser el que sigo por cojones. Puede haber variaciones. Si no veis claro cual es el siguiente PHP que tenéis que mirar buscad por el código.





Por poner 2 ejemplos. En “consultaArticulos.php” podemos tener que mirar cualquiera de estos 4 para buscar un fallo, aparte del propio código de “consultaArticulos.php”.

## gestionBD.php

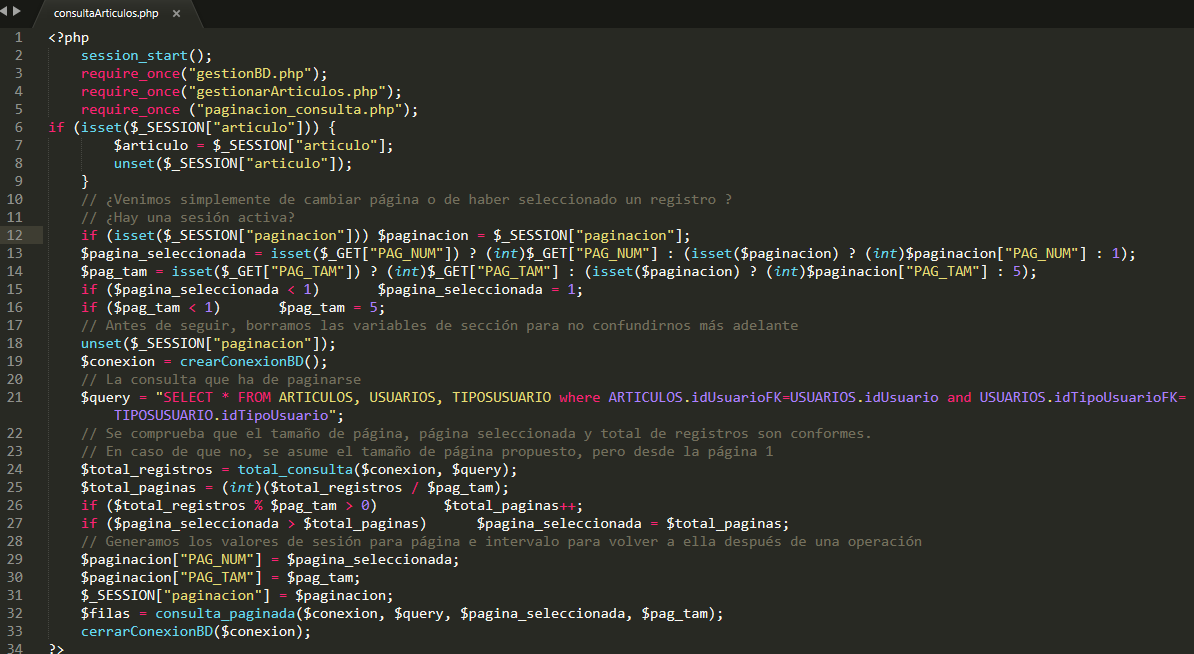


De aquí solo nos interesa la parte donde pone:



Ese usuario y esa contraseña deben ser los que pongamos en SQL (por eso os insistí en que todos pusiéramos “GRUPAL” “GRUPAL”). Si alguna vez escribís un código para otra base de datos deberéis cambiar esto.

## consultaArticulos.php



Primera parte del código.

Las primeras líneas son las que necesita la pagina para funcionar (llamadas a la base de datos para ver los artículos, mas adelante se detalla como mostrarlo).

A partir del primer “if…” es un código necesario para comprobar la sesión en la que estamos (información actual de la página tal como el usuario que estará logueado o los datos del formulario cuando hagamos un insert o un update).

A partir del segundo “if…” es mas especial. Su utilidad es poder mostrar “x” artículos a gusto. Esta parte es 100% copia pega, si os leéis los comentarios se explica su funcionamiento.

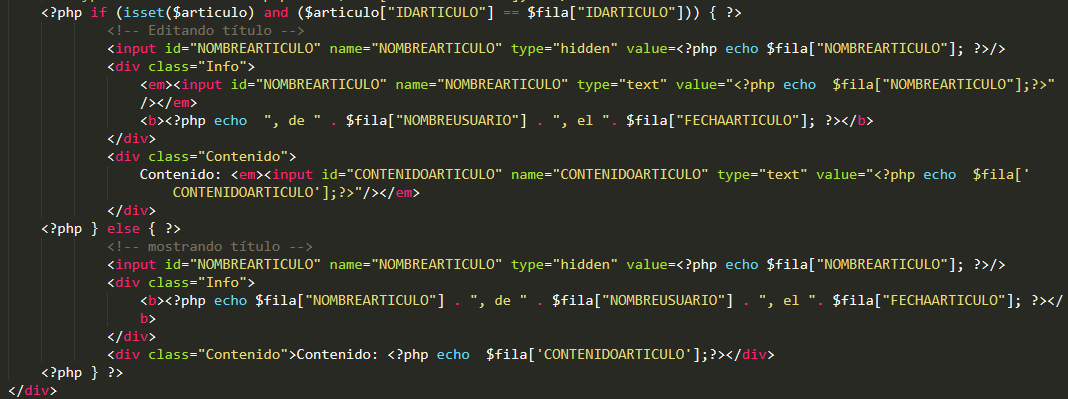


Aquí empieza el HTML que hemos visto en clase… más o menos.

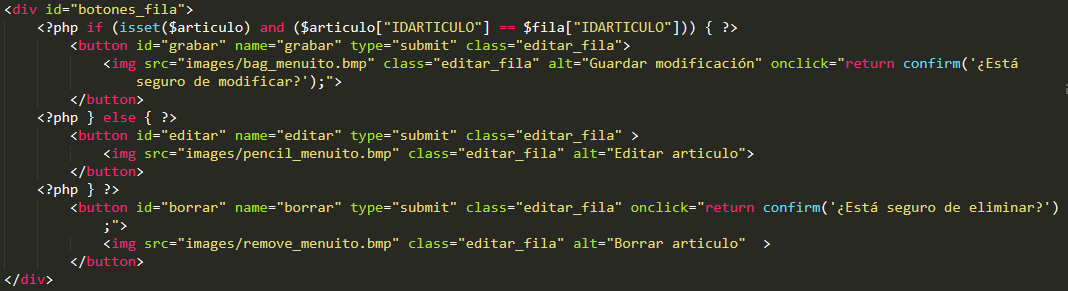
Lo primero que necesitamos para mostrar la info del PHP es lo que está en el main. Si os fijáis cada vez que se usa el PHP se empieza con “<?php” y se termina con “?>”.

Luego se crea la clase “articulo” y se establece un formulario. Lo del formulario es para poder modificar después. Si no vamos a modificar no haría falta.

Cada “input” es cada uno de los valores que vamos a trabajar después. Dentro de cada uno hay un trozo de PHP en la parte de “value”, enlazando cada dato con su nuevo “id”.



Aquí se muestra la información o se modifica. Se llama al objeto “$fila” que es el que contiene cada apartado de lo que vamos a mostrar.



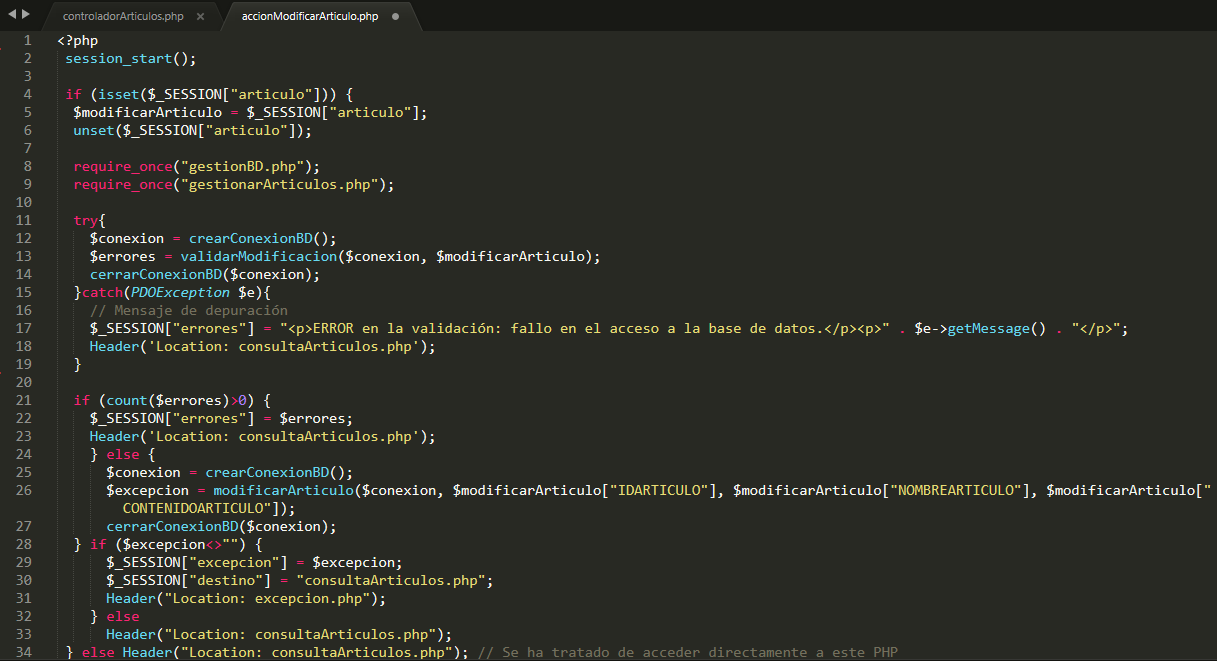
Los botones para las distintas opciones, cuando pulsemos nos mandarán a controladorArticulos.php como pusimos al principio del formulario. Las funciones “onclick=”return confirm(…)” son de JavaScript.

## controladorArticulos.php

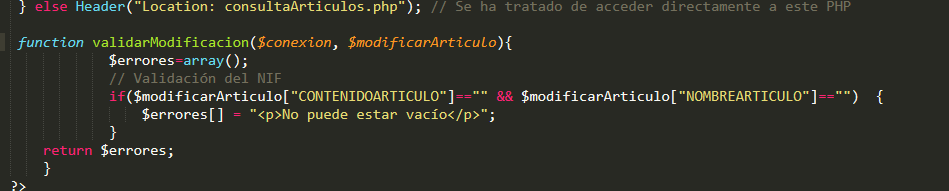
## 

Aquí no hay mucho que ver. Solo se identifica la información que se añade en $articulo y se manda a otro de los php o se vuelve a consultaArticulos.

## accionModificarArticulo.php



Aquí volvemos a necesitar conectar con la BBDD y probamos a llamar a función modificarArticulo en la línea que llamará a la BBDD en gestionarArticulos.php (más adelante)



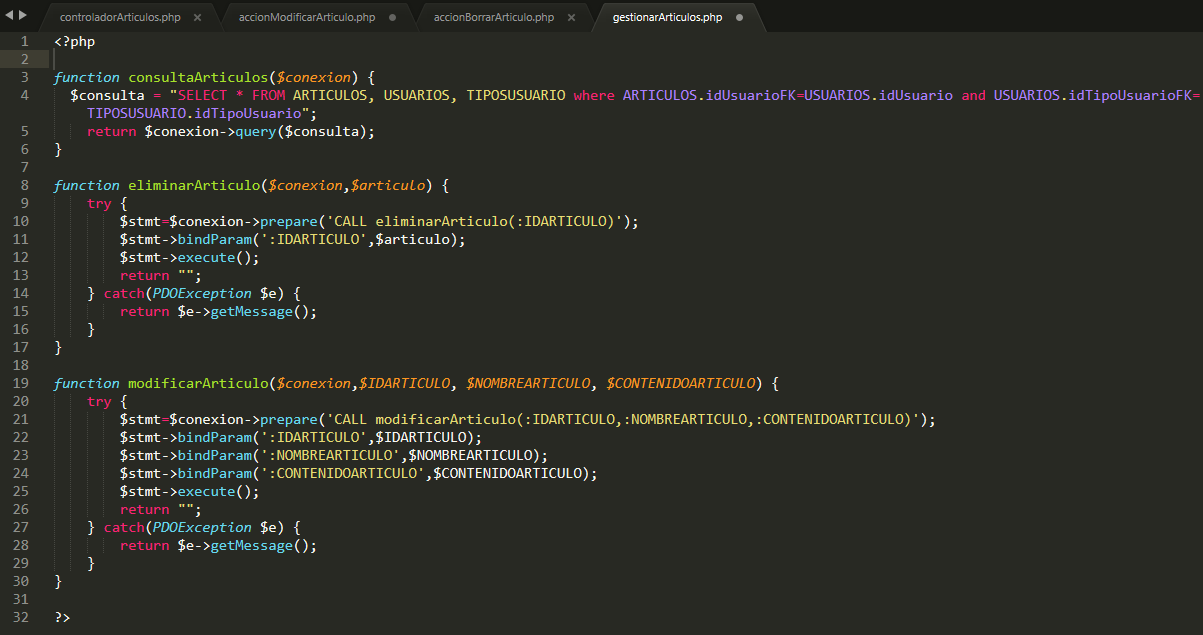
Aparte solo comprobamos que la modificación no venga vacía.

## accionBorrarArticulo.php



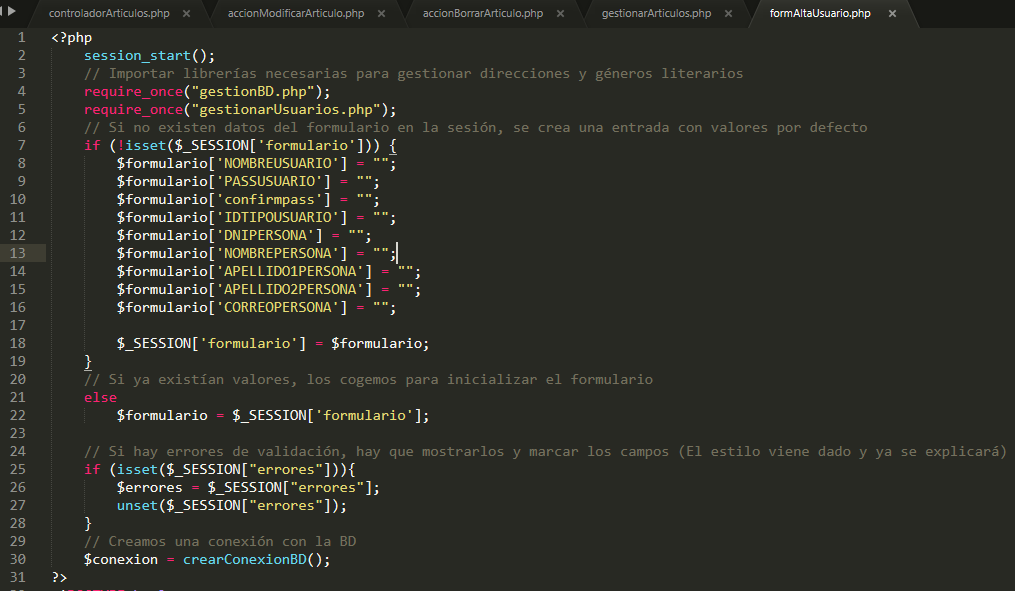
Lo mismo que en el anterior, pero llamando a eliminarArticulo.

## gestionarArticulos.php

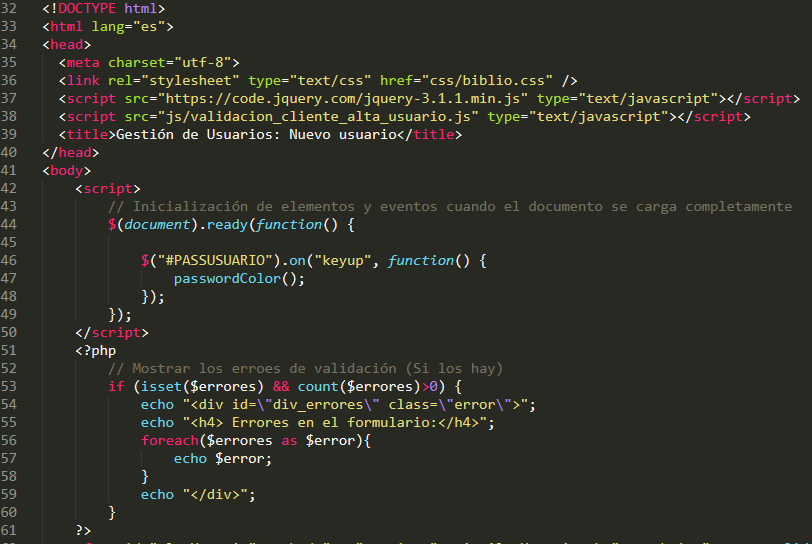


Aquí es donde están las funciones a las que hemos estado llamando. Y esas funciones a su vez llaman a la BBDD con “CALL nombreProcedimiento (variablesProcedimiento)”.

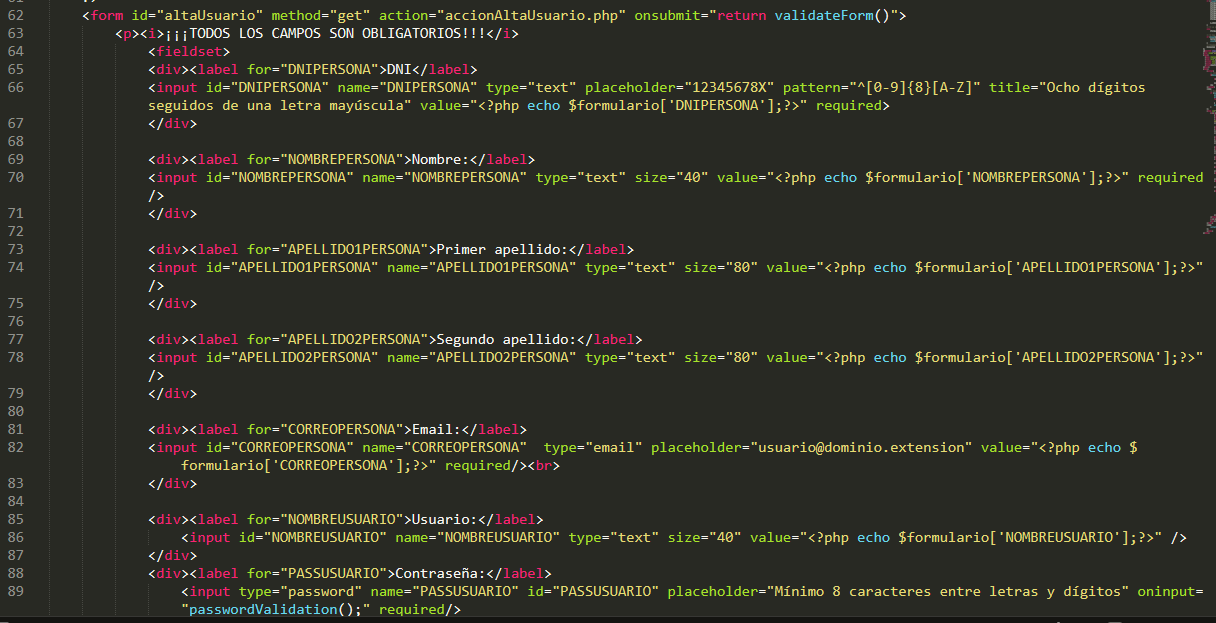
## formAltaUsuario



Parecido al inicio de consultaArticulos.php, pero en este caso declaramos la sesión “formulario” con todas las variables que vamos a necesitar. OJO que se llama a “gestionarUsuarios.php”, necesario para datos del formulario.



Inicio del HTML con las cosas de siempre y con un trozo de PHP que recopilará y mostrará los errores que cometamos al introducir la info en el formulario (como usuarios).

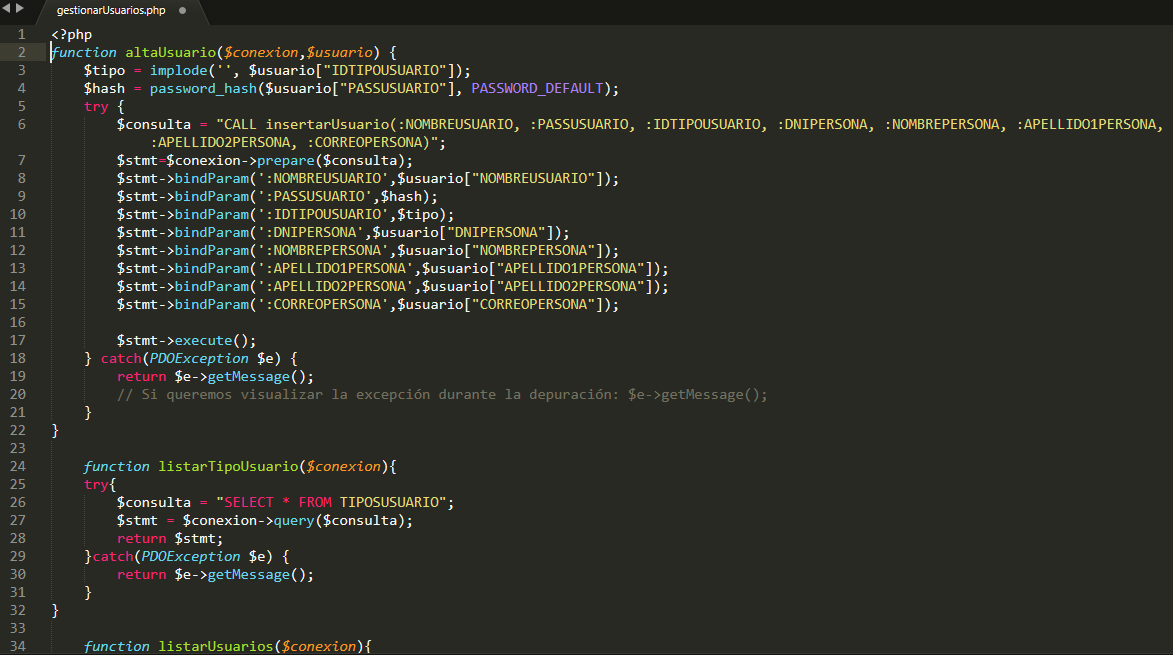


El formulario normal en HTML pero en cada value introducimos el correspondiente PHP.



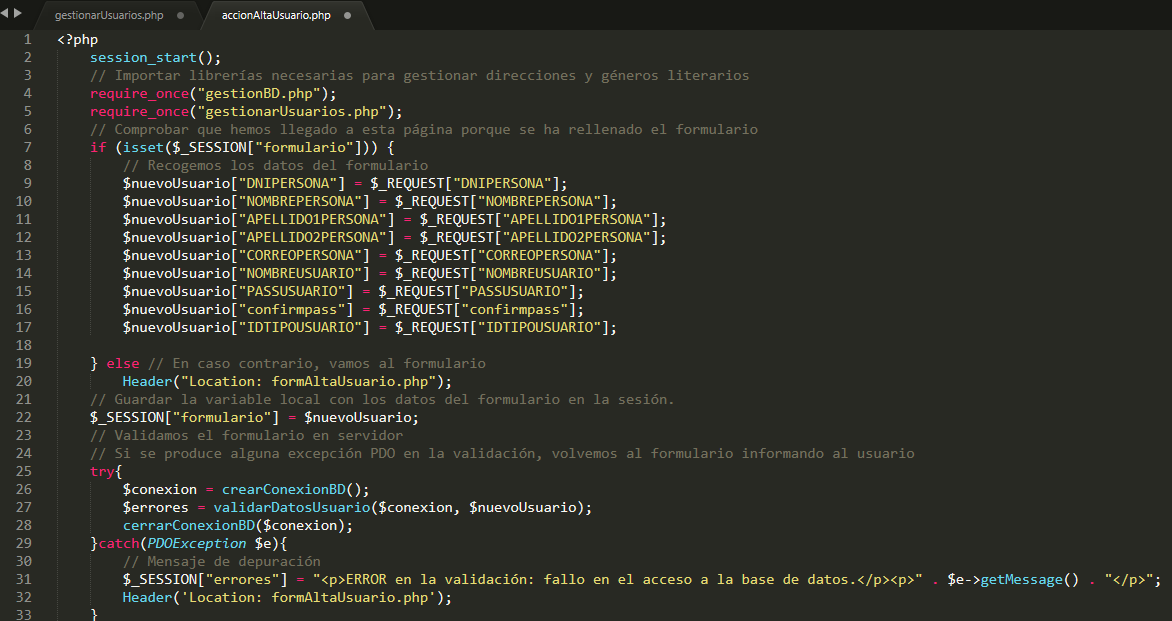
Aquí creamos un select que traeremos de la base de datos y que a la vez será un dato para el formulario. Para terminar, se cierra la conexión con la base de datos (abierta hasta aquí por lo del select).

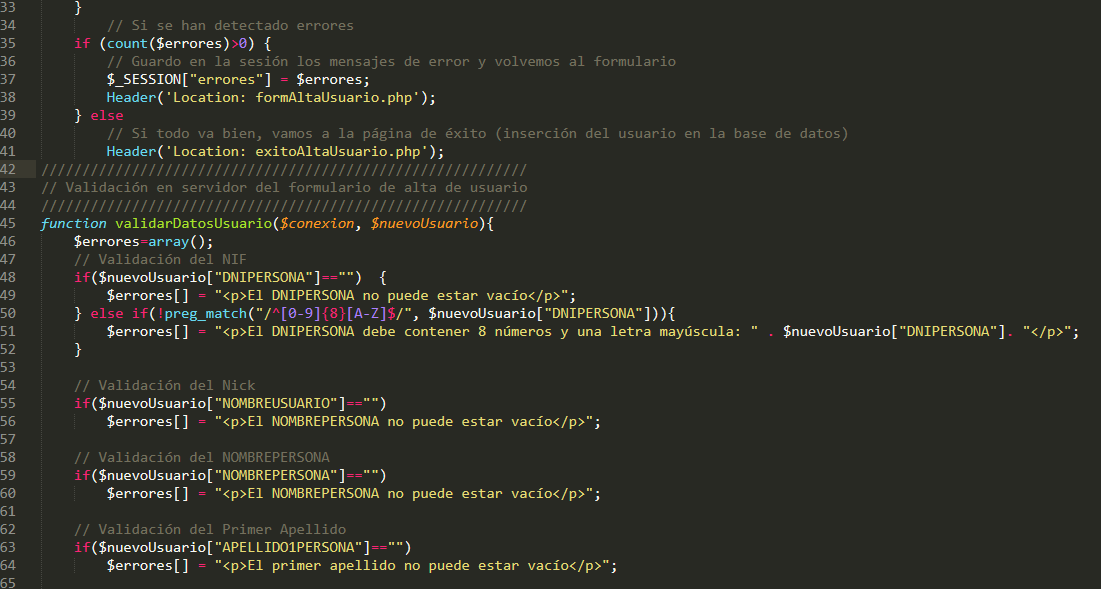
## gestionarUsuarios.php

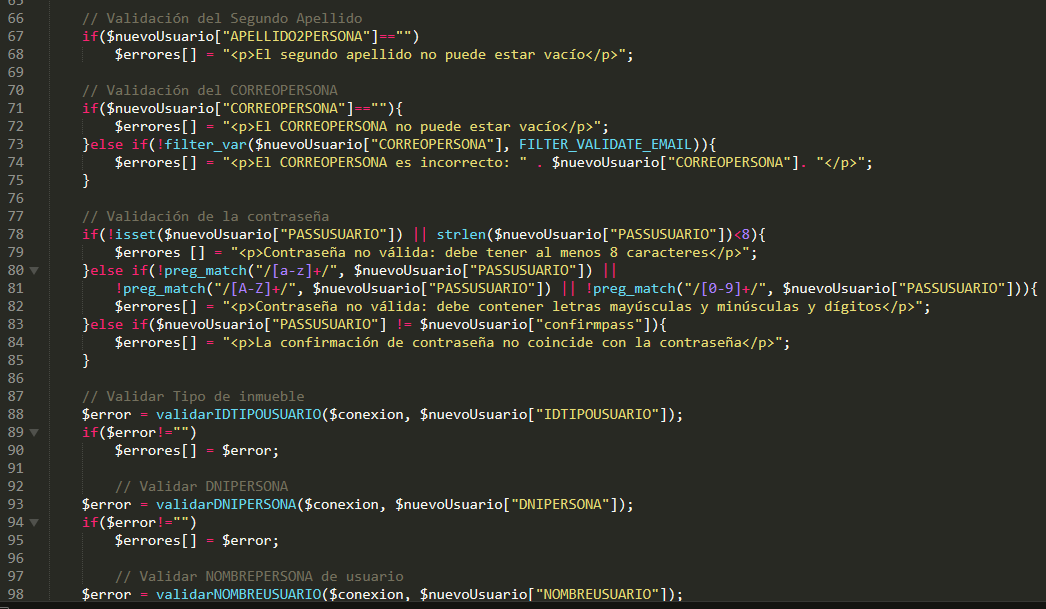


Este php esta lleno de las funciones necesarias para los Usuarios. Lo mismo que en gestionarArticulos.php. Hay CALL a procedimientos y SELECT que se dirigen a la BBDD.

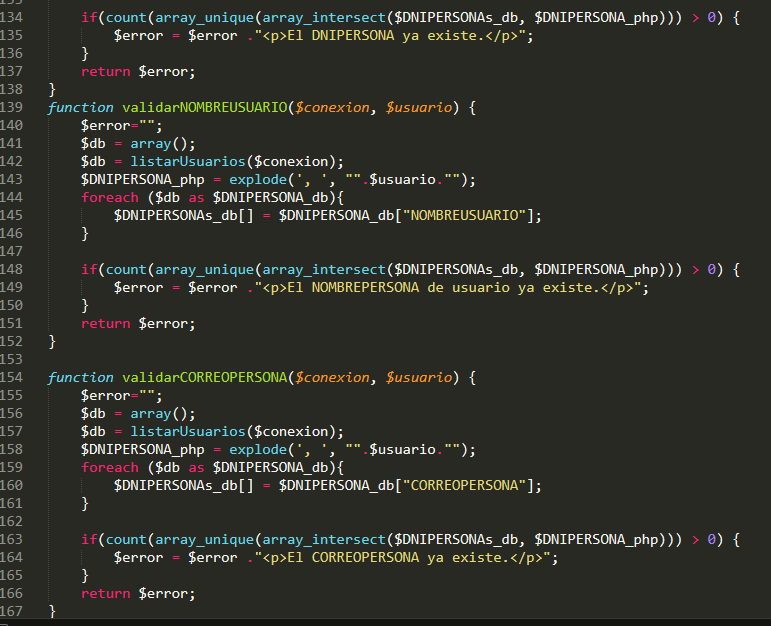
## accionAltaUsuario.php







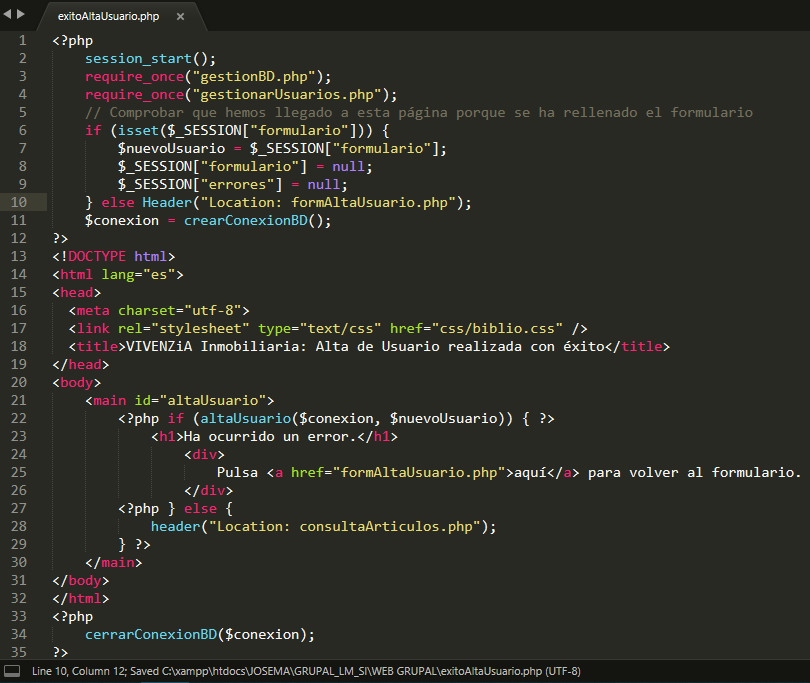




Este lo pongo entero porque una cosa tira de otra.

Es el mas lioso. Al final lo que se comprueba es que nada vaya vacío y que todo cumpla con los requisitos y no haya repeticiones de cosas que no pueda haberlas.

## exitoAltaUsuario.php



En este php lo que se hace es que si la validación anterior ha ido bien se manda a la función altaUsuario (que está en gestionarUsuarios.php) y si no ha ido bien pues nos da la opción de volver al formulario.

## excepcion.php y paginacionConsulta.php

Estas no se tocan, se copian y se pegan. Esto me lo dieron tal como está.

La idea principal es que en excepcion.php se mostrarían los errores en caso de error al conectar con la BBDD. Y paginacionConsulta es lo necesario para que el mostrar “x” artículos en consultaArticulos.php sea viable.