**Manual Técnico**

**Índice**

**Introducción**

El objetivo de este Manual Técnico es explicar los recursos que utilizamos para la creación del sistema los cuales son:

* HTML.
* PHP.
* MySQL.
* Apache HTTP Server.
* Sublime Text 3.
* Bootstrap.

Asumimos que en la máquina que se ejecutará el proyecto, tiene instalado algún navegador web, HTML (sublime, notepad o el blog de notas), PHP, mySQL y Apache HTTP Server.

**Definiciones Importantes**

¿Qué significa HTML?

HTML, sigla en inglés de HyperText Markup Language (lenguaje de marcas de hipertexto), hace referencia al [lenguaje de marcado](https://es.wikipedia.org/wiki/Lenguaje_de_marcado" \o "Lenguaje de marcado) para la elaboración de [páginas web](https://es.wikipedia.org/wiki/P%C3%A1gina_web" \o "Página web). Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros. Es un estándar a cargo del [World Wide Web Consortium](https://es.wikipedia.org/wiki/World_Wide_Web_Consortium" \o ") ([W3C](https://es.wikipedia.org/wiki/W3C" \o "W3C)) o Consorcio WWW, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. Se considera el lenguaje web más importante siendo su invención crucial en la aparición, desarrollo y expansión de la [World Wide Web](https://es.wikipedia.org/wiki/World_Wide_Web" \o "World Wide Web) (WWW). Es el estándar que se ha impuesto en la visualización de páginas web y es el que todos los navegadores actuales han adoptado.

¿Qué significa PHP?

PHP, acrónimo recursivo en inglés de PHP Hypertext Preprocessor (procesador de hipertexto), es un [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n" \o "Lenguaje de programación)[de propósito general](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_de_prop%C3%B3sito_general" \o ") de [código del lado del servidor](https://es.wikipedia.org/wiki/Script_del_lado_del_servidor" \o "Script del lado del servidor) originalmente diseñado para el [desarrollo web](https://es.wikipedia.org/wiki/Desarrollo_web" \o "Desarrollo web) de [contenido dinámico](https://es.wikipedia.org/wiki/Contenido_din%C3%A1mico" \o "Contenido dinámico). Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en un documento [HTML](https://es.wikipedia.org/wiki/HTML" \o "HTML) en lugar, de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el HTML resultante.

¿Qué significa MySQL?

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

Existen muchos tipos de bases de datos, desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL, como base de datos relacional, utiliza multiples tablas para almacenar y organizar la información. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interactuación con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

¿Qué significa Apache HTTP Server?

El servidor HTTP Apache es un [servidor web](https://es.wikipedia.org/wiki/Servidor_web" \o "Servidor web) [HTTP](https://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol" \o "Hypertext Transfer Protocol) de [código abierto](https://es.wikipedia.org/wiki/Software_de_c%C3%B3digo_abierto" \o "Software de código abierto), para plataformas [Unix](https://es.wikipedia.org/wiki/Unix" \o "Unix) ([BSD](https://es.wikipedia.org/wiki/BSD" \o "BSD), [GNU/Linux](https://es.wikipedia.org/wiki/GNU/Linux" \o "GNU/Linux), etc.), [Microsoft Windows](https://es.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows), [Macintosh](https://es.wikipedia.org/wiki/Macintosh" \o "Macintosh) y otras, que implementa el protocolo HTTP/1.1​ y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular [NCSA HTTPd](https://es.wikipedia.org/wiki/NCSA_HTTPd" \o "NCSA HTTPd) 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que alguien quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de Estados Unidos, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. En inglés, a patchy server (un servidor "parcheado") suena igual que Apache Server.

**Requerimientos del Apache Web Server:**

* **Para Unix:**

1. Mínimo de 64 MB. de memoria RAM.
2. Mínimo de 50 MB. de disco duro para la instalación del Apache solamente (las paginas Web que Ud. vaya a cargar posteriormente no están calculadas en este mínimo de memoria).
3. Tener instalados los últimos drivers para TCP/IP en sus sistema.
4. Este seguro de tener instalado un compilador ANSI-C. El GNU Compiller (GCC) es el recomendado.

* **Para Windows:**

1. Mínimo de 64 MB. de memoria RAM.
2. Mínimo de 50 MB. de disco duro para la instalación del Apache solamente (las paginas Web que Ud. vaya a cargar posteriormente no están calculadas en este mínimo de memoria).

¿Qué significa Sublime Text 3?

**Sublime Text es un editor de código multiplataforma**, ligero y con pocas concesiones a las florituras. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

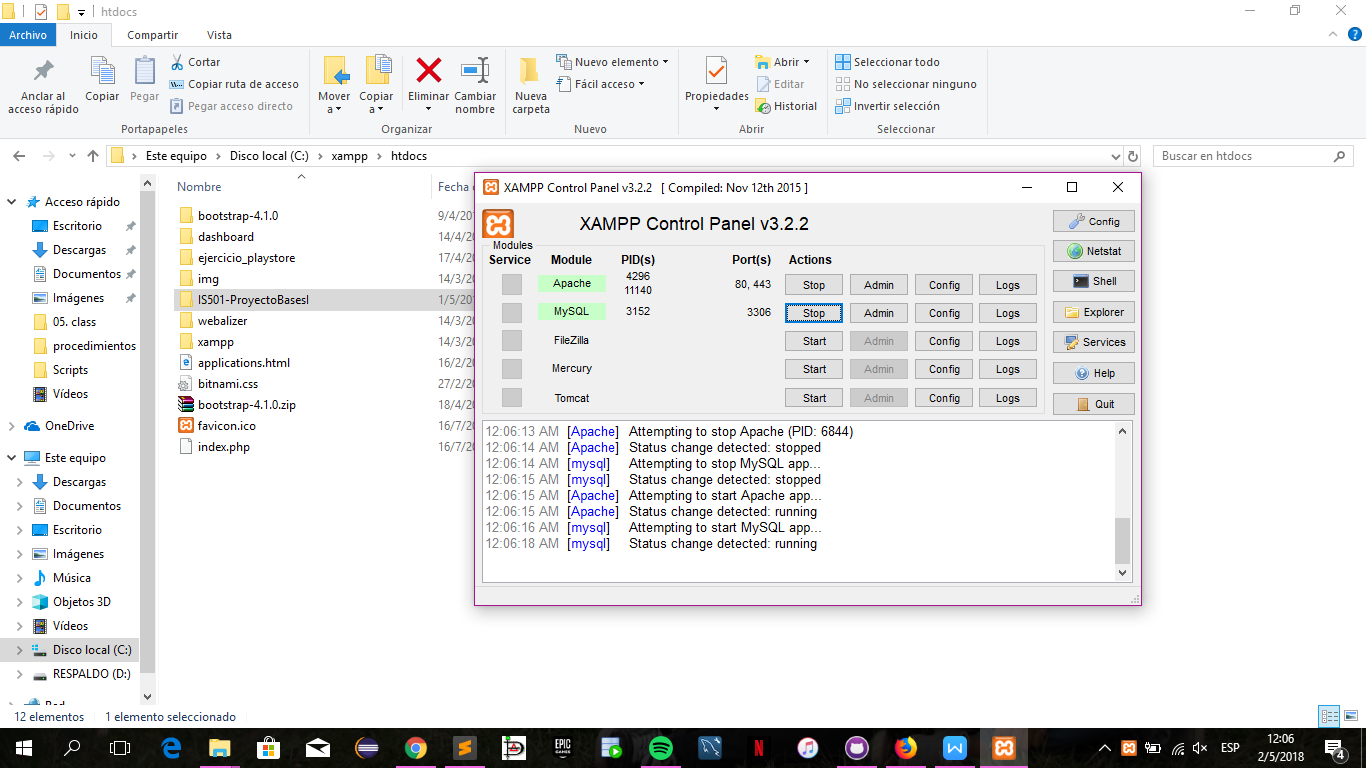
El sistema de resaltado de sintaxis de Sublime Text **soporta un gran número de lenguajes** (C, C++, C#, CSS, D, Erlang, HTML, Groovy, Haskell, HTML, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, Matlab, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML).

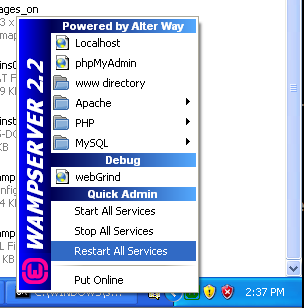
¿Qué significa Bootstrap?

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como “**responsive design**” o diseño adaptativo.

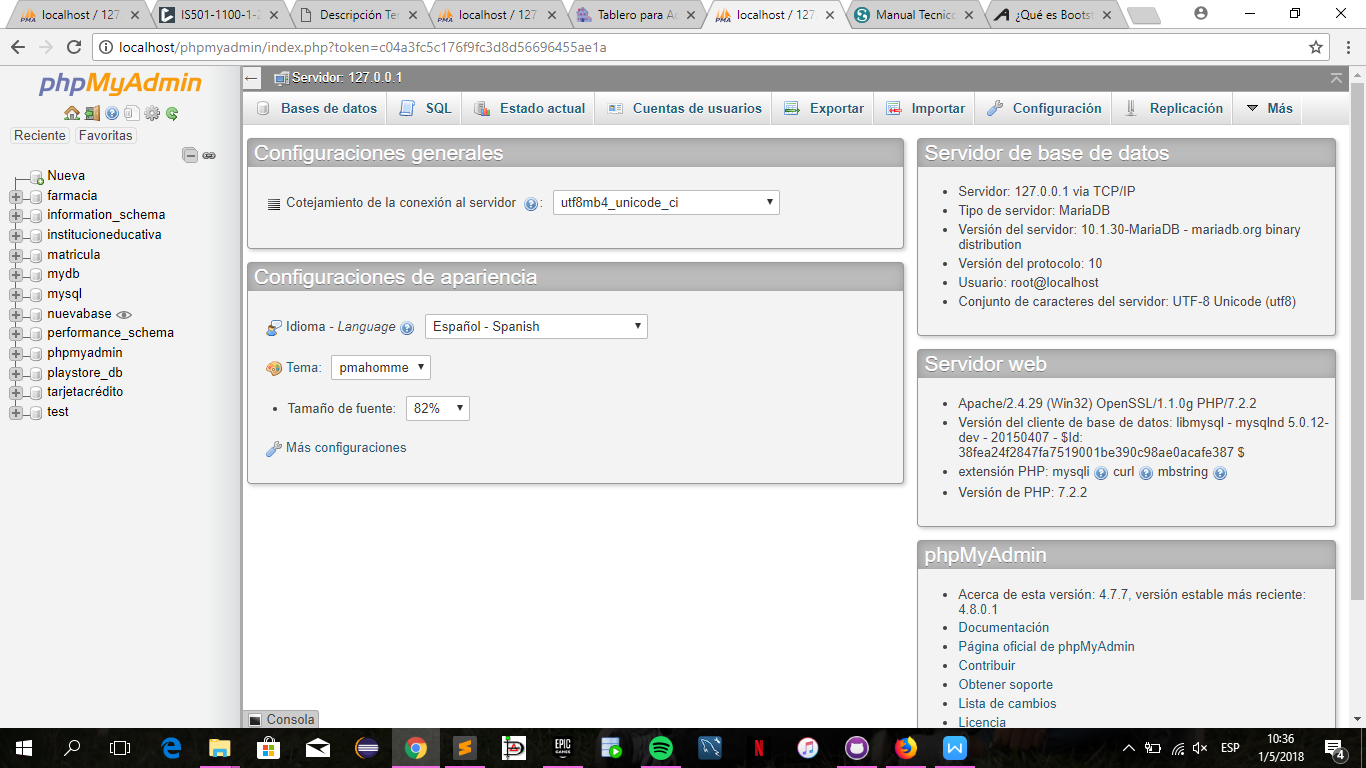
**Creación de Base de Datos MySQL:**

1. **Encender el xampp/wamp**

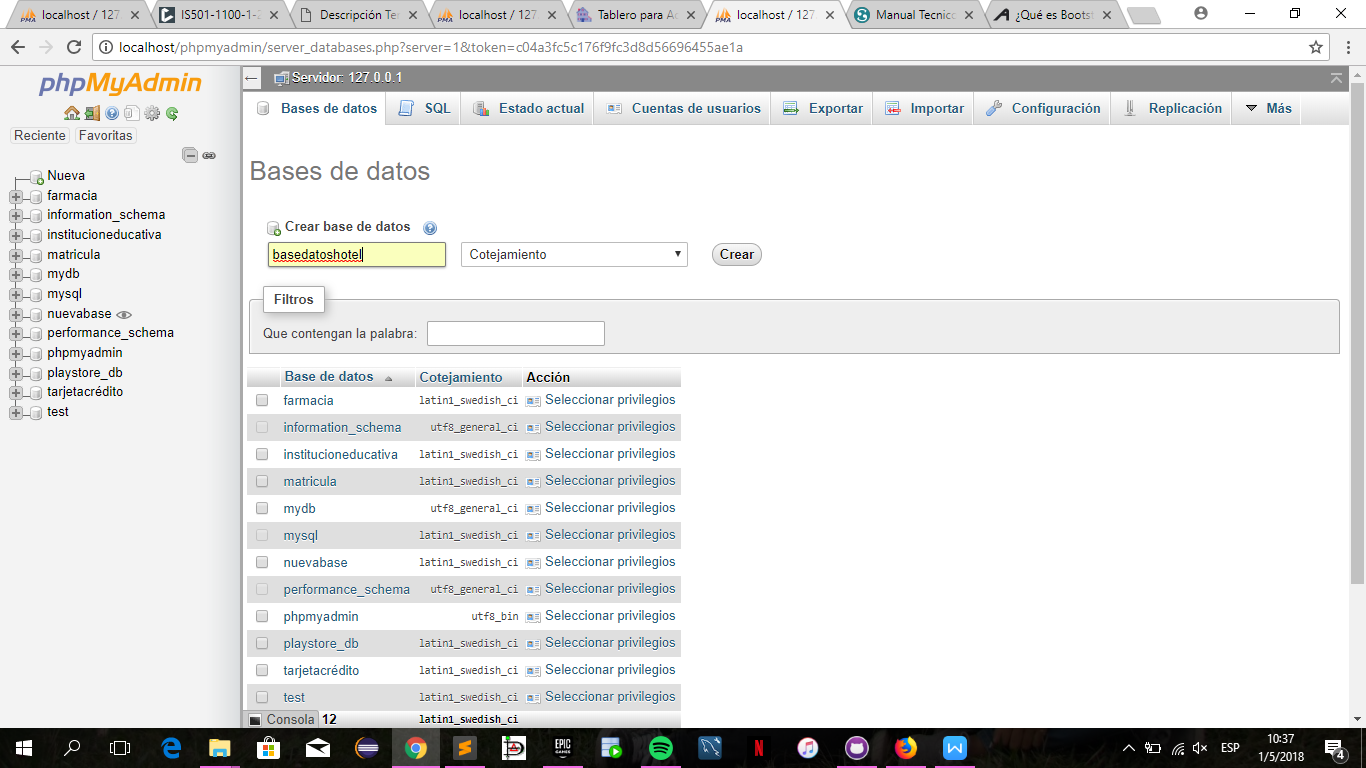




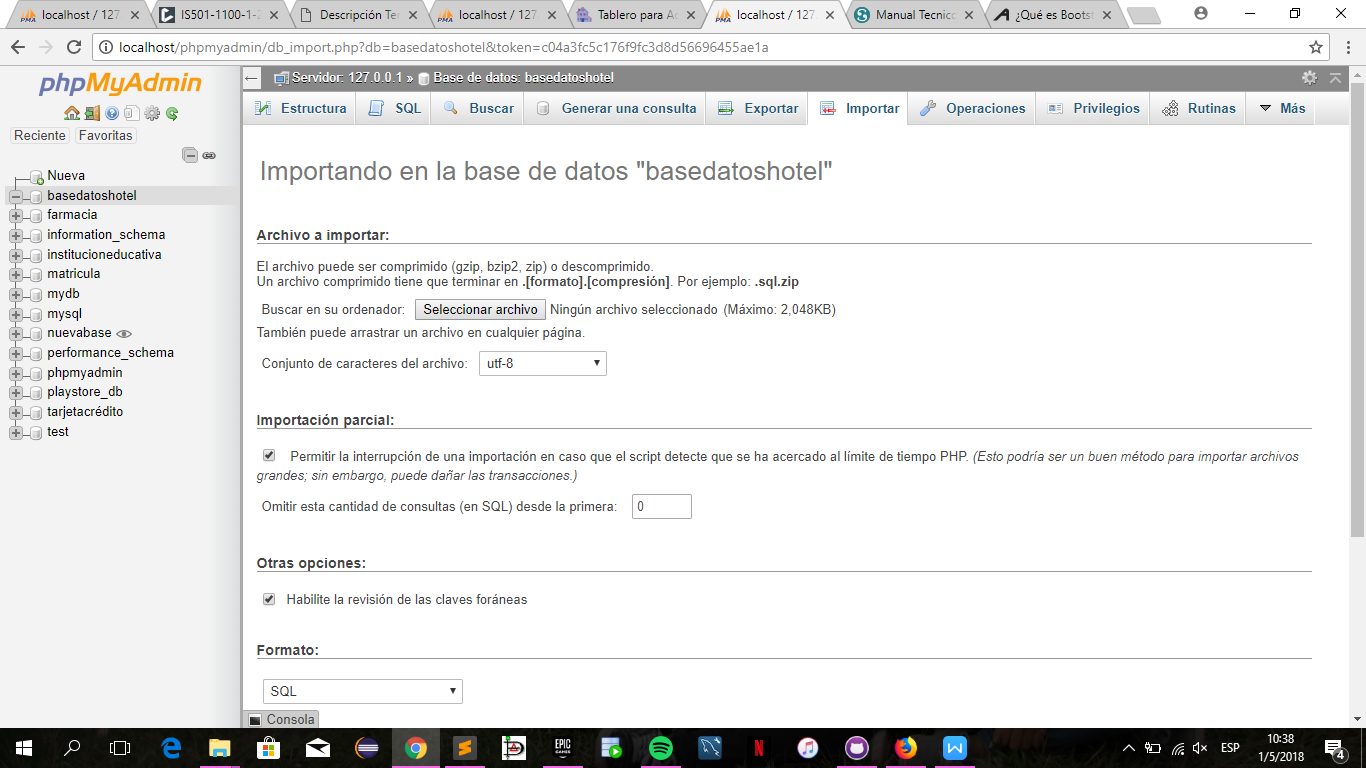
1. **Ingresar a localhost/phpmyadmin**



1. **Crear una nueva base de datos con el nombre : basedatoshotel**



1. **Dar click en “importar”**

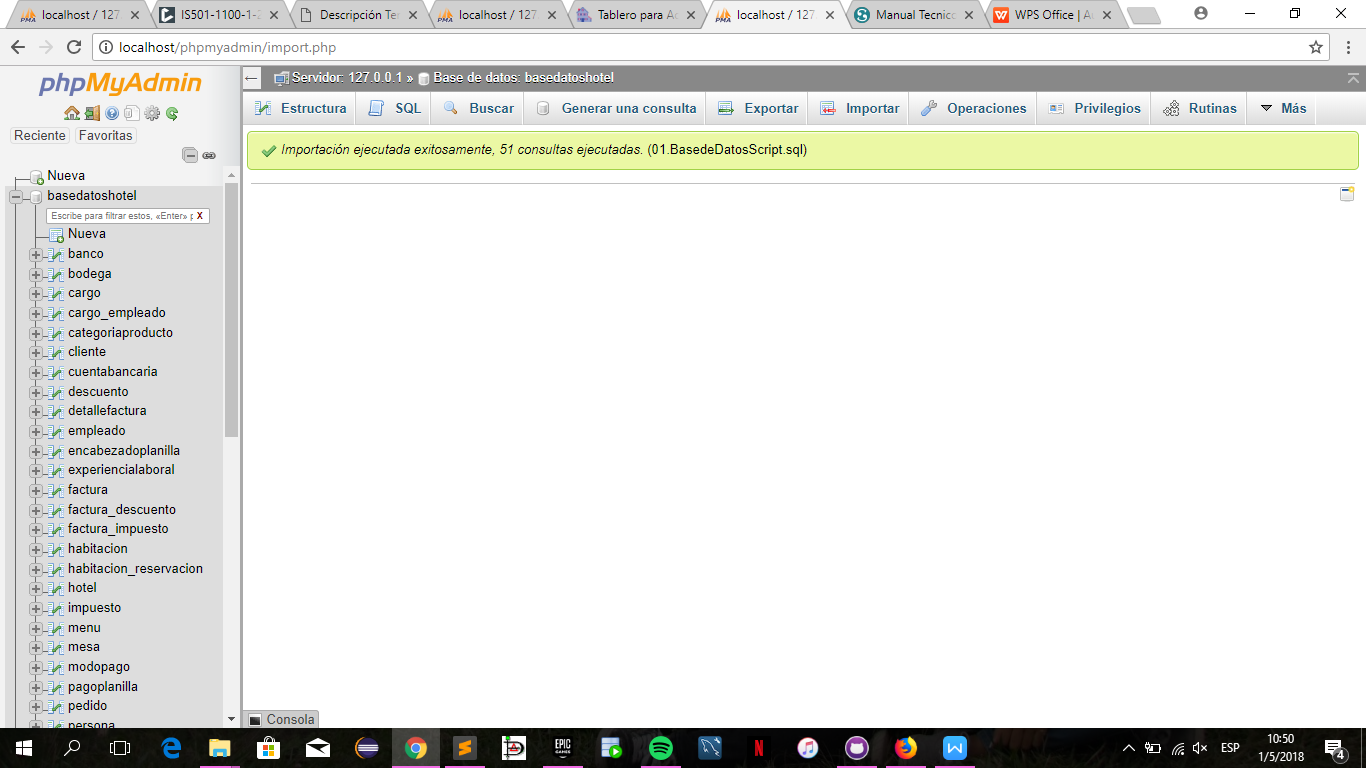


1. **En seleccionar archivo, copiar el siguiente archivo:**

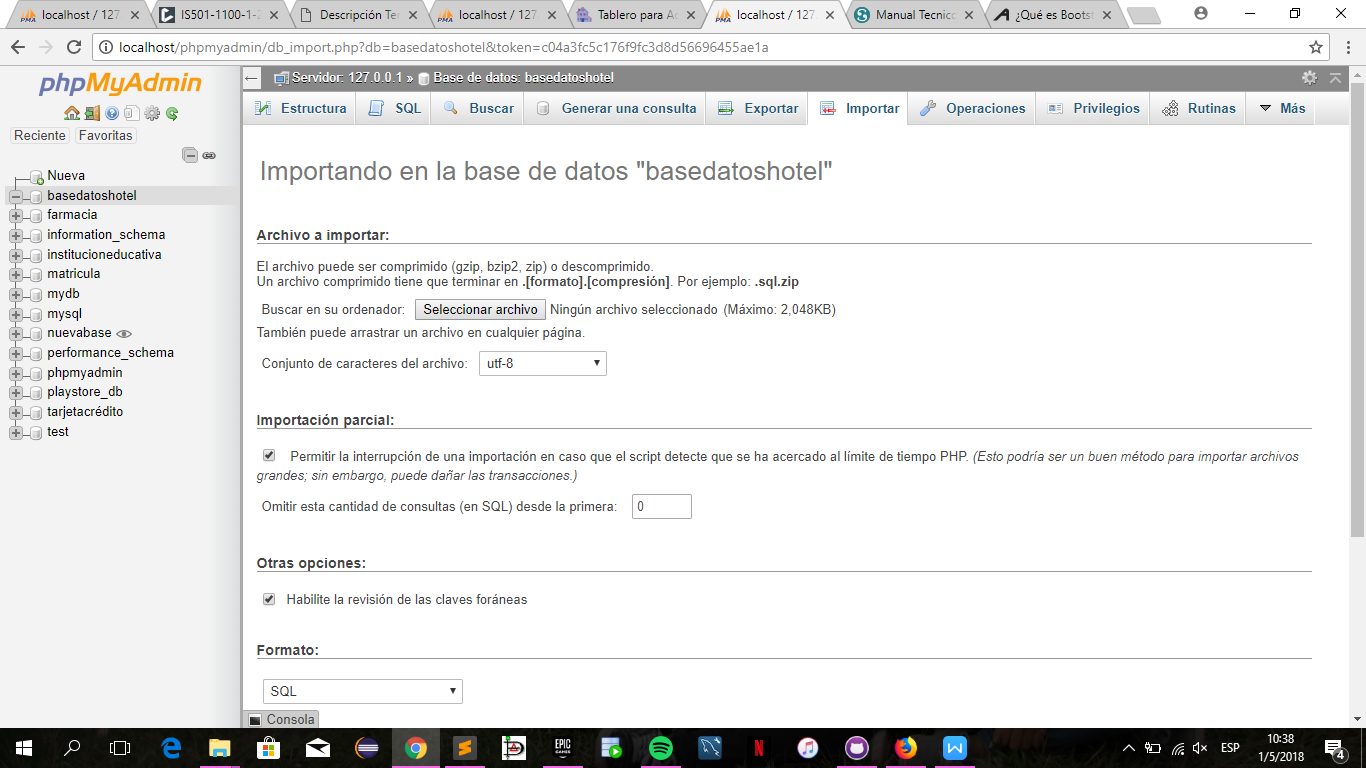
C:\xampp\htdocs\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\01.BasedeDatosScript.sql

Ó

C:\wamp\www\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\01.BasedeDatosScript.sql



1. **Volver a “importar”**

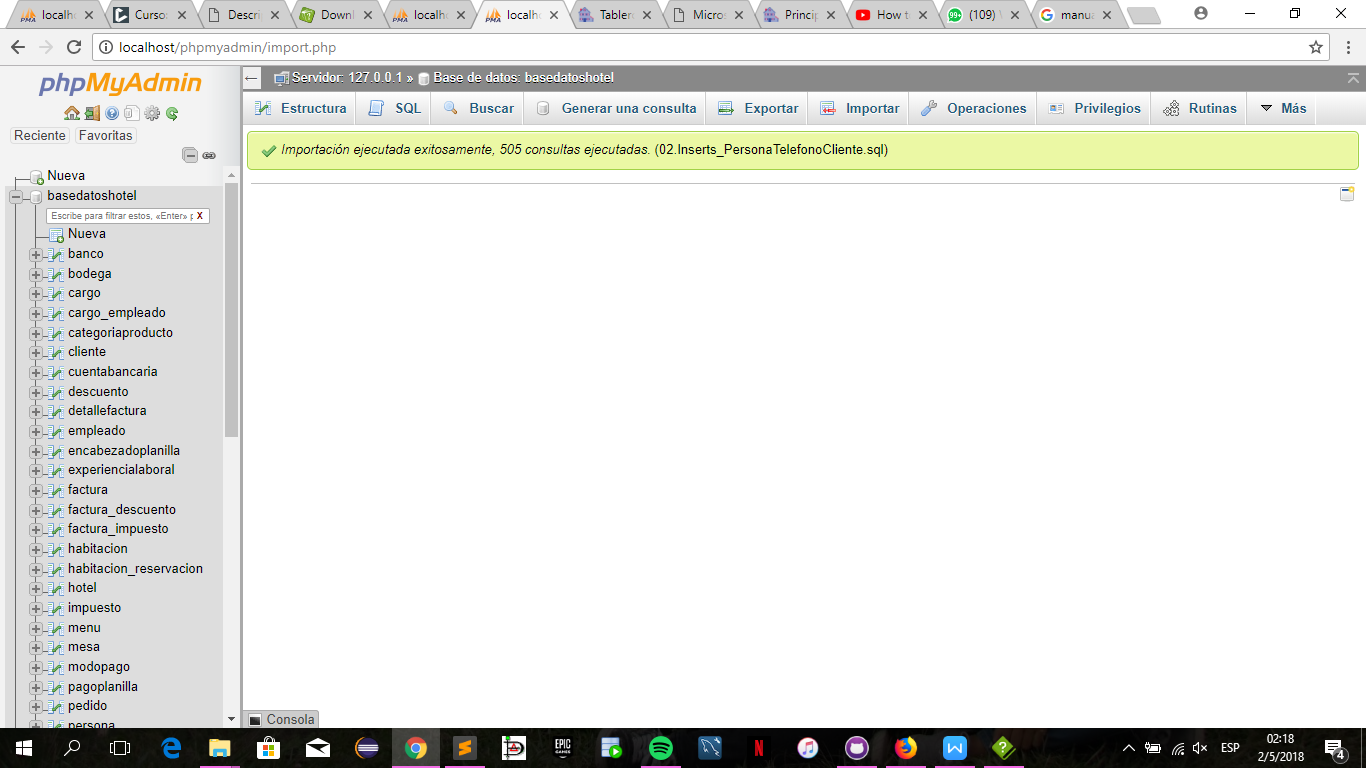


1. **En seleccionar archivo, copiar el siguiente archivo:**

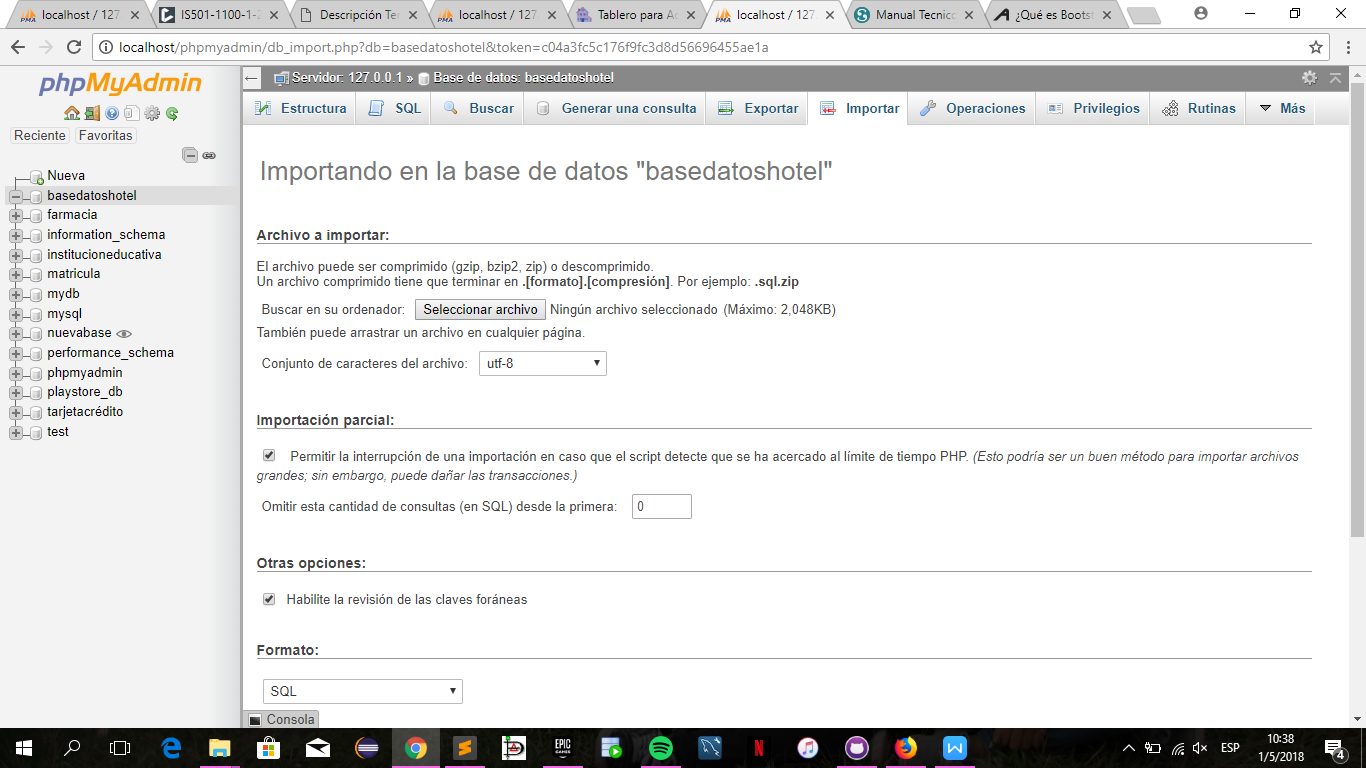
C:\xampp\htdocs\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**02.Inserts\_PersonaTelefonoCliente.sql**

Ó

C:\wamp\www\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**02.Inserts\_PersonaTelefonoCliente.sql**



1. **Volver a “importar”**

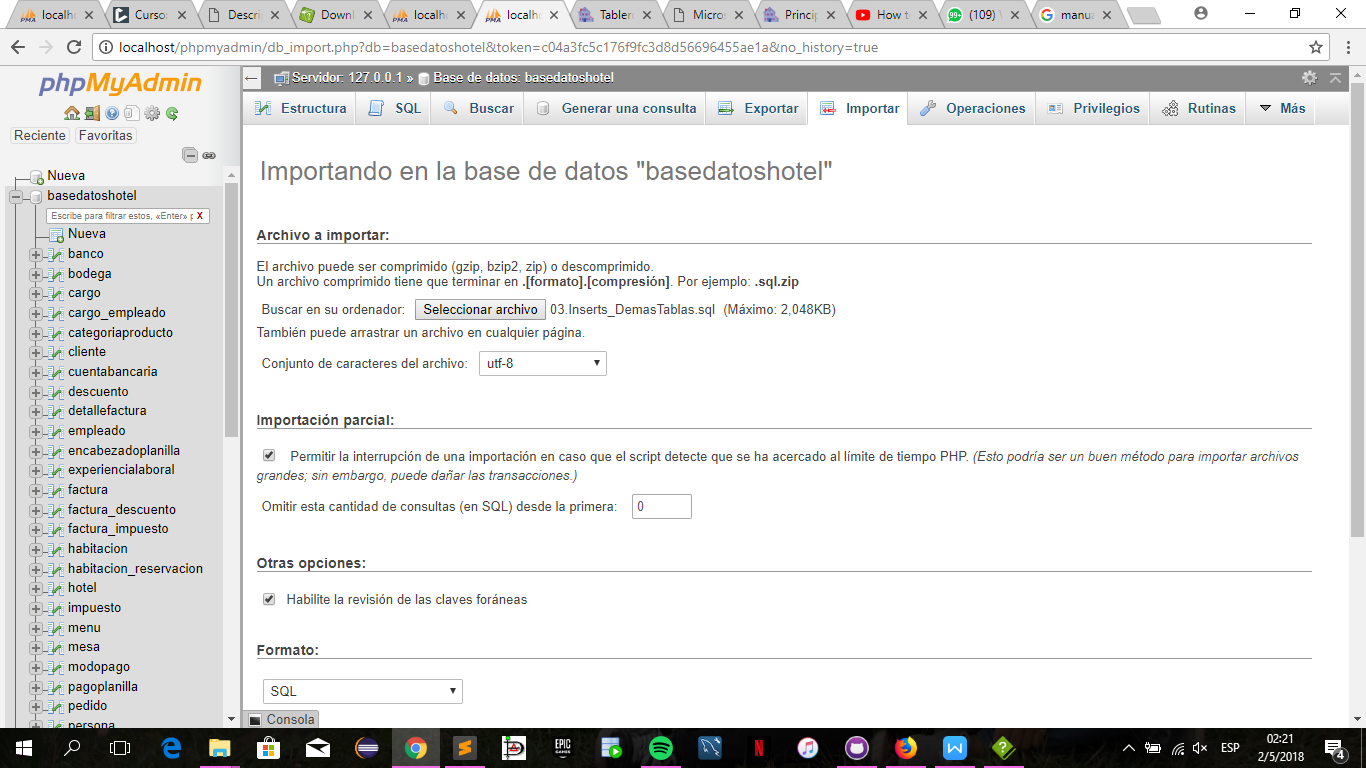


1. **En seleccionar archivo, copiar el siguiente archivo:**

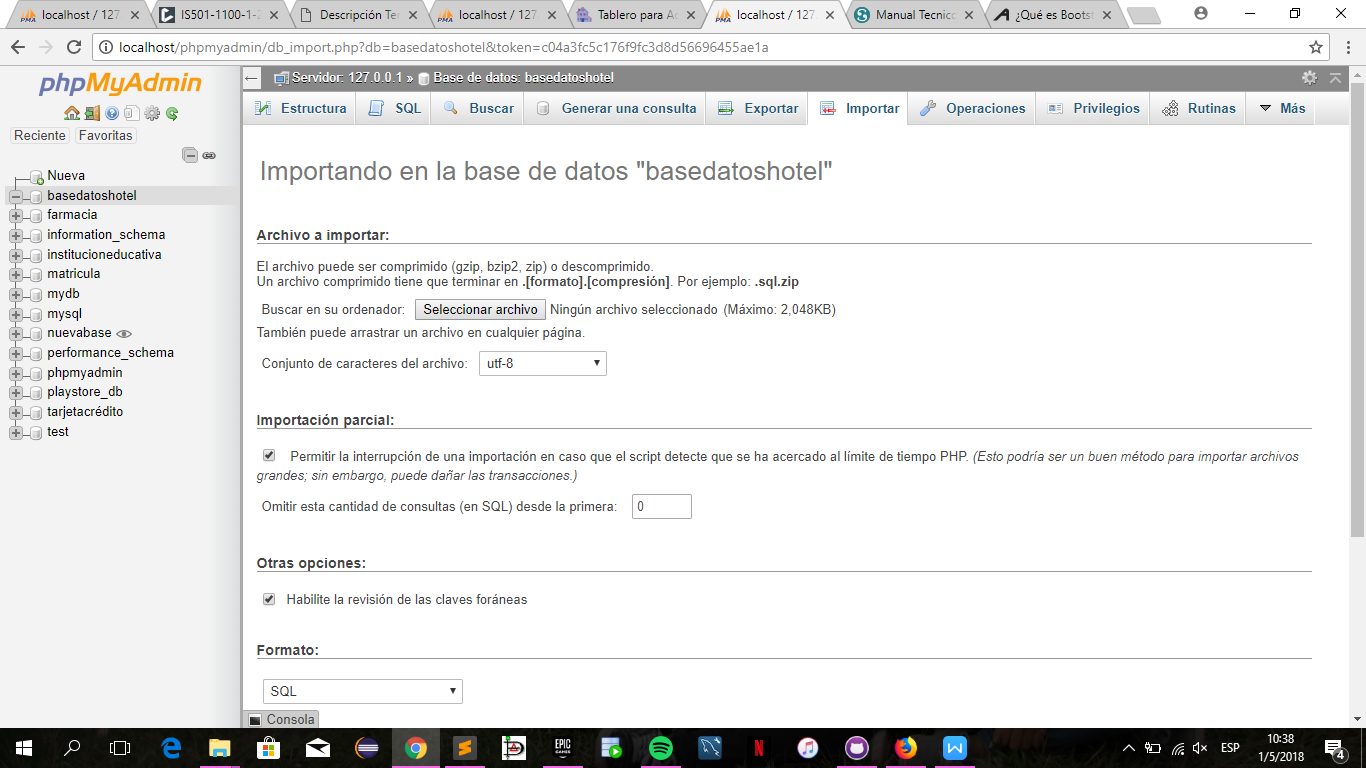
C:\xampp\htdocs\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**03.Inserts\_DemasTablas.sql**

Ó

C:\wamp\www\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**03.Inserts\_DemasTablas.sql**



1. **Volver a “importar”**

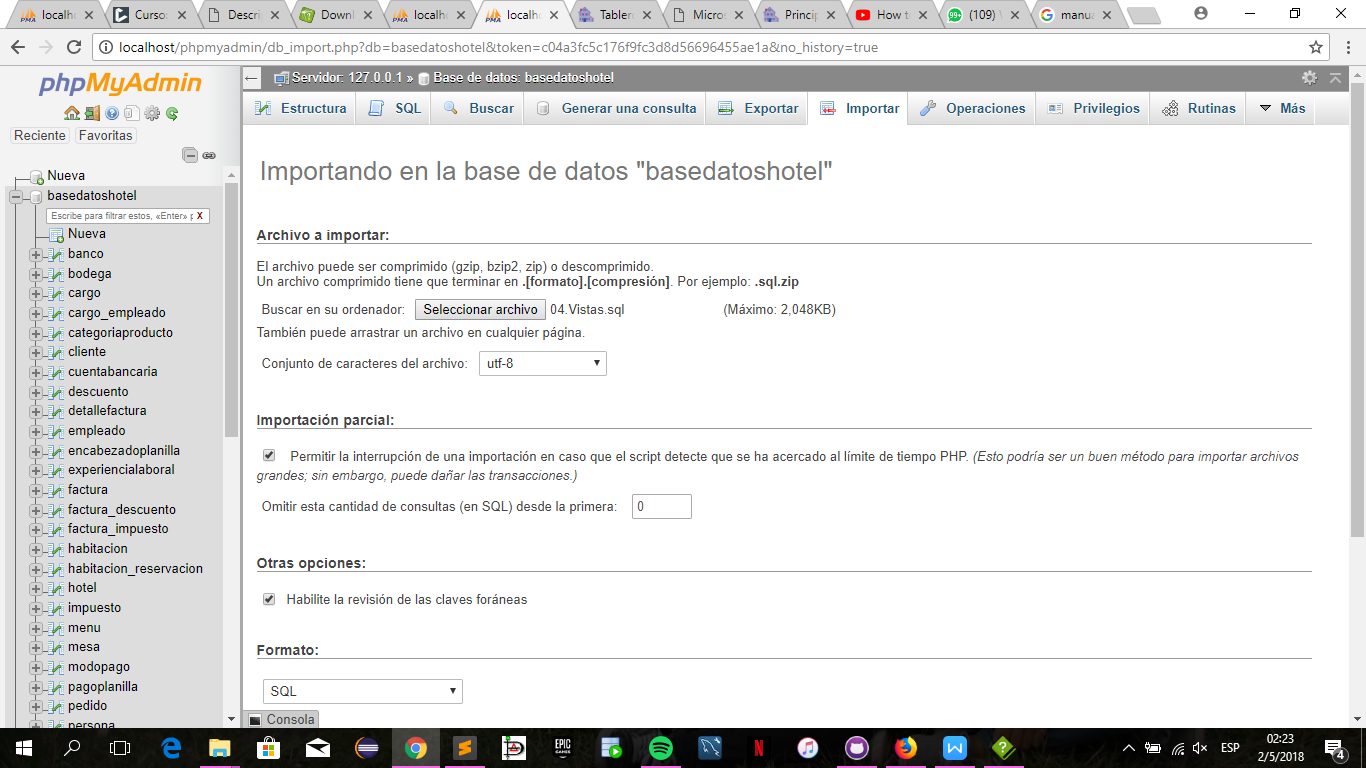


1. **En seleccionar archivo, copiar el siguiente archivo:**

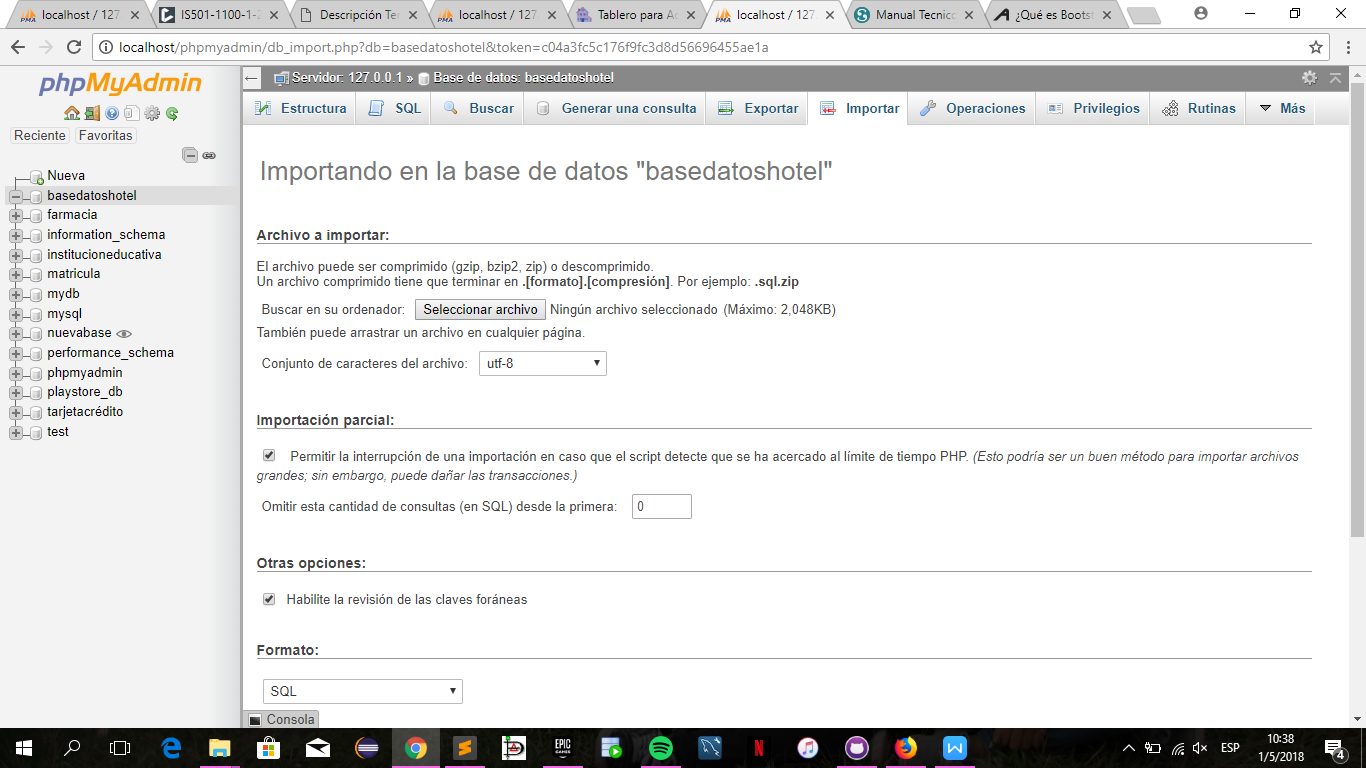
C:\xampp\htdocs\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**04.Vistas.sql**

Ó

C:\wamp\www\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\**04.Vistas.sql**



1. **Volver a “importar”**

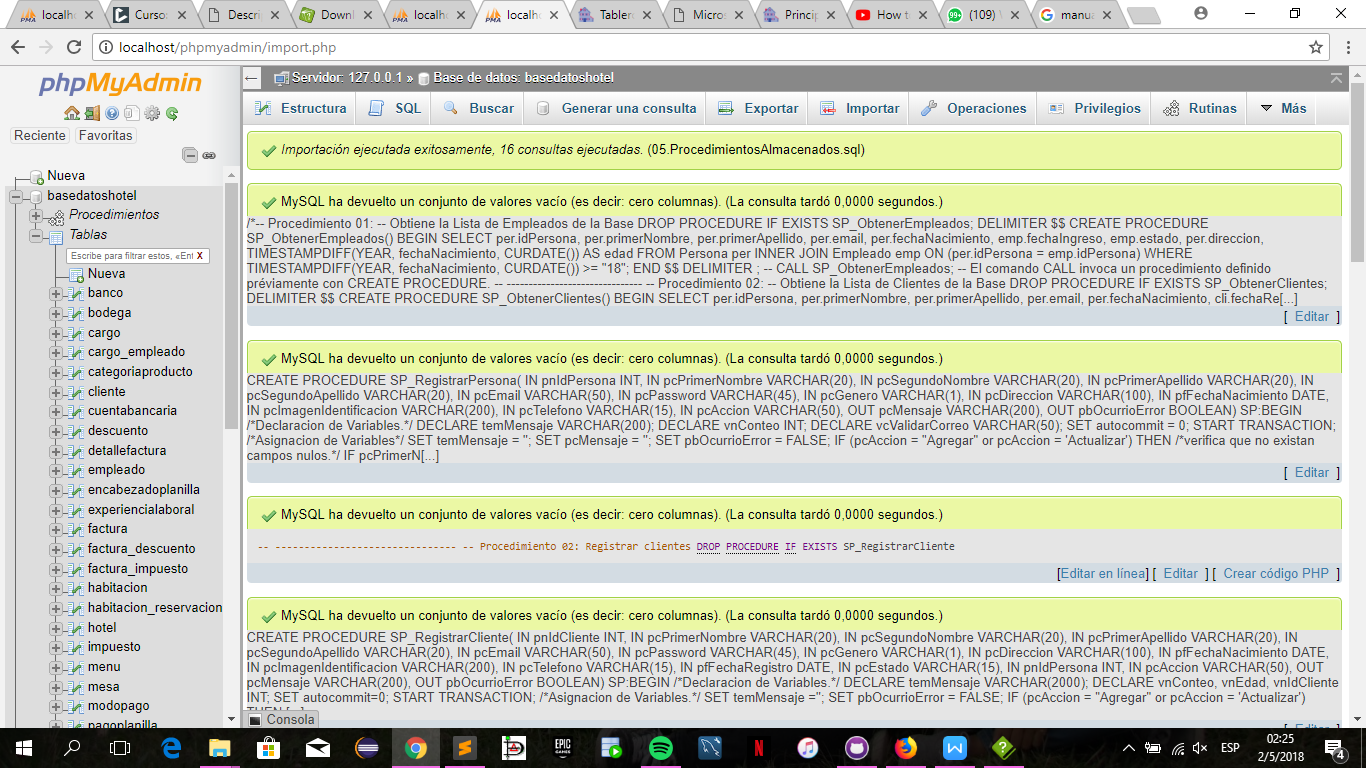


1. **En seleccionar archivo, copiar el siguiente archivo:**

C:\xampp\htdocs\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\procedimientos almacenados\**05.ProcedimientosAlmacenados.sql**

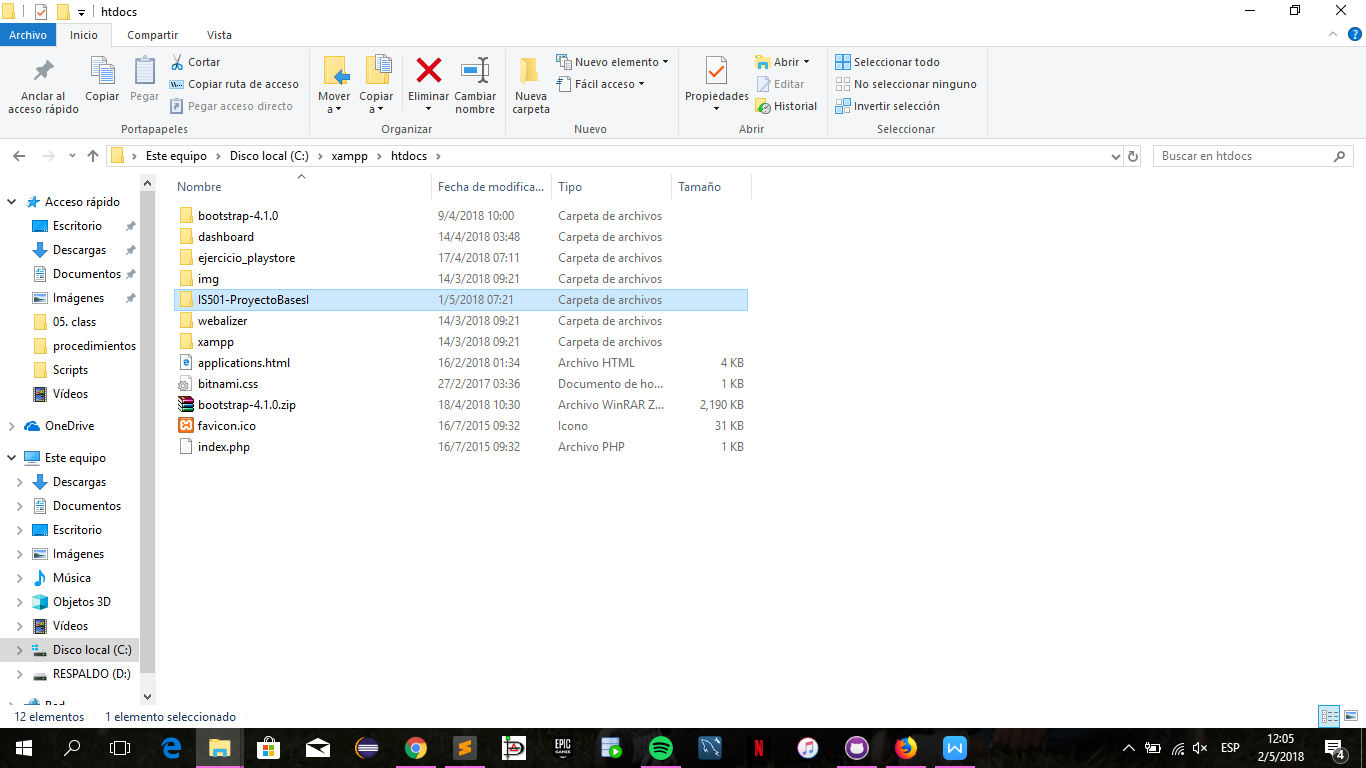
Ó

C:\wamp\www\IS501-ProyectoBasesI\[ProyectoBaseDatos]\Scripts\procedimientos almacenados\**05.ProcedimientosAlmacenados.sql**



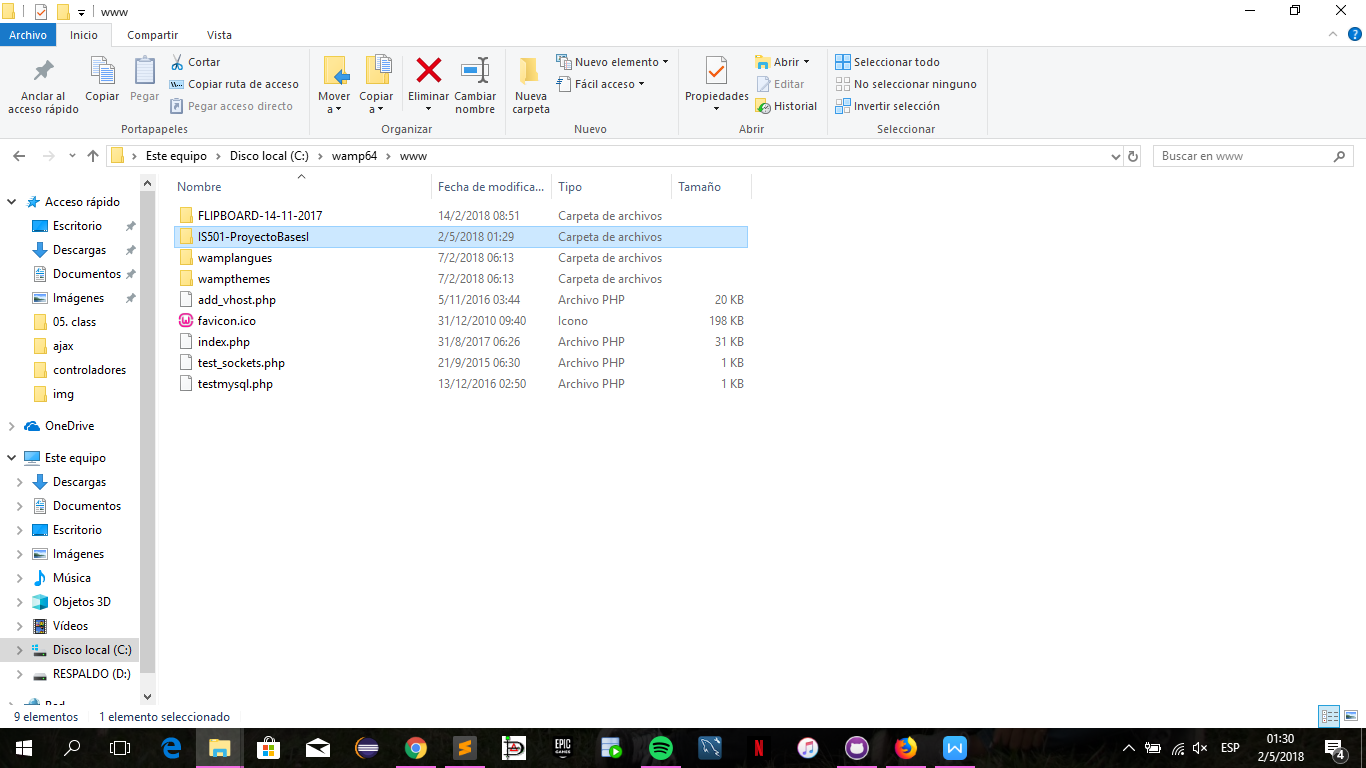
**Agregar el proyecto al xampp/wamp**

1. Copiar la carpeta “IS501-ProyectoBasesI” en : C:\xampp\htdocs

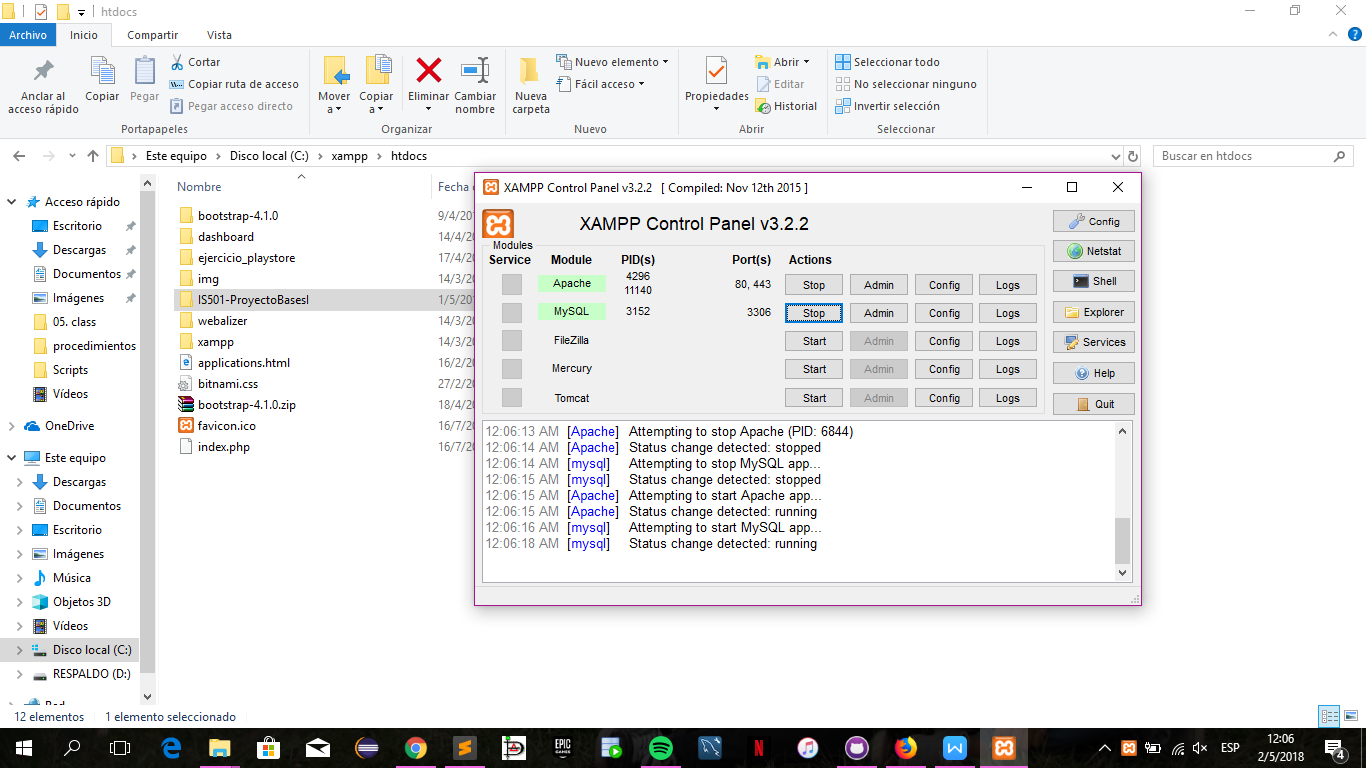


Ó

Copiar la carpeta “IS501-ProyectoBasesI” en: C:\wamp64\www

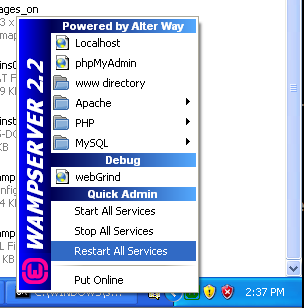


1. Encender el xampp:



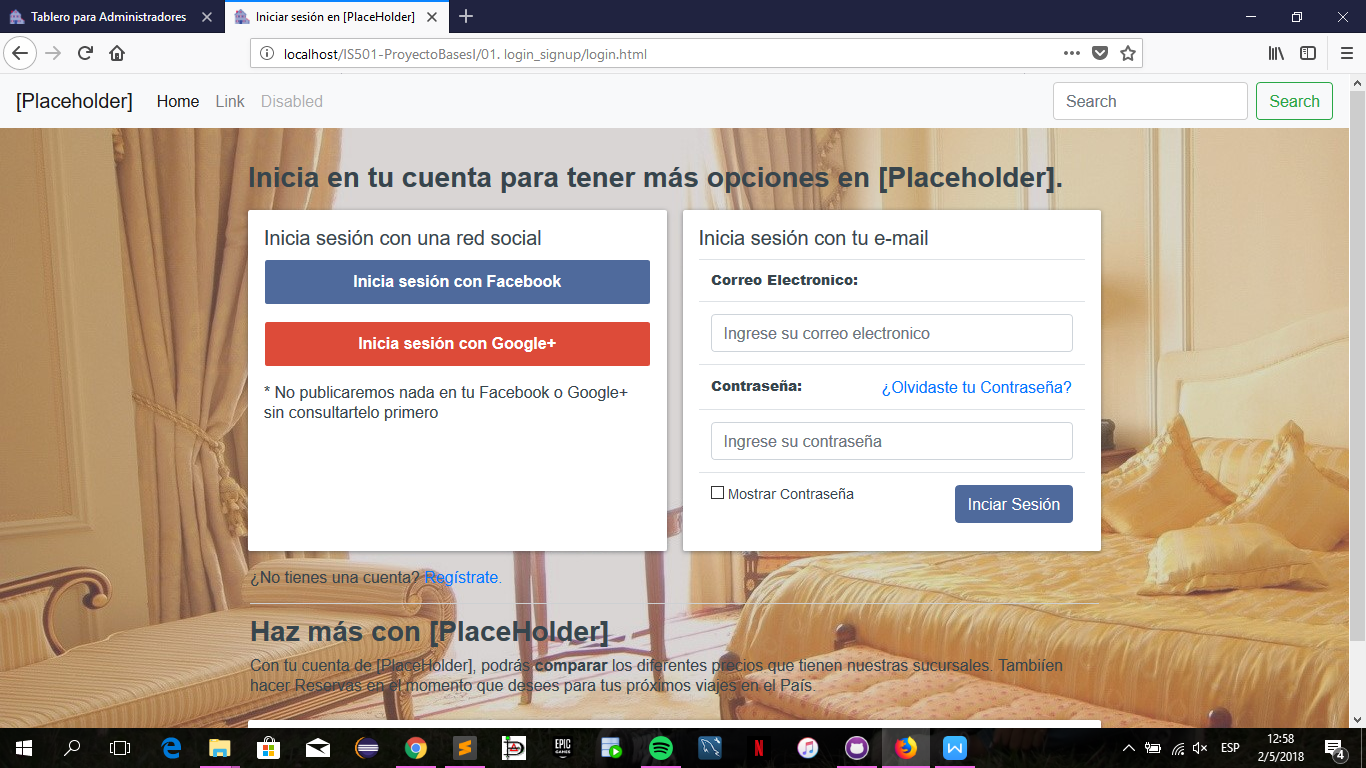
Ó

Encender el wamp

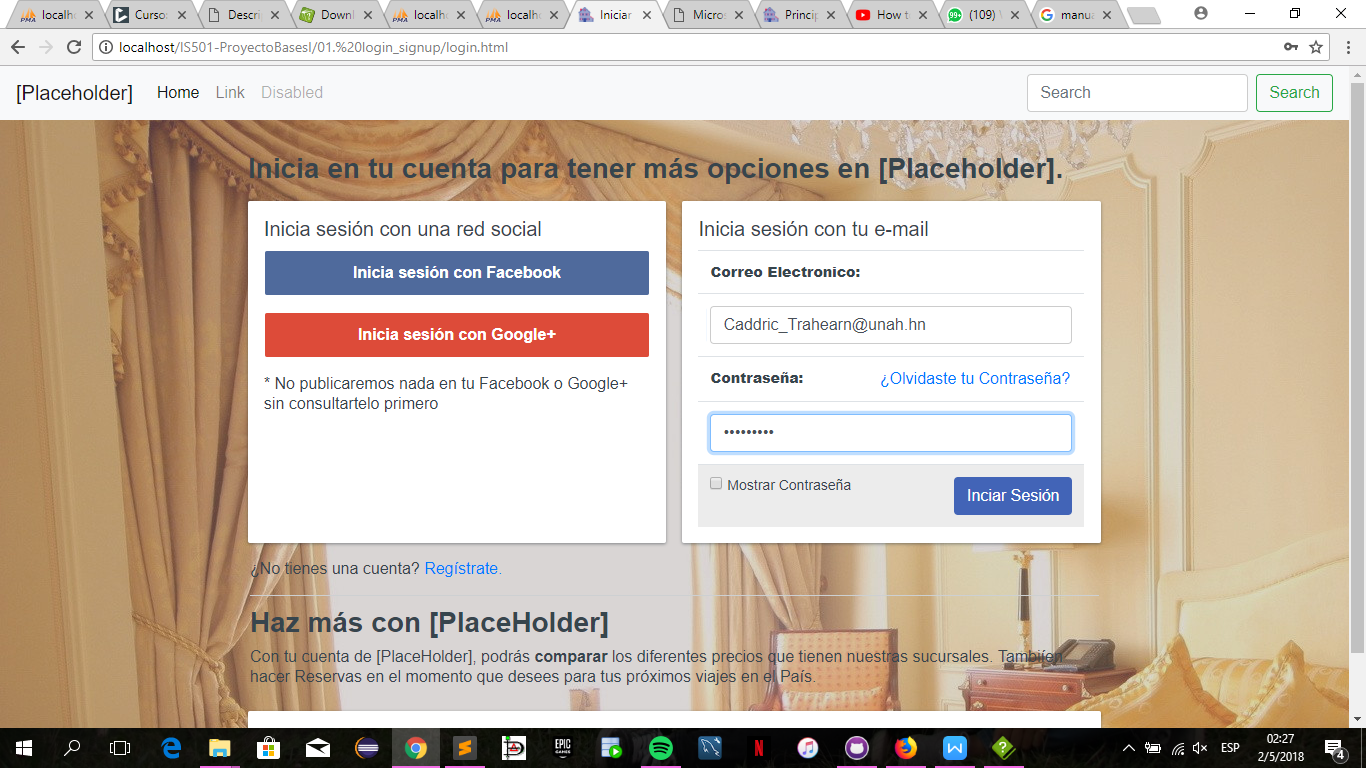


1. Ingresar en cualquier navegador a:

<http://localhost/IS501-ProyectoBasesI/01.%20login_signup/login.html>



1. Ingresar un correo y contraseña válida (ya sea como cliente o empleado)



**Conectar mySQL desde PHP:**

1. Necesitamos crear la clase: class-conexion.php

<?php

class Conexion {

private $host = "localhost";

private $usuario = "root";

private $password = "";

private $baseDatos = "basedatoshotel";

private $puerto = 3306;

private $link;

public function \_\_construct(){

$this->link = mysqli\_connect(

$this->host,

$this->usuario,

$this->password,

$this->baseDatos,

$this->puerto

);

}

public function ejecutarConsulta($sql){

return mysqli\_query ($this->link, $sql);

}

public function ejecutarInstruccion($sql){

return $this->enlace->query($sql);

}

public function obtenerFila($resultado){

return mysqli\_fetch\_array ($resultado);

}

public function obtenerRegistro($resultado){

return mysqli\_fetch\_assoc($resultado);

}

public function cerrarConexion(){

mysqli\_close ($this->link);

}

public function getLink(){

return $this->link;

}

public function antiInyeccion($texto){

return mysqli\_real\_escape\_string ($this->link, $texto);

}

public function ultimoId(){

return mysqli\_insert\_id ($this->link);

}

public function cantidadRegistros($resultado){

return mysqli\_num\_rows($resultado);

}

public function getParametroSP ($parametros) {

$res\_sp = mysqli\_query ($this->link, "SELECT " . $parametros . ";");

$row = mysqli\_fetch\_assoc ($res\_sp);

return $row;

}

}

/\*$sql="SELECT Lastname,Age FROM Persons ORDER BY Lastname";

$result=mysqli\_query($con,$sql);

// Associative array

$row=mysqli\_fetch\_assoc($result);

printf ("%s (%s)\n",$row["Lastname"],$row["Age"]);\*/

?>

1. Conectarlo con la clase cliente

(Función para obtener la lista de clientes desde el aplicativo)

public static function obtenerListaClientes ($conexion) {

$resultado = $conexion->ejecutarConsulta (

'SELECT cli.idCliente, per.primerNombre, per.primerApellido, per.email, per.fechaNacimiento,

cli.fechaRegistro, cli.estado, per.direccion, TIMESTAMPDIFF(YEAR, fechaNacimiento, CURDATE()) AS Edad

FROM Persona per

INNER JOIN Cliente cli ON (per.idPersona = cli.idPersona)

WHERE TIMESTAMPDIFF(MONTH, fechaNacimiento, CURDATE()) >= "216"'

);

// Esto maneja esta consulta con el procedimiento almacenado: CALL SP\_ObtenerClientes

//$resultado = $conexion->ejecutarConsulta("CALL SP\_ObtenerClientes");

while ($fila = $conexion->obtenerFila($resultado)) {

echo '<tr>';

echo '<td id="">' . $fila["idCliente"] . '</td>';

echo '<td id="">' . $fila["primerNombre"] . '</td>';

echo '<td id="">' . $fila["primerApellido"] . '</td>';

echo '<td id="">' . $fila["email"] . '</td>';

echo '<td id="">' . $fila["fechaNacimiento"] . '</td>';

echo '<td id="">' . $fila["fechaRegistro"] . '</td>';

echo '<td id="">' . $fila["estado"] . '</td>';

echo '<td id="">' . $fila["direccion"] . '</td>';

echo '<td><button type="button" onclick="obtenerDetalleCliente('.$fila["idCliente"].')" class="btn btn-default btn-sm" data-toggle="modal" data-target="#modalForm"><span class="fas fa-edit"></span></button>

<button type="button" onclick="eliminarCliente('.$fila["idCliente"].')" class="btn btn-default btn-sm"><span class="fas fa-trash-alt"></span></button></td>';

echo '</tr>';

}

}

(Función para obtener los detalles de clientes desde el aplicativo para poder editarlo/actualizarlo)

public static function obtenerDetalleCliente ($conexion, $idCliente){

//echo "Entra en la funcion";

$resultado = $conexion->ejecutarConsulta(

"SELECT cli.idCliente, per.primerNombre, per.segundoNombre, per.primerApellido, per.segundoApellido,

per.email, tel.numeroTelefono, per.fechaNacimiento, cli.estado, per.direccion

FROM persona per

INNER JOIN cliente cli ON (per.idPersona = cli.idPersona)

INNER JOIN telefono tel ON (per.idPersona = tel.idPersona)

WHERE cli.idCliente = '$idCliente'

");

$fila = $conexion->obtenerFila($resultado);

echo json\_encode($fila);

}

(Parte de la función para registrar clientes, se manda a llamar al procedimiento SP\_RegistrarCliente)

$sql\_callSP = "CALL SP\_RegistrarCliente("

.$null. ",".

"'".$this->primerNombre. "',".

"'".$this->segundoNombre. "',".

"'".$this->primerApellido. "',".

"'".$this->segundoApellido. "',".

"'".$this->email. "',".

"'".$passwordHash. "',".

"'".$this->genero. "',".

"'".$this->direccion. "',".

"'".$this->fechaNacimiento. "',"

.$null. ","

.$telefono. ","

.$null. ",".

"'".$this->estado. "',"

.$null. ",".

"'".$accion."',".

"@pcMensaje,

@pbOcurrioError)";

//echo "<br>Lammado: " .$sql\_callSP ."<br>";

$resultado = $conexion->ejecutarConsulta($sql\_callSP);

(Parte de la función para editar/actualizar clientes, mediante consultas)

$resultado = $conexion->ejecutarConsulta(

"UPDATE persona per

INNER JOIN cliente cli ON (per.idPersona = cli.idPersona)

INNER JOIN telefono tel ON (per.idPersona = tel.idPersona)

SET per.primerNombre = '$this->primerNombre',

per.segundoNombre = '$this->segundoNombre',

per.primerApellido = '$this->primerApellido',

per.segundoApellido = '$this->segundoApellido',

per.email = '$this->email',

per.direccion = '$this->direccion',

per.fechaNacimiento = '$this->fechaNacimiento',

tel.numeroTelefono = '$telefono',

cli.estado = '$this->estado'

WHERE cli.idCliente = '$this->idCliente'

");

echo "<b>Registro actualizado con Exito</b>";

(Función para eliminar clientes, mediante consultas)

public static function eliminarCliente ($conexion, $idCliente){

//echo "Entra en la función";

$resultado = $conexion->ejecutarConsulta(

"DELETE per

FROM persona per

INNER JOIN cliente cli ON (per.idPersona = cli.idPersona)

WHERE cli.idCliente = '$idCliente'

");

echo "<b>Registro Elimimnado con Exito</b>";

}

(Función para registrar una sucursal mandando a llamar al procedimiento almacenado SP\_RegistrarSucursales )

public function registrarSucursal($conexion, $descripcionHotel){

$accion = "Agregar";

$null = "null";

$sql\_callSP = "CALL SP\_RegistrarSucursales("

.$null. ",".

"'".$this->nombre. "',"

.$this->cantidadHabitaciones. ",".

"'".$this->telefono. "',".

"'".$this->email. "',".

"'".$this->direccion. "',".

"'".$this->descripcion. "',".

"'".$this->idRestaurante. "',"

.$null. ",".

"'".$descripcionHotel. "',".

"'".$accion."',".

"@pcMensaje,

@pbOcurrioError)";

echo "<br>Lammado: " .$sql\_callSP ."<br>";

$resultado = $conexion->ejecutarConsulta($sql\_callSP); // mysqli\_query ($this->link, $sql);

$return = $conexion->getParametroSP("@pcMensaje, @pbOcurrioError");

if ($resultado != '1') { //

echo "Error: " . $resultado . " <br>";

}

$mensajeSP = $return['@pcMensaje'];

$ocurreError = $return['@pbOcurrioError'];

if ($ocurreError == "1"){

echo '<b>'. $mensajeSP . '</b>'." !@!true" . " <br>";

} else {

echo "<b>Registro Insertado con Exito</b><br>";

//echo $mensajeSP . " !@!false" . " <br>";

}