

APRENDE A PROGRAMAR EN

PHP YA!

y Domina las Bases de Datos

Tu primer base de datos de
PHP con MySQL paso a paso

Gratis!!

<http://AprendeaprogramarEnPHPya.com/blog>

CONTENIDO

Introducción	3
Capítulo1 (Bases de Datos)	
Tienda de CD's	4
Tablas, Campos y Atributos	6
Capítulo 2 (Función PHP para conectar con MySQL)	
Archivo de Configuración	9
Función para conectar	10
Capítulo 3 (Insertando datos desde un Formulario HTML)	
Código Inserta Panel	12
Código Alta Productos y/o Cajeros	16
Capítulo 4 (Programar Consultas a la Base de Datos)	
Código Consulta Panel	18
Código para ejecutar consulta	19
Capítulo 5 (Eliminar Registros)	
Código Elimina Panel	21
Código para eliminar registros	25
Capítulo 6 (Modifica Registros con PHP y MySQL)	
Código Modifica Panel	27
Código para modificar registros	30
Capítulo 7 (Ventas y Reportes)	
Ventas Sesión	34
Verifica (uso de contraseña)	35
Código para realizar ventas	36
Sentencia SQL para consulta de dos tablas	38
Reportes	39
Sentencia SQL para consulta de tres tablas	40

INTRODUCCION

Este tutorial esta redactado y pensado en ayudar a los nuevos interesados en aprender a programar en php ya, así que si ya tomaste la decisión de aprender php verdaderamente o bien tienes que presentar algún proyecto de forma urgente este es el tutorial mas adecuado que estabas buscando, no te detengas léelo hasta el final.

Pero, ¿qué vas a encontrar en este tutorial?, El desarrollo paso a paso de un proyecto completo, eso si sencillo pero completo. Si sigues todo el tutorial paso a paso al final tendrás desarrollado un sistema de base de datos en php que te servirá para llevar el control de una tienda de discos.

Si eres un completo novato te recomiendo que primero veas los 3 videos del curso básico que te regalo al registrarte en mi página <http://AprendeAProgramarEnPHPya.com> y enseguida leas este tutorial. Entre otras recomendaciones que te puedo dar es que sigas paso a paso este tutorial, trata de no brincar secciones y ve realizando todos los códigos que te voy proponiendo para que al final no tengas ningún problema con tu producto final y ya una vez funcional el producto final, ahora si puedas agregarle cosas tuyas, características personales que eso te hará aprender aún mas.

Con los tres videos aprendes porque aprendes a programar en php, lo que si es que para volverte profesional necesitas tomar experiencia y esta solo se logra programando, manteniéndose actualizado y por su puesto compartiendo.

A un último detalle, este proyecto esta desarrollado sobre la configuración automática de XAMPP y no voy a poner el link de descarga porque mi objetivo, mi deseo es que aprendas a programar en php ya y para ello es necesario que al menos al principio hagas tus propios script.

Manos a la obra y comencemos con este gran tutorial:

Capítulo 1: BASE DE DATOS

La base de datos sobre la que se va a trabajar en este tutorial debe tener la siguiente la estructura:

Base de datos: **TiendaCDs**

Tablas

Campos

Productos	id, nombre, codigo, tipo, genero, precio, existencia
Ventas	id, fecha, codigo_producto, cantidad, id_cajero
Cajeros	id, nombre, pass, permisos

A los campos les faltan saber los tipos de datos que se van a guardar en ellos, esto lo sabrás ahora que te diga como configurar esto en el phpMyAdmin.

Nota: Para este paso ya debes de tener instalado todo lo necesario en tu computadora para poder realizar los ejercicios, en caso contrario puedes ver los 3 videos del curso básico de programación en php que te regalo al registrarte solo con tu nombre y tu correo electrónico en <http://AprendeAProgramarEnPHPya.com>

Vamos paso a paso y con capturas de pantalla, pero antes del primer paso para configurar la base de datos asegúrate que se este ejecutando el apache y el mysql. Si tienes el Xampp lo puedes ver abriendo el panel del control del mismo, y en donde dice apache debe estar una leyenda que diga “*running*” esto mismo debe aparecer en la etiqueta de mysql. En la imagen 1.1 puedes ver un ejemplo del panel de control.

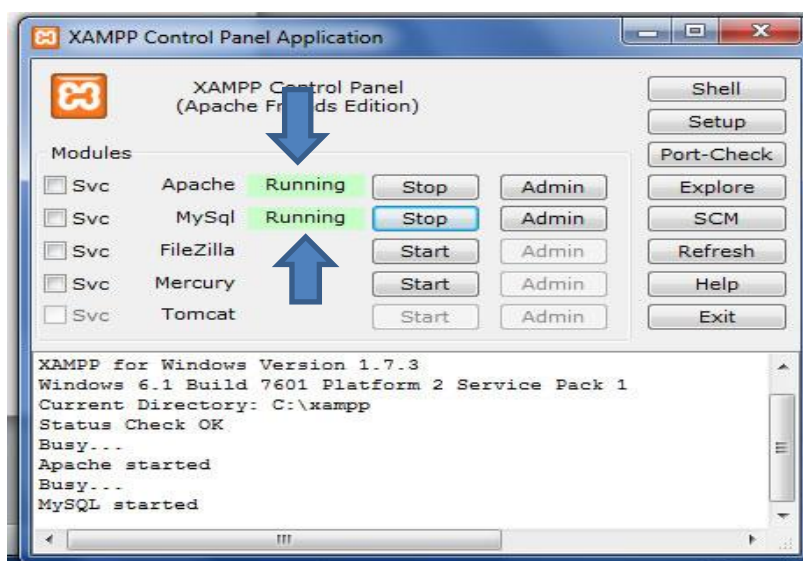


Imagen 1.1 Panel de Control del Xampp

Bien, una vez que estés seguro que se encuentra corriendo en tu computadora el Apache y el Mysql, abre el explorador de internet que utilices (google chrome, internet explorer, firefox, opera, etc.) y anotas dentro de la barra de direcciones “<http://localhost/phpmyadmin>” te tiene que dar un resultado muy parecido al que se muestra en la imagen 1.2.

Nota: Si configuraste tu computadora con otro paquetería diferente al Xampp o bien lo realizaste manualmente, asegúrate de que tengas instalado el phpMyAdmin. Para que veas un resultado parecido al de la imagen 1.2. Puedes ver como realizar la instalación en <http://AprendeAProgramarEnPHPya.com/blog/>

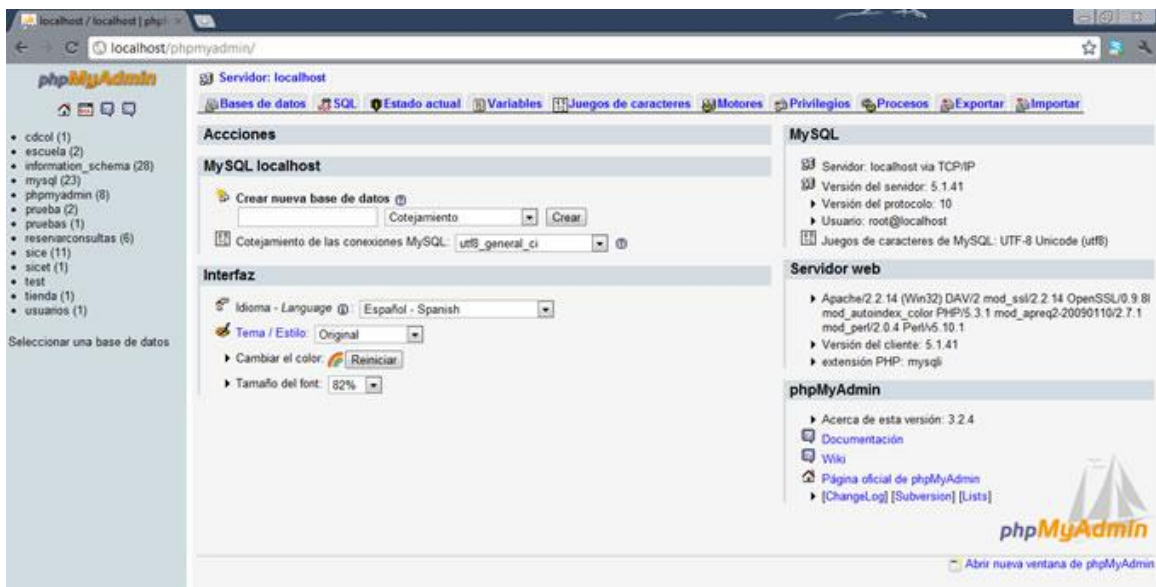


Imagen 1.2 – Interfaz del phpMyAdmin.

Una vez que encuentres en la interfaz principal del phpMyAdmin, vas a crear la base de datos **TiendaCDs**, para hacer esto ubica en la interfaz la parte que dice “*Crear nueva base de datos*” y anota el nombre TiendaCDs antes de darle click en crear. Utiliza la imagen 1.3 de referencia.



Imagen 1.3 – Como crear la base de datos TiendaCDs

Si realizaste esto de forma correcta ahora deberá aparecerte una nueva pantalla de phpMyAdmin donde te solicita crear una nueva tabla para la base de datos recién creada. En el primer cuadro de texto que aparece y que pide el nombre le vas a anotar **“productos”** y donde dice número de campos le vas a anotar **“7” (que es el número de campos que lleva la tabla productos)** y luego le das click en continuar, si todo va bien te debe mostrar una pantalla parecida a la que se puede ver en la siguiente imagen 1.4.

Imagen 1.4 – Interfaz de phpMyAdmin para capturar los campos con sus atributos.

Como puedes ver en la imagen, hay 7 filas cada una corresponde a uno de los campos que vas a capturar con todos sus atributos. Así que comencemos con los atributos del campo id: en campo anotas *id*, en tipo seleccionas *int*, en índice seleccionas *primary* y luego le das click en la caja de selección que dice *A_I* (que significa *autoincremento*).

Nota: Todas las tablas deben tener una llave primaria, así que en este proyecto notarás que siempre utilizo un campo llamado id, auto-incremental como llave primaria.

Para hacer este tutorial practico no voy a explicar que significan cada uno de los atributos solo te diré cuales debes de ponerle a cada campo, aclarado este punto sigamos. Estos son los atributos correspondientes a los 6 campos que faltan y debes configurar (los atributos que no se mencionan se dejan como están):

- **Campo:** nombre, **tipo:** varchar, **longitud/valores:** 20.
- **Campo:** codigo, **tipo:** varchar, **longitud/valores:** 10.
- **Campo:** tipo, **tipo:** varchar, **longitud/valores:** 20.
- **Campo:** genero, **tipo:** varchar, **longitud/valores:** 10.
- **Campo:** precio, **tipo:** int

- **Campo:** existencia, **tipo:** int

Una vez capturados los datos anteriores das click en el botón que dice grabar, ubicado en el fondo de la página para que se guarde la configuración de esta nueva tabla. Para saber si lo realizaste correctamente debes de tener en pantalla un resultado parecido a la que muestra la imagen 1.5, verifica que diga que la tabla se creó (ver flecha roja).

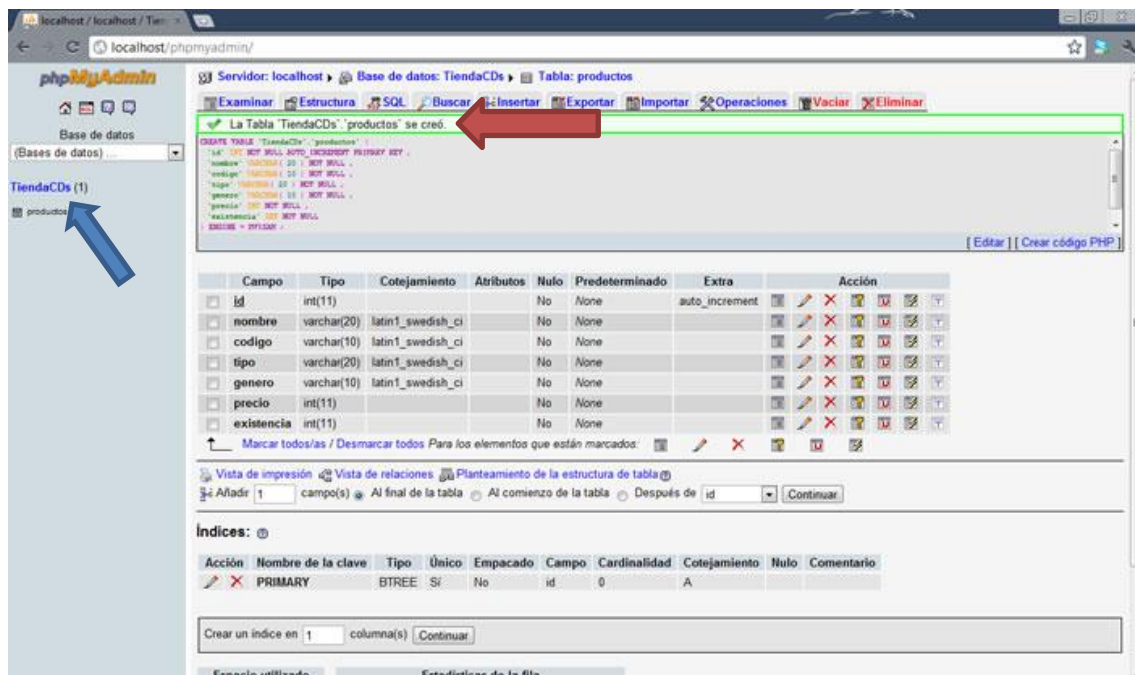


Imagen 1.5 – phpMyAdmin después de configurar correctamente la tabla productos

Hasta este punto se ha creado la base de datos TiendaCDs con una tabla llamada productos. Ahora falta crear las otras dos tablas con sus campos, para esto le damos click donde dice TiendaCDs (observa la flecha azul de la imagen 1.5). Y en la pantalla que aparece debes ubicar donde dice **“Crear nueva tabla en la base de datos TiendaCDs”** y en el cuadro de texto que dice **“Nombre:”** anotas **“ventas”** y en el cuadro que dice **“Número de campos:”** pones **5**. Cuando le des click en continuar te va mandar nuevamente a una pantalla parecida a la de la imagen 1.4 pero ahora con 5 filas. En esa pantalla vas a configurar los siguientes campos con sus atributos como sigue:

- **Campo:** id, **tipo:** int, **índice:** primary, **A_I** :seleccionado
- **Campo:** fecha, **tipo:** datetime
- **Campo:** código_producto, **tipo:** varchar, **longitud/valores:** 10.
- **Campo:** cantidad, **tipo:** int
- **Campo:** id_cajero, **tipo:** int

Y vuelves a darle click en TiendaCDs como lo muestra la flecha azul en la imagen 1.5, repitiendo los pasos anteriores para ahora dar de alta la tabla **“cajeros”** con los campos y atributos siguientes:

- **Campo:** id, **tipo:** int, **índice:** primary, **A_I** :seleccionado
- **Campo:** nombre, **tipo:** varchar, **longitud/valores:** 20.
- **Campo:** pass, **tipo:** varchar, **longitud/valores:** 5.
- **Campo:** permisos, **tipo:** int.

Verifica que al final tengas tus tres tablas, para hacerlo puedes comparar tu pantalla con la de la imagen que te muestro a continuación y asegúrate que la parte marcada de la imagen 1.6, también la tengas en tu pantalla.



Imagen 1.6 – phpMyAdmin mostrando la base de datos TiendaCDs con tres tablas (cajeros, productos y ventas)

Has terminado la primer sección de este tutorial que es donde se realiza la configuración de la base de datos sobre la que vas a trabajar el resto del proyecto, así que no continúes hasta que estés seguro de haber completado correctamente este primer capítulo, si por alguna razón no explico correctamente como llevar acabo la configuración de las tablas, campos y atributos, sabes que cuentas con el blog: <http://AprendeAProgramarEnPHPya.com/blog>.

Capítulo 2: FUNCIÓN EN PHP PARA CONECTAR CON LA BASE DE DATOS

Ahora que ya tienes configurada tu base de datos, es tiempo de crear el código que se va a encargar de conectar el php con el motor de base de datos de MySQL. Para que esto sea mas practico y te puedas ahorrar un buen número de líneas de código, vas a realizar esta tarea en un archivo que se encargue de guardar las funciones responsables de llevar acabo esta tarea.

Nota: No avances si no has configurado tu base de datos, ya que es necesariamente importante para continuar, desarrollando correctamente este proyecto.

Así que vas a crear 2 archivos, el primero de ellos va a tener los datos de configuración esto lo haces por si tuvieras que cambiar de servidor solo pones en este archivo los nuevos datos de configuración y tu código sigue funcionando sin la necesidad de tener que cambiar cada aparición de los datos en tu proyecto. Entonces abre un archivo nuevo al que llamas **"config_mysql.php"** y le anotas el siguiente código:

```
<?php
//servidor
define ('HOST','TU HOST');
//usuario bd
define ('USER','TU USUARIO');
//pass
define ('PASS','TU CONTRASEÑA');
//base de datos
define ('DB','TiendaCDs');

?>
```

Te explico este código simple: lo que hace este código es declarar las constantes **HOST**, **USER**, **PASS** y **DB**, con los valores que están después de la coma, así que ahora tienes que cambiar los datos por los tuyos, por ejemplo, si estas trabajando en tu maquina local lo mas seguro es que la constante HOST deba tener de valor "localhost", entonces la línea quedaría *define ('HOST','localhost');* una vez que hayas editado este código correctamente guárdalo en la siguiente dirección, **"xampp/htdocs/tiendacd/funciones/"**.

Nota: Si no configuraste tu instalación con el xampp, entonces no vas a tener la ruta que se encuentra arriba, la solución es que ubiques la carpeta **htdocs** y dentro de ella guardes tu proyecto, comúnmente esta dentro de la carpeta **Apache**.

Es la hora del segundo archivo y es en realidad el que va a tener la función encargada de conectar la base de datos, para poder tener acceso a ello y ejecutar la tarea que en caso se requiera, como es insertar, consultar y/o modificar. Vamos paso a paso:

- Abre un nuevo documento en el editor que estés usando (si no sabes como, te recomiendo que veas los 3 videos de conceptos básicos de php que te regalo en <http://AprendeAprogramarEnPHPya.com/blog/>).
- Anótale el siguiente código:

```
<?php

include ("config_mysql.php");

function conectar(){

    $con = mysql_connect (HOST,USER,PASS)
    or die ("ERROR EN CONEXION: ".mysql_error());
    $base_datos = mysql_select_db (DB,$con)
    or die ("ERROR AL SELECCIONAR BASE DE DATOS: ".mysql_error());

    return $con;

}

?>
```

- Un párrafo mas adelante te explico que onda con este código, mientras guardarlo con el nombre de **"funciones_mysql.php"** y dentro de la carpeta **"xampp/htdocs/tiendacd/funciones/"** o bien en la ruta donde estés guardando este proyecto.

Ahora si te explico el código: lo primero es indicarle que este script va a requerir los datos del archivo **"config_mysql.php"** por eso lo incluimos dentro de este proyecto con la función definida **include** y entre paréntesis la ubicación relativa o exacta del archivo a incluir. Enseguida se declara una función llamada **conectar()** y no recibe ningún parámetro.

Nota: Si quieres profundizar mas en el uso de funciones recuerda que cuentas con el sitio <http://AprendeAprogramarEnPHPya.com/blog>.

Dentro de la función, lo que esta en negritas son funciones definidas del php:

- **mysql_connect:** Se utiliza para realizar la conexión con del php con el mysql y recibe tres parámetros:

- **El host** al que se quiere realizar la conexión, si estas trabajando en tu maquina y con la configuración por defecto de xampp es *localhost*.
- **El usuario** que tiene permisos para ingresar a esa base de datos.
- **El password** que le configuraste a ese usuario para tener interacción con el mysql.
 - **or die:** Si la conexión con la base de datos o bien la selección de la base de datos no se dio con éxito, imprime en la pantalla *ERROR EN CONEXIÓN* o *ERROR AL SELECCIONAR BASE DE DATOS* o bien lo que hayas anotado entre las comillas.
 - **mysql_error():** Esta función sirve para cachar y mostrar el error que provoco que la función en cuestión no se llevara acabo con éxito.
 - **return:** Regresa el link de conexión que se necesita para llevar acabo las tareas propias de las base de datos (mas adelante se te aclarara este punto).

Bien hasta aquí tienes listo tu base de datos, el archivo con la configuración para conectar tu PHP con el motor de MySQL y el archivo con el script que se necesita para realizar con éxito la conexión a tu MySQL.

Capítulo 3: INSERTANDO DATOS DESDE UN FORMULARIO HTML

Ahora sí, una vez completada las tareas anteriores es hora de realizar una de las tareas básicas de toda base de datos, como lo es dar de alta productos o dicho de otra forma de insertar datos. Para esto vas a tener que programar 2 archivos distintos, uno va ser encargado de mostrar el formulario en html que va a solicitar los nuevos datos a ingresar y el otro es el que precisamente se va a encargar de insertarlos, es decir, el script en php encargado de meter los datos del formulario en la tabla correspondiente.

Analiza un poco la base de datos y las tablas que creaste, **tienes la tabla productos, la tabla ventas y la tabla cajeros**, entonces se entiende que tienes que dar de alta tanto productos como cajeros para poder realizar ventas, verdad?. Pues bien, vas a crear el menú en html para seleccionar si quieres insertar en productos o en cajeros, el código en cuestión puede quedar así:

```
<html>
<head>
<title>:: Menú para insertar datos ::</title>
</head>

<body>
<h2> --> SELECCIONA DANDO CLICK LO QUE DESEAS HACER <-- </h2>
<ul>
<li><a href='insertar.php?id=1'>INSERTAR CAJEROS</a></li>
<li><a href='insertar.php?id=2'>INSERTAR PRODUCTOS</a></li>
</ul>
</body>
</html>
```

Una vez que tengas escrito el código guardar el archivo en una carpeta llamada **tiendacds** y ponle el nombre de **inserta_panel.html** (recuerda guardar la carpeta en htdocs si estas trabajando en tu maquina). Es hora del script en php, pero te explico que función debe tener; debe identificar primero que desea insertar el usuario, **cajeros (id = 1) o productos (id=2)**, luego debe mostrar el formulario adecuado para finalmente insertar los datos, esto lo puedes realizar con el siguiente código y como el objetivo es que aprendas a programar en php, voy a ir mostrando y explicando lo mejor que pueda cada parte del código en el entendido que al final debe ser uno solo. Esta primer parte del código solo es para iniciar el código html.

```
<?php
echo "<html>
    <head><title>:: FORMULARIOS PARA INSERTAR ::</title></head>
    <body>";
```

```
1 //verifica que se selecciono
2 if ($_GET['id']==1){ //inserta cajeros
3 //muestra el formulario para cajero
4 echo "<h2>CAPTURA LOS DATOS DEL NUEVO CAJERO</h2>
5     <form name='inserta' method='post' action='$PHP_SELF'>
6     <strong>Nombre:</strong><input type='text' name='nombre'><br />
7     <strong>Pass:</strong>(no mayor a 5 caracteres)<input type='text' name='pass'
maxlength='5'><br />
8     <strong>Permisos:</strong>
9     <select name='permisos'>
10         <option value='1'>Solo ventas</option>
11         <option value='2'>Reportes y ventas</option>
12     </select>
13     <input type='submit' name='boton' value='Alta_Cajero'>
14 </form>";
15 } //fin del if get
```

Nota: **echo** se utiliza para imprimir en el explorador o en pantalla.

Nota: “//” estas dos diagonales indican que de allí en adelante esa línea es comentario. También se puede utilizar /* para indicar el inicio de comentarios y */ para terminar.

Te explico esta otra parte del código, en la línea 2 checamos si se desea insertar cajeros o productos, indicado por la variable **id** que viene de la página anterior por la barra de direcciones (es decir, utilizando el método GET), si la condición se cumple se imprime en el explorador el código HTML encargado de mostrar el formulario que muestra los campos solicitando los datos a insertar. En la línea 5 donde dice **action='\$PHP_SELF'** significa que los datos los va enviar a esta misma página, es decir, se recarga esta página pero recibe los datos por el método post (indicado en esa misma línea) capturados en el formulario y por último de este trozo de código la línea 13 es **type='submit'** para indicarle que cuando se presione ese botón se envíe el formulario, observa que se crea una instancia de nombre **boton** y con un valor **Alta_Cajero**.


```
1     else{//inserta productos
2     //muestra el formulario para producto
3     echo "<h2>CAPTURA LOS DATOS DEL NUEVO PRODUCTO<h2>
4         <form name='inserta' method='post' action='$PHP_SELF'>
5         <strong>Nombre:</strong><input type='text' name='nombre'><br />
6         <strong>Codigo:</strong><input type='text' name='codigo' maxlength='10'><br
7         <strong>Tipo:</strong>(Musica, pelicula)<input type='text' name='tipo'><br />
8         <strong>Genero:</strong><input type='text' name='genero'><br />
9         <strong>Precio por unidad ($):</strong><input type='text' name='precio'><br />
10        <strong>Existencia:</strong><input type='text' name='existencia'><br />
11        <input type='submit' name='boton' value='Alta_Producto'>
12        </form>";
13    }//fin del else get
```

Con esta parte de código se imprime el formulario para insertar productos y se muestra en los casos donde la variable **id** adquirida por el método GET sea diferente de 1 inclusive ejecuta esta sección si no define **id**. El código es muy similar al anterior así que doy por entendido que se comprende esta parte del código.

```
1 //si le dio click al boton de alta cajero o producto
2 //entra aqui
3 if (isset($_POST['boton'])){
4     /*****
5     *****cargamos las funciones mysql *****/
6     *****/
7     include ("funciones/funciones_mysql.php");
8     conectar();
9     /*****/
```

Nota: En el capítulo de crear la conexión se explica a detalle como se puede realizar aquí solo usamos esa función ya programada. Puedes obtener mas información en <http://AprendeAProgramarEnPHPya.com/blog>

No pierdas de vista que este código es parte del mismo archivo y que solo lo he dividido en partes para explicarlo con mas detalle. En esta parte del código lo que buscas es verificar que se ha presionado el botón **submit** de alguno de los dos formularios (que tienen el mismo nombre porque solo se mostrara uno a la vez), en caso afirmativo se llama la función encargada de conectar con MySQL y con la base de datos a manejar.

```
1 //depende cual se selecciono es la
2 //sentencia sql a usar
3 if ($_POST['boton']=='Alta_Producto'){
4 $sql="INSERT INTO productos (id,nombre,codigo,tipo,genero,precio,existencia)
5 VALUES (null, '$_POST[nombre]', '$_POST[codigo]', '$_POST[tipo]',
6 '$_POST[genero]', '$_POST[precio]', '$_POST[existencia]");
7 $result = mysql_query($sql)
8 or die("ERROR AL EJECUTAR CONSULTA PRODUCTOS:".mysql_error());
9
10 echo "<strong><center>DATOS INSERTADOS CORRECTAMENTE <a
    href='inserta_panel.html'>Volver al menu, da click aqui </a></center></strong>";
11 }//fin if alta_producto
```

De esta parte del código en la línea 3 preguntamos si la instancia **boton** que ha sido presionado tiene el valor de **Alta_Producto**, en caso que se cumpla, creas la sentencia SQL encargada de insertar datos en la tabla correspondiente.

Nota: La estructura de las sentencia para insertar con SQL es :**INSERT INTO nombredetabla (campo1,campo2...) VALUES (valorcampo1,valorcampo2,...)**.Puedes obtener mas información en <http://AprendeAProgramarEnPHPya.com/blog>

Un detalle importante es que las cadenas van entre comillas y los valores tipo numero van sin comillas. En la línea 6 de este trozo de código se ejecuta la sentecia con la función predefinida para ello **mysql_query**.

```
1 //si no fue alta producto entonces es
2 //alta cajero
3 else{
4 $sql = "INSERT INTO cajeros (id,nombre,pass,permisos) VALUES
5 (null,'$_POST[nombre]', '$_POST[pass]', '$_POST[permisos]");
6
7 $result = mysql_query($sql)
8 or die("ERROR AL EJECUTAR CONSULTA PRODUCTOS:".mysql_error());
9
10 echo "<strong><center>DATOS INSERTADOS CORRECTAMENTE <a
    href='inserta_panel.html'>Volver al menu, da click aqui </a></center></strong>";
11 }//fin alta_cajero
12 }//fin boton submit
13 echo "</html>";
14 ?>
```

Este es la parte final del código, y en el caso que se haya presionado el **boton** tipo submit y este traiga un valor diferente a **Alta_Producto** entra y ejecuta las líneas 4 al 10 del código. En la línea 12 cierras el código HTML y en la 13 cierras el bloque de php.

Por fin terminamos el archivo encargado de insertar los datos ya sea en la tabla de productos como en la tabla cajeros, el código final debe quedar como sigue:

```
1 <?php
2 echo "<html>
3     <head><title>:: FORMULARIOS PARA INSERTAR ::</title></head>
4     <body>";
5 //verifica que se selecciono
6 if ($_GET['id']==1){ //inserta cajeros
7 //muestra el formulario para cajero
8 echo "<h2>CAPTURA LOS DATOS DEL NUEVO CAJERO<h2>
9     <form name='inserta' method='post' action='$PHP_SELF'>
10     <strong>Nombre:</strong><input type='text' name='nombre'><br />
11     <strong>Pass:</strong>(no mayor a 5 caracteres)<input type='text' name='pass'
maxlength='5'><br />
12     <strong>Permisos:</strong>
13     <select name='permisos'>
14         <option value='1'>Solo ventas</option>
15         <option value='2'>Reportes y ventas</option>
16     </select>
17     <input type='submit' name='boton' value='Alta_Cajero'>
18     </form>";
19 }//fin del if get
20 else{//inserta productos
21 //muestra el formulario para producto
22 echo "<h2>CAPTURA LOS DATOS DEL NUEVO PRODUCTO<h2>
23     <form name='inserta' method='post' action='$PHP_SELF'>
24     <strong>Nombre:</strong><input type='text' name='nombre'><br />
25     <strong>Codigo:</strong><input type='text' name='codigo' maxlength='10'><br
/>
26     <strong>Tipo:</strong>(Musica, pelicula)<input type='text' name='tipo'><br />
27     <strong>Genero:</strong><input type='text' name='genero'><br />
28     <strong>Precio por unidad ($):</strong><input type='text' name='precio'><br />
29     <strong>Existencia:</strong><input type='text' name='existencia'><br />
30     <input type='submit' name='boton' value='Alta_Producto'>
31     </form>";
32 }//fin del else get
33
34 //si le dio click al boton de alta cajero o producto
35 //entra aqui
36 if (isset($_POST['boton'])){
37     /*****
38     ***** cargamos las funciones mysql *****
39     *****/
40     include ("funciones/funciones_mysql.php");
41     conectar();
42     /*****
43     *****/
44 }
```

```
44
45
46
47     //depende cual se selecciono es la
48     //senten
49     cia sql a usar
50     if ($_POST['boton']=='Alta_Producto'){
51         $sql="INSERT INTO productos (id,nombre,codigo,tipo,genero,precio,existencia)
52         VALUES (null, '$_POST[nombre]', '$_POST[codigo]', '$_POST[tipo]',
53         '$_POST[genero]', '$_POST[precio]', '$_POST[existencia])";
54         $result = mysql_query($sql)
55         or die("ERROR AL EJECUTAR CONSULTA PRODUCTOS:".mysql_error());
56
57         echo "<strong><center>DATOS INSERTADOS CORRECTAMENTE <a
58         href='menu.html'>Volver al menu, da click aqui </a></center></strong>";
59     }
60     //fin if alta_producto
61     //si no fue alta producto entonces es
62     //alta cajero
63     else{
64         $sql = "INSERT INTO cajeros (id,nombre,pass,permisos) VALUES (null,
65         '$_POST[nombre]', '$_POST[pass]',$_POST[permisos])";
66         $result = mysql_query($sql)
67         or die("ERROR AL EJECUTAR CONSULTA PRODUCTOS:".mysql_error());
68
69         echo "<strong><center>DATOS INSERTADOS CORRECTAMENTE <a
70         href='menu.html'>Volver al menu, da click aqui </a></center></strong>";
71     }
72     //fin alta_cajero
73 }
74 //fin boton submit
75
76
77 echo "</html>";
78 ?>
```

Bueno antes de dar por terminado el capitulo 3, déjame decirte que este código esta testeado y ha sido copiado desde el editor de código a este tutorial precisamente para asegurarme de su funcionamiento. Te pido de favor que si seriamente quieres Aprender A Programar en PHP ya, revises este tutorial paso a paso y transcribas los códigos, esta es la razón por la que **NO VOY A PONER LOS CODIGOS PARA DESCARGAR**, solo este tutorial.

Capítulo 4: PROGRAMAR CONSULTAS A LA BASE DE DATOS

Todas las tareas que se ejecutan a la base de datos son importantes, pero desde mi punto de vista esta es la más importante ya que una base de datos que no pueda ser consultada para obtener la información que en ella se encuentre pues prácticamente no serviría de nada.

Para que aprendas a programar estas consultas, vas a realizar dos archivos el primero de ellos es solo para que permita seleccionar a que tabla se desea ejecutar la consulta, a la tabla productos o la tabla cajeros.

Nota: Las consultas a múltiples tablas y a la tabla de ventas la aprenderás en el capítulo 7 de este tutorial. Puedes obtener mas información en <http://AprendeAProgramarEnPHPya.com/blog>

El archivo en cuestión es un HTML y tendrá el nombre de “**consulta_panel.html**”, en él estará un formulario que permita seleccionar la tabla a consultar y capturar el nombre a buscar, este es el código:

```
1 <html>
2 <head>
3 <title>:: Menú para Consultar Datos ::</title>
4 </head>
5
6 <body>
7 <h2>Realiza una Consulta</h2>
8 <form name="consulta" method="post" action="consulta.php">
9 <strong>Puedes anotar el nombre o bien hacer uso de comodin "%":</strong><br />
10 <input type="text" name="nombre" />
11 Tabla a Consultar:<select name="tabla">
12 <option value="productos">BUSCAR PRODUCTO</option>
13 <option value="cajeros">BUSCAR CAJEROS</option>
14 </select>
15 <input type="submit" name="Consultar" value="Consultar" />
16 </form>
17 </body>
18 </html>
```

De este código no hay mucho que explicar, solo me enfocare en la línea 8 que es donde le estamos indicando que la información será enviada por el método “POST” (oculta a la barra de direcciones) al archivo “**consulta.php**” que ahorita vas a programar. El resultado del código anterior debe ser una pantalla igual a la imagen 4.1.



Realiza una Consulta

Puedes anotar el nombre o bien hacer uso de comodin "%":

Tabla a Consultar:

Imagen 4.1 Muestra la interfaz del código de consulta_panel.html

Nota: Si no sabes como ver el resultado del archivo, puedes ver los 3 videos tutoriales básicos gratuitos en <http://AprendeAProgramarEnPHPya.com>

Es la hora de aprender a programar en php las consultas así que abre un archivo nuevo y anota en él, el siguiente código:

```
1 <?php
2 //incluir la funcion encargada de conectar
3 include("funciones/funciones_mysql.php");
4
5 //extrae los datos enviados por el formulario
6 //a la variable con el nombre del elemento
7 //por ejemplo; el elemento nombre
8 //queda como $nombre = valor
9 extract($_POST);
10
11 //llamar la función que conecta con el motor MySQL
12 conectar();
13 //diseñas la sentencia SQL
14 $sql = "SELECT * FROM $tabla WHERE nombre LIKE '$nombre'";
15 //ejecutas la consulta
16 $result = mysql_query($sql);
17 //imprime el encabezado de la pagina resultado
18 echo "<html><head><title>Resultado Consulta</title></head>
19      <body><table border=1>";
20 //sacas los valores del arreglo
21 while ($campo = mysql_fetch_array($result)){
22     if ($tabla == 'cajeros'){
23         echo "<tr><td>$campo[id]</td>
24             <td>$campo[nombre]</td><td>$campo[pass]</td><td>$campo[permisos]</td>";
25     }else{
26         echo "<tr><td> $campo[id]</td><td>$campo[nombre]</td><td>$campo[codigo]
27 </td><td>$campo[tipo]</td><td>$campo[genero]</td><td>$campo[precio]</td>
28 <td>$campo[existencia]</td>";
29     }
30 }
31 echo "</table><center><strong>Realizar otra b&uacute;squeda</strong>
32     <a href='consulta_panel.html'>Click Aquí</a></body></html>";
33
34 ?>
```

Te explico, el código para su mejor comprensión; de la línea 1 a la 14 esta bastante entendible con lo que esta comentado en el mismo código (recuerda que tienes disponibles los artículos en <http://AprendeAprogramarEnPHPya.com/blog>, para profundizar en lo que no comprendas), así que pasemos a la línea 14 donde esta la sentencia SQL, y mas especifico la palabra **"LIKE"** es la que permite realizar búsquedas con comodín, por ejemplo, si deseo buscar todos los nombre de cajero que empiecen con la **S** entonces puedo poner en la caja de texto **"S%"** o bien si deseo obtener todos los registros donde el nombre del producto incluya la palabra **"cosa"** en cualquier parte del nombre, puedes anotar **"%cosa%"** y esto lo puedes lograr gracias al **LIKE** del SQL.

Creo que es lo único comentable de este código, lo demás esta entendible con lo comentado en el mismo y si no, no te preocupes notarás como hay partes del código muy repetitivas que terminaras por comprender y dominar fácilmente, claro siempre y cuando realices estos ejemplos manualmente.

Bueno esto es todo por este capítulo espero haber sido claro y sobretodo haber logrado alcanzar mi objetivo primordial que es **QUE APRENDAS A PROGRAMAR EN PHP YA como realizar una consulta simple.**

Capítulo 5: CODIGO PARA ELIMINAR DATOS

Es el turno a una de las tareas básicas que también es muy importante, la de eliminar registros de una base de datos, pero claro desde una interfaz en HTML.

Como eliminar datos con código php es una de las consultas mas buscadas en internet ya que de repente resulta difícil comprender como arreglar lógicamente tu código para que solo elimine ese registro que tú quieres y no todos los registros de la tabla. Así que estoy plenamente seguro que aquí aprenderás como realizarlo de forma sencilla.

Bueno continúa con tu proyecto ya que el tiempo apremia. Vas a necesitar en este punto realizar nuevamente dos archivos, uno para mostrar los datos que tienes capturados con la opción de seleccionar el o los registros que quieres eliminar y el otro que va a ser el encargado de ejecutar la sentencia SQL para eliminar el dato.

En el primer archivo va a tener que mostrar resultados en HTML, pero esos resultados son obtenidos por una consulta realizada en PHP y estos deben ser seleccionables para poder realizar la eliminación del o los registros deseados. Seguiremos utilizando la misma dinámica, te expongo primero el código comentado y luego te explico las partes que a mi punto de vista son menos entendibles.

El siguiente código transcríbelo en un archivo nuevo al que vas a llamar **“elimina_panel.php”**:

```
1 <html>
2 <head>
3 <title>:: PANEL PARA ELIMINAR</title>
4 </head>
5
6 <body>
7 <h2> :: PANEL PARA ELIMINAR :: </h2>
8 <strong>Para eliminar primero debes consultar los registros, asi que en la siguiente caja 9
9 de texto puedes anotar el nombre del o los producto(s) o cajero(s) que desees eliminar. El
10 comodín que soporta es el simbolo->"%"</strong>
11 <!-- PHP_SELF es para recargar esta misma página cuando le den click
12 en el elemento tipo submit, enviando los datos del formulario por post -->
13 <form name="elimina" method="post" action="<?php echo $PHP_SELF; ?>">
14 Nombre:<input type="text" name="nombre" />
15 Tabla a Consultar:<select name="tabla">
16 <option value="productos">BUSCAR PRODUCTO</option>
17 <option value="cajeros">BUSCAR CAJEROS</option>
18 </select>
19 <input type="submit" name="consulta" value="Enviar">
20 </form>
```

```
21 <?php
22 include ("funciones/funciones_mysql.php");
23 extract($_POST);
24 /*preguntamos si existe la instancia "consulta", es
25 el tipo submit del formulario y se crea cuando se envia el formulario
26 es decir si existe es porque ya se envio la consulta */
27 if (isset($consulta)){
28 //llamar la función que conecta con el motor MySQL
29 conectar();
30 //diseñas la sentencia SQL
31 $sql = "SELECT * FROM $tabla WHERE nombre LIKE '$nombre'";
32 //ejecutas la consulta
33 $result = mysql_query($sql);
34 /*imprime la tabla con los resultados y con la opcion de
35 seleccionar para eliminarlo*/
36 echo "<form name='envia_eliminar' method='post' action='elimina.php'>
37     <table>";
38 //genera el encabezado de la tabla con los resultados
39 echo $tabla=="cajeros" ? "<tr><td>ID</td><td>NOMBRE</td><td>PASSWORD</td>
40     <td>PERMISOS</td><td>SELECCIONA</td></tr>":
41     "<tr><td>ID</td><td>NOMBRE</td><td>CODIGO</td>
42     <td>TIPO</td><td>GENERO</td><td>PRECIO</td>
43     <td>EXISTENCIA</td><td>SELECCIONA</td></tr>";
44 //sacas los valores del arreglo
45 while ($campo = mysql_fetch_array($result)){
46     if ($tabla == 'cajeros'){
47         echo "<tr><td>$campo[id]</td><td>$campo[nombre]</td>
48             <td>$campo[pass]</td><td>$campo[permisos]</td>
49             <td><input type='checkbox' name='cajeros.$campo[id]'
50             value='$campo[id]'></td></tr>";
51     }else{
52         echo "<tr><td>$campo[id]</td><td>$campo[nombre]</td>
53             <td>$campo[codigo]</td><td>$campo[tipo]</td>
54             <td>$campo[genero]</td><td>$campo[precio]</td>
55             <td>$campo[existencia]</td>
56             <td><input type='checkbox' name='productos.$campo[id]'
57             value='$campo[id]'></td></tr>";
58     }
59 }
60 //fin del while
61 echo "</table>
62 <input type='hidden' name='tabla' value='$tabla'><input type='submit' name='envia'
63 value='Eliminar' /></form>";
64 }
65 //fin del isset consulta ?>
66 </body>
67 </html>
```

Este código ya está un poco más largo y complejo, porque combina una consulta con código HTML para maquetar el resultado. Déjame explicarte las partes más complicadas (a mi consideración de este archivo), lo primero que voy a hacer es mostrarte en la imagen 5.1 el resultado de este código en el explorador:

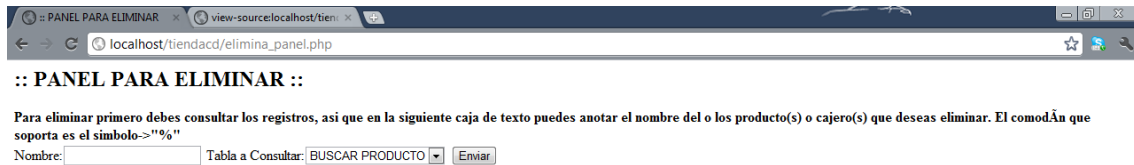


Imagen 5.1 – Primer pantalla que muestra el archivo elimina_panel.php

La intención es que el usuario pueda buscar de la tabla productos o de la tabla cajeros el o los datos que desea eliminar, es para eso que está el formulario de la línea 13 – 20, este formulario cuando se le da click al botón de enviar manda los datos con el método **post** a la misma página, esto se logra con lo que está en la línea 13, **action="<?php echo \$PHP_SELF; ?>"**.

De la línea 21 a la 42 creo que está entendible con los comentarios dentro del código y de otra forma ya lo he explicado en este mismo tutorial capítulos atrás o bien puedes reforzar tu aprendizaje con los artículos que están en la página <http://AprendeAProgramarEnPHPya.com/blog>. En la línea 44 está el **while** que se encarga de ir obteniendo en un arreglo asociativo llamado **campo** con subíndices el nombre del campo, esto lo hace por filas con cada uno de los resultados obtenidos de la consulta, por ejemplo: *“supongamos que el usuario pone **%pedro%** en la caja nombre del formulario mostrado por elimina_panel.php y en la tabla están los siguientes registros capturados:*

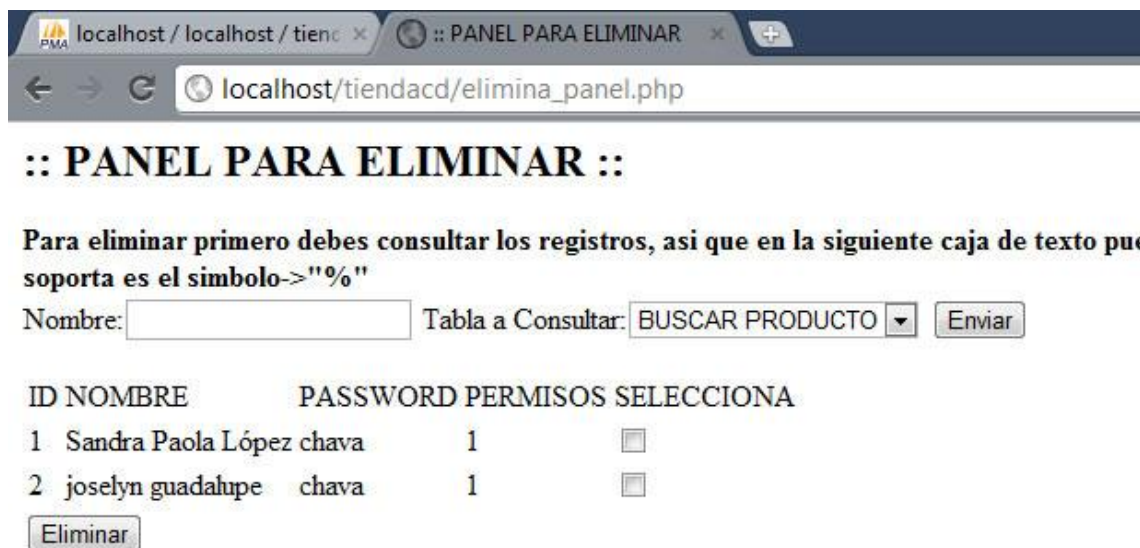
- 1 --- Jose Luis Sanchez Hinojoza --- elquesea --- 1
- 2 --- Pedro Rivera Solis --- password --- 1
- 3 --- Oscar Pedro Rivas --- contrase --- 1

*El primer resultado que va a entrar al while y por lo tanto al **mysql_fetch_array** es la fila con el id igual a 2, entonces esos datos quedarían en el arreglo como sigue: **\$campo[id]=2**, **\$campo[nombre]=Pedro Rivera Solis**, **\$campo[pass]=password** y **\$campo[permisos]=1** y lo mismo haría con la fila con id igual a 3, porque tiene el nombre Pedro en alguna parte del nombre (note que no distingue entre mayúsculas y minúsculas)”*

No sé si quedo claro con ejemplo espero que sí. Una vez obtenido los datos de la fila en el arreglo, los acomodamos en la tabla y le agregamos un campo más del tipo checkbox, (cuyo nombre del campo es “checkbox” más el id del campo para evitar nombre repetidos, lo puedes ver en la línea 48 y 55 depende la tabla seleccionada, puedes ver un ejemplo del resultado en la imagen 5.2) para tener la opción de poder seleccionar el o los registros a

eliminar y así poderlos identificar en el programa que se va a encargar de ello, un ejemplo de como se muestran los resultados lo puedes ver en la imagen 5.2.

Nota: De la línea 38 a la 42 es para imprimir el encabezado de la tabla resultado y este depende si se seleccionó “cajeros” o “productos”, entonces se hace uso del if (yo le llamo de la forma elegante) con la siguiente estructura: **condición ? valor si verdadero : valor si falso** ; Hay un artículo donde te explico esto a detalle en <http://AprendeAprogramarEnPHPya.com/blog>.



:: PANEL PARA ELIMINAR ::

Para eliminar primero debes consultar los registros, así que en la siguiente caja de texto pue soporta es el simbolo->"%"

Nombre: Tabla a Consultar:

ID	NOMBRE	PASSWORD	PERMISOS	SELECCIONA
1	Sandra Paola López chava	chava	1	<input type="checkbox"/>
2	joselyn guadalupe	chava	1	<input type="checkbox"/>

imagen 5.2 – Muestra el resultado de buscar %s% en la tabla cajeros

Es hora de programar el código encargado de eliminar con php y mysql los datos que selecciones de los resultados dados por la consulta, estos datos son enviados al archivo **elimina.php** por el método **post** del formulario que cubre la tabla resultado (puedes ver este formulario en la línea 35). Para terminar la explicación de este código pasemos a la línea 60 donde dice: **<input type='hidden' name='tabla' value='\$tabla' >** indica que el formulario va a enviar un elemento mas con el nombre de la tabla a trabajar y este es de manera oculta al explorador. Es hora de Aprender A Programar En PHP ya como eliminar registros de una tabla, este es el código que puedes utilizar:

```
1 <?php
2 //todo lo necesario para conectar con
3 //el mysql
4 include ("funciones/funciones_mysql.php");
5 //llamamos la funcion encargada de conectar
6 //con mysql
7 conectar();
8 //esta funcion extrae las variables
9 //enviadas por el formulario
10 extract($_POST);
11
12 /* nota, solo va a llegar el id que se haya
13 Seleccionado en el formulario resultado de la
14 busqueda, asi que por cada uno lo eliminamos y punto*/
15 $cont =0; //para saber cuantos se eliminan
16 foreach ($_POST as $campo=>$valor){
17     //preguntamos si en el campo hay algo que diga
18     //producto o cajero, de encontrar algo significa
19     //que viene minimo un seleccionado para eliminar
20     //si es asi diseñamos la sentencia sql encargada
21     //y aumentamos en 1 el contador
22     if(strstr($campo,"producto") || strstr($campo,"cajero")){
23         $sql = "DELETE FROM $tabla WHERE id='$valor'";
24         mysql_query($sql) or die ("ERROR AL ELIMINAR REGISTRO  ". $cont.
mysql_error());
25         $cont++;
26     }
27     //borramos el contenido de la variable sql
28     //para volver a crear la nueva
29     $sql="";
30 }
31
32 //si no hubo error llega hasta esta linea
33 echo "<center><strong>$cont</strong> REGISTRO(S) ELIMINADOS CORRECTAMENTE
34     <a href='elimina_panel.php'> VOLVER AL PANEL </a></center>";
35 ?>
```

De este código bastara con explicar la línea 16 y la 22 ya que lo demás es sencillo de comprender con lo comentado dentro del código. En la línea 16 se aprovecha la funcionalidad que ofrece el **foreach** el cual va recorriendo el arreglo uno por uno, y poniendo el nombre de la instancia lanzada por el formulario en la variable **\$campo** y el valor en la variable **\$valor**. Permíteme explicarte con un ejemplo: “Supongamos que tienes un formulario con tres instancias: `<input type='text' name='nombre' value='HOLA' />`, `<input type='radio' name='radio' value='2' />` y `<input type='checkbox' name='chechar' value='4' />` y estas se

envían para ser procesadas por otro archivo. En este otro archivo se encuentra el siguiente código:

```
<?php foreach($_POST as $campo=>$valor)
    echo $campo."-->".$valor."<br />"; ¿>
```

El resultado del código seria:

nombre-->HOLA

radio -->2

chechar --> 4

Como puedes ver la variable **\$campo** va cachando el atributo **name** de cada instancia lanzada por el formulario y la variable **\$valor** cacha el valor que trae esa instancia."

Verdad que esta sencillo de comprender cual es el funcionamiento de esta fabulosa función y recuerda que siempre cuentas con la página: <http://AprendeAProgramarEnPHPya.com/blog> para reforzar el aprendizaje y lograr mas pronto tu objetivo de aprender a programar en php.

En la línea 22 tienes que identificar si la variable **\$campo** tiene en alguna parte **producto** o **cajero** porque si lo tiene significa que es la variable cuyo valor es el numero **id** del registro que necesitamos eliminar.

No voy a dar ningún capítulo por terminado sin recordarte que este e-book lo he escrito con la única intención de que aprendas a programar en php ya de una forma fácil y sencilla donde además termines con un proyecto completo desarrollado y entendido al 100% por ti. Puedes entregar este proyecto en tu escuela o bien puedes encontrar en cada capítulo como se llevan a cabo las tareas básicas de una base de datos con php, así que dudes en compartir este e-book, pero mas importante aún no dejes de traspasar el código manualmente con eso me aseguro que aprendes mejor que si solo haces "copy, paste", **recuerda que no voy a poner los códigos descargables. Para evitar a los que solo buscan que alguien les resuelva todo, si a ti realmente te interesa aprender a programar en php sabrás valorar esta información.**

Ahora si, hasta aquí queda terminado el capítulo 5: **CODIGO PARA ELIMINAR DATOS**, en el siguiente capítulo aprenderás como puedes modificar los registros capturados, así que no te detengas falta poco para que tengas tú primer proyecto en php terminado.

Capítulo 6: MODIFICAR REGISTROS CON PHP Y MYSQL.

Todos los humanos estamos expuestos a cometer errores a la hora de capturar algún registro o también puede haber alguna actualización en los datos que alguna vez se insertaron, pensando en estas dos tareas tan comunes una base de datos no puede quedar completa sin su parte sistematizada de modificación, por eso este capítulo esta diseñado para que aprendas a realizar esta tarea.

Igual que para eliminar vas a necesitar programar dos archivos, uno para permitir al usuario buscar el o los registros a los que les deseas realizar alguna modificación y el otro archivo se va a encargar precisamente de modificar los registros seleccionados con los nuevos datos capturados.

Al primer archivo prográmale el siguiente código y llámalo ***"modifica_panel.php"***:

```
1 <html>
2 <head>
3 <title>:: PANEL PARA MODIFICAR</title>
4 </head>
6 <body>
7 <h2> :: PANEL PARA MODIFICAR :: </h2>
8 <strong>Para modificar primero debes consultar los registros, así que en la siguiente caja
9 de texto puedes anotar el nombre del o los producto(s) o cajero(s) que deseas modificar.
10 El comodín que soporta es el simbolo->"%"</strong>
11 <!-- PHP_SELF es para recargar esta misma página cuando le den click
12 en el elemento tipo submit, enviando los datos del formulario por post -->
13 <form name="modifica" method="post" action="<?php echo $PHP_SELF; ?>">
14 Nombre:<input type="text" name="nombre" />
15 Tabla a Consultar:<select name="tabla">
16 <option value="productos">BUSCAR PRODUCTO</option>
17 <option value="cajeros">BUSCAR CAJEROS</option>
18 </select>
19 <input type="submit" name="consulta" value="Enviar">
20 </form>
21
22 <?php
23 include ("funciones/funciones_mysql.php");
24 extract($_POST);
25 /*preguntamos si existe la instancia "consulta", es
26 el tipo submit del formulario y se crea cuando se envia el formulario
27 es decir si existe es porque ya se envió la consulta */
28 if (isset($consulta)){
29 //llamar la función que conecta con el motor MySQL
30 conectar();
31 //diseñas la sentencia SQL
32 $sql = "SELECT * FROM $tabla WHERE nombre LIKE '$nombre'";
33 //ejecutas la consulta
34 $result = mysql_query($sql);
```

```
35 /*imprime la tabla con los resultados y con la opcion de
36 seleccionar para modificar*/
37 echo "<form name='envia_modifica' method='post' action='modifica.php'>
38     <table>";
39 //genera el encabezado de la tabla con los resultados
40 echo $tabla=="cajeros" ? "<tr><td>ID</td><td>NOMBRE</td><td>PASSWORD</td>
41 <td>PERMISOS</td><td>SELECCIONA</td></tr>" :
42 "<tr><td>ID</td><td>NOMBRE</td><td>CODIGO</td><td>TIPO</td><td>GENERO</td>
43 <td>PRECIO</td><td>EXISTENCIA</td><td>SELECCIONA</td></tr>";
44 //sacas los valores del arreglo
45 while ($campo = mysql_fetch_array($result)){
46     if ($tabla == 'cajeros'){
47         echo "<tr><td>$campo[id]</td><td><input type='text' name='nombre.$campo[id]'
48 value='$campo[nombre]' /></td><td><input type='text' name='pass.$campo[id]'
49 value='$campo[pass]' maxlength='5' /></td><td><input type='text'
50 name='permisos.$campo[id]' value='$campo[permisos]' /></td><td><input
51 type='checkbox' name='cajeros.$campo[id]' value='$campo[id]'></td></tr>";
52     }else{
53         echo "<tr><td>$campo[id]</td><td><input type='text' name='nombre.$campo[id]'
54 value='$campo[nombre]' /></td><td><input type='text' name='codigo.$campo[id]'
55 value='$campo[codigo]' /></td><td><input type='text' name='tipo.$campo[id]'
56 value='$campo[tipo]' /></td><td><input type='text' name='genero.$campo[id]'
57 value='$campo[genero]' /></td><td><input type='text' name='precio.$campo[id]'
58 value='$campo[precio]' /></td><td><input type='text' name='existencia.$campo[id]'
59 value='$campo[existencia]' /></td><td><input type='checkbox'
60 name='productos.$campo[id]' value='$campo[id]'></td></tr>";
61     }//fin de else
62 }//fin del while
63 echo "</table>
64 <input type='hidden' name='tabla' value='$tabla'><input type='submit' name='envia'
65 value='Modifica' /></form>";
66 }//fin del isset consulta ?>
67 </body>
68 </html>
```

Este código tiene la misma finalidad que el utilizado para el panel de eliminar, de echo si pones especial atención en el código notarás que son muy similares y la diferencia esta en los títulos y en la información que se muestra en campos de texto editables para poder anotar la información que se quiere modificar, además al ser un elemento **input** el formulario entiende que lo tiene que enviar al presionar el botón **submit**, esto lo puedes notar de la línea 47 a la 60. En el atributo **name** de los elementos **input** tienen el nombre de la variable y además separado con un punto el **id** correspondiente al registro que se muestra en ese momento, por ejemplo: "si al buscar un cajero resulta que este tiene 7 en el id, entonces la línea del input quedaría así (de forma automática): `<input type='text' name='nombre.7' value='Nombre del Cajero' />`", esto para poder identificar los datos seleccionados y así no haya variables repetidas.

Recuerda programarlo, puedes hacer “copiar y pegar” pero los resultados en el aprendizaje no son los mismos. Si solo quieres cumplir con alguna tarea o proyecto hazlo y entrega un proyecto que ni tu entenderás, pero si realmente quieres aprender pásalo manualmente y te darás cuenta de muchos detalles lo que provocará en ti un aprendizaje.

Nota: De la línea 40 a la 43 es para imprimir el encabezado de la tabla resultado y este depende si se seleccionó “cajeros” o “productos”, entonces se hace uso del if (yo le llamo de la forma elegante) con la siguiente estructura: **condición ? valor si verdadero : valor si falso ;**

Recuerda que siempre puedes mejorar la comprensión de estos códigos leyendo los artículos que están en: <http://AprendeAprogramarEnPHPya.com/blog>.

Le toca el turno al código encargado de realizar las modificaciones de los registros seleccionados de la tabla de resultados del formulario anterior, déjame explicarte con las siguientes imágenes. La imagen 6.1 es la primer pantalla que te debe mostrar el código de **modifica_panel.php**.



Imagen 6.1 – Pantalla que muestra antes de realizar una búsqueda en el archivo **modifica_panel.php**

De la línea 28 a la 66 en adelante no se ejecuta hasta que en esta pantalla se realiza una consulta, es decir cuando presiones el botón **enviar** que es de tipo **submit** se crea una nueva instancia llamada **consulta** (puedes verificarlo en la línea 19). Entonces en la línea 28 con la función del **isset()** pregunta si existe esa instancia por lo tanto la primera vez que carga la página obviamente no existe la instancia **consulta** por lo tanto las líneas de código que están entre la 28 y 66 no se ejecutan.

Para poder realizar la búsqueda tienes que anotar en el campo “*nombre*” el registro que deseas modificar, déjame explicarte con un ejemplo: “*voy a poner % que significa que va a*

mostrar todos los registros de la tabla **productos**” el resultado de esta búsqueda la puedes ver en la imagen 6.2.

:: PANEL PARA MODIFICAR ::

Para modificar primero debes consultar los registros, así que en la siguiente caja de texto puedes anotar el nombre del o los producto(s) o cajero(s) que deseas modificar soporta es el símbolo->"%"

Nombre: Tabla a Consultar: BUSCAR PRODUCTO

ID	NOMBRE	CODIGO	TIPO	GENERO	PRECIO	EXISTENCIA	SELECCIONA
5	<input type="text" value="batuka"/>	<input type="text" value="bb87"/>	<input type="text" value="pelicula"/>	<input type="text" value="adiestrami"/>	<input type="text" value="5"/>	<input type="text" value="83"/>	<input type="checkbox"/>
2	<input type="text" value="Hatty Potter E Final"/>	<input type="text" value="HPEPE01223"/>	<input type="text" value="Pelicula"/>	<input type="text" value="Suspenso 1"/>	<input type="text" value="2"/>	<input type="text" value="23"/>	<input type="checkbox"/>
4	<input type="text" value="El final 2 Harry Pot"/>	<input type="text" value="HRRP02"/>	<input type="text" value="Pelicua"/>	<input type="text" value="suspenso"/>	<input type="text" value="235"/>	<input type="text" value="345"/>	<input type="checkbox"/>

Imagen 6.2 – Es el resultado de buscar todos los registros (%) en la tabla productos

Aclaro los siguientes puntos, **primero** solo he insertado 3 registros en la tabla productos (para fines prácticos de este ejemplo) y **segundo** se le ha agregado a cada uno de los resultados un **checkbox** para permitir la selección de uno o de varios registros a modificar, es decir si deseas modificar el nombre al registro uno que dice **batuka por Batukada** debes anotar primero el nuevo nombre, luego seleccionar el **checkbox** correspondiente y darle click en modifica. Este te enviara a una página donde te muestra el número de registros afectados con éxito y un link para volver a la página de **modifica_panel.php**.

Creo con eso queda claro el funcionamiento del código encargado de mostrar el panel de control para modificaciones, ahora le toca el turno al código encargado de realizar la modificación directamente en el registro y que esta modificación quede almacenada en la base de datos. Siguiendo la misma estructura de este libro digital primero te muestro el código php que debes de anotar en un archivo nuevo al que vas a llamar **“modifica.php”**.

```
1 <?php
2 include ("funciones/funciones_mysql.php");
3 conectar();
4 extract($_POST);
5
6 $cont=0;
7 foreach ($_POST as $campo=>$valor) {
8     /* Un dato importante es que el formulario del código
9     anterior manda el nombre del campo checkbox que este
10    seleccionado, por lo tanto si existe un campo llamado
11    producto o cajero, sabes que se ha seleccionado minimo
12    un registro y sobre ese tienes que trabajar*/
```

```
13 if (strstr ($campo,"producto") || strstr($campo,"cajero")){
14 /* La función explode divide la cadena en un arreglo con
15 un numero de elementos de acuerdo al separador indicado,
16 en este caso el separador es el guión bajo (_) y solo
17 hay uno por lo tanto lo divide en un arreglo de dos */
18     $id = explode ("_", $campo);
19
20 /* Esta parte es muy padre y un poquito compleja de explicar
21 así que dejo la explicación completa en el e-book */
22     $nombre = "nombre_". $id[1];
23     $pass = "pass_". $id[1];
24     $permisos = "permisos_". $id[1];
25     $codigo = "codigo_". $id[1];
26     $tipo = "tipo_". $id[1];
27     $genero = "genero_". $id[1];
28     $precio = "precio_". $id[1];
29     $existencia = "existencia_". $id[1];
30 /*-----*/
31
32 /* Aqui aprovechas el uso del if elegante(como le llamo yo o
33 corto como se conoce) para identificar cual tabla se va
34 a afectar y obviamente los campos que le corresponden*/
35     $tabla == "cajeros" ? $sql = "UPDATE $tabla SET nombre='".$.$nombre."',
36         pass='".$.$pass."',permisos='".$.$permisos.'" WHERE id=$id[1]" :
37     $sql = "UPDATE $tabla SET nombre='".$.$nombre."',
38     codigo='".$.$codigo."', tipo='".$.$tipo."',genero='".$.$genero."',precio=$precio_ $id[1],
39     existencia='".$.$existencia.'" WHERE id=$id[1]";
40 //Ejecutamos la sentencia diseñada
41     $result = mysql_query($sql)
42     or die ("ERROR EN LA MODIFICACION ".$cont.mysql_error());
43 //si llega hasta aqui, es porque todo salio bien
44 //e incrementa uno a la variable contador
45     $cont++;
46 }
47 }
48 echo "<center>$cont REGISTRO(S) MODIFICADOS CON EXITO <a
href='modifica_panel.php' > Volver</a></center>";
49 ?>
```

Si te has dado cuenta ya le venido poniendo a estos códigos líneas de programación avanzada, lo hago para compartir contigo “*detalles*” especiales de programación que a mi llevo mucho tiempo de investigación y varias líneas de código antes de ver como se podían utilizar para ahorrar un montón de líneas de programación. Te explico el funcionamiento de este código: de la línea 1 a la 4 esta explicado en capítulos anteriores y son fácil de comprender, de cualquier forma si no te ha quedado claro con este e-book que ya casi terminas de leer sabes que cuentas con la página <http://AprendeAProgramarEnPHPya.com/blog>.

La variable **\$cont** que esta declarada en la línea 6 es para ir contando el numero de registros afectados en la operación. En la siguiente línea la 7, se usa el **foreach** para obtener el campo y valor que se esta enviando por el método **post** del formulario anterior y así poder interactuar con esos valores. El uso de la función **explode** en la línea 18 es para obtener el **id** del registro con el que se este interactuando en ese momento del ciclo (la funcionalidad de **explode** se describe en los comentarios del código).

Ojo con lo que se esta haciendo en las líneas de código que están de la 20 a la 29, se le esta asignando a una variable el nombre de otra variable, si leíste bien **el nombre de otra variable** no el valor, para luego poder utilizar el valor de la variable con doble signo(**\$\$**).

Para que quede mas claro te explico con un ejemplo corto: *“Si escribes el siguiente código en un archivo nuevo:*

```
<?php $nombre3=8;
      $var = "nombre"."3";
      echo $$var;
?>
```

*Al abrirlo en el explorador debe imprimir 8, es decir, el valor de la primer variable. Si pusieras **echo \$var** imprimiría **nombre3**.”*

Porque es necesario hacerlo así, recuerdas que al crear el formulario con el resultado de la consulta de nombre a cada instancia se le agrego el número **id** al nombre de la misma para evitar que se duplicarán las variables y evitar con ello confusión en los datos enviados por el formulario, entonces al leer esos datos en este formulario debes de cambiarle a esa base del nombre el numero de **id** para asegurarnos que se leen todas las variables.

Es un poco complejo esta parte del código pero estoy seguro que con la explicación lo entendiste y en caso contrario ya sabes que cuentas con la página <http://AprendeAprogramarEnPHPya.com/blog> para que puedas reforzar estos nuevos contenidos para ti, recuerda que mi mayor interés es que aprendas a programar en php lo mas pronto que puedas aunque el profesionalismo lo logras solo con la experiencia y mi objetivo es que logres programar en php profesionalmente lo mas pronto que puedas y para ello se necesitan bases fuertes.

Nota: De la línea 32 a la 39 es para crear la sentencia SQL encargada de llevar acabo la actualización dependiendo si se seleccionó “cajeros” o “productos”, entonces se hace uso del if (yo le llamo de la forma elegante) con la siguiente estructura: **condición ? valor si verdadero : valor si falso ;**

Recuerda que siempre puedes mejorar la comprensión de estos códigos leyendo los artículos que están en: <http://AprendeAprogramarEnPHPya.com/blog>.

Dentro de la sentencia SQL encargada de actualizar la información del registro puedes ver y comprobar el uso del doble **\$\$**, están entre las líneas 32 y 39. Y para terminar la explicación de este código solo me falta explicar la línea 45 que es donde se aumenta en uno la variable **\$cont** y llega hasta instrucción siempre y cuando no se haya producido algún error en la ejecución de las sentencias SQL.

Hasta este punto y si has realizado cada una de los códigos propuestos *¡¡¡ FELICITATE !!!* porque has terminado de programar las cuatro tareas básicas que debe tener todas las bases de datos, ahora solo faltara programar lo necesario para realizar una venta.

Si no sabes como probar o escribir tus códigos puedes ver el curso básico de 3 videos tutoriales que te regalo en: <http://AprendeAprogramarEnPHPya.com/blog>.

Capítulo 7: Consulta de dos o mas tablas. Ventas y Reportes

En realidad los principales motivos de este capítulo es mostrarte primero **como puedes realizar una consulta a más de una tabla** ya que es esta una de las preguntas que mas comúnmente me hacen y el otro motivo es para demostrarte como a partir de las **4 tareas básicas se pueden realizar las demás funciones** que quieras agregar a tu base de datos.

Este es el último capítulo así que no te detengas ahora, para que termines no tu primer script si no tu primer proyecto en php. **Para esta parte del proyecto vas a necesitar programar 5 archivos mas, en realidad se puede hacer en menos archivos pero he tomado la decisión de hacerlo así para poder explicar mejor y agregarles algunos tips que usan los profesionales y así puedas avanzarle mas en tu proceso de aprendizaje de este fabuloso lenguaje de programación web.**

El proceso de navegación para poder **realizar una venta** o **sacar un informe de ventas por producto vendido** que se va a programar en este ejemplo de desarrollo de proyecto es el siguiente: *“el usuario deberá identificarse con el nombre y contraseña que se dio de alta en la tabla cajeros, una vez validada la información abrirá la página del menú donde permite seleccionar **la venta** o bien **el reporte**; si él selecciona venta, abre la página donde permite en primer instancia seleccionar el producto por nombre o por código (se extraen de la tabla productos) y poner la cantidad de productos a vender, al darle click en vender muestra en la misma página el listado de los productos por nombre, la cantidad, el precio y el total, así como el total de todos los productos.*

Si selecciona reporte te manda a una página donde te permite escoger el nombre del producto del que deseas obtener su reporte de ventas.”

Espero haber sido claro en la explicación y si existiera alguna duda veamos si te la resuelvo en el transcurso de la explicación de cada código. Empieza pues con el primer archivo al que vas a llamar **“ventas_session.html”** y el código a utilizar en él es el siguiente:

```
1 <html>
2 <head>
3 <title>:: VENTAS ::</title>
4 </head>
5
6 <body>
7 <h3>INICIA SESION PARA PODER VENDER...</h3>
8 <form name="ventas" method="post" action="verifica.php">
9 USUARIO:<input type="text" name="user" height="15" /><br />
10 CONTRASEÑA:<input type="password" name="pass" height="15" maxlength="5"
/><br />
11 <input type="submit" name="enviar" value="Entrar" />
12 </form>
13 </body>
14 </html>
```

Este código es muy simple y solo es para pedirle al usuario que se identifique con su nombre y su contraseña, como puedes ver en la línea 8 estos dos datos serán enviados por el formulario al archivo que enseguida vas a programar y que se llama **“verifica.php”**, otro detalle antes de que se me olvide, en la línea 10 se utiliza el **type (tipo) password** para que no muestre lo digitado por el usuario.

Para el archivo de **verifica.php** este es el código que le corresponde:

```
1 <?php
2 include ("funciones/funciones_mysql.php");
3 conectar();
4 extract($_POST);
5
6 $sql = "SELECT * FROM cajeros WHERE nombre='$user' AND pass='$pass'";
7
8 $result = mysql_query($sql)
9 or die ("ERROR AL BUSCAR USUARIO EN LA BD ".mysql_error());
10 /* Aqui el resultado de la consulta se pasa
11 a un arreglo asociativo donde la base es
12 la variable campo */
13 $campo = mysql_fetch_assoc($result);
14 /* Preguntamos si el indice campo['nombre']
15 trae algo, en caso de afirmativo significa
16 que el usuario con esa contraseña si esta
17 registrado por lo tanto permite la entrada*/
18 if (!empty($campo['nombre']))
19 //Para abrir la pagina con el panel de ventas
20 //y reportes haces uso del javascript, y gracias
21 //al metodo GET mandamos el id del usuario
22     echo "<script language='javascript'>
window.open('ventas_panel.php?id=$campo[id]','_self','');</script>";
23 else
24 //En caso que el usuario y/o la contraseña no
25 //se haya encontrado se muestra el siguiente
26 //mensaje
27     echo "<center>ERROR EN EL NOMBRE DE USUARIO O CONTRASEÑA<a
href='ventas_sesion.php'>INTENTAR DE NUEVO</a></center>";
28 ?>
```

Este código ya esta un poco mas complicado que el anterior, de la línea 1 a la 13 pienso que ya esta claro su funcionamiento ya que se repite casi en todos los archivos donde se ejecuta alguna tarea en la base de datos, además que he comentado la parte que pudiera ser nueva en este proyecto dentro del código, recuerda que siempre cuentas con la página <http://AprendeAProgramarEnPHPya.com/blog> para reforzar tus aprendizajes con los artículos que encuentras en él. Algo nuevo que se incluye en este archivo mas especifico en la línea 22 es la **inclusión de código javascript** dentro de php y se usa en este caso para abrir de forma

automática la página “**ventas_panel.php**” y si te preguntas que significa “**?id=\$campo[id]**” entonces deja te contesto tu duda, vamos enviar por medio de la línea de dirección un dato aprovechándonos del método **GET**. Haber deja explico mas despacio este asunto, hasta ahorita hemos enviado datos de un formulario a otro archivo (e inclusive al mismo) utilizando el método **POST** esto es oculto al navegador de internet, pero en esta ocasión y aprovechando para enseñártelo se esta usando el otro método el **GET**, este envía los datos por medio de la barra de direcciones y por supuesto que esto NO ES SEGURO y menos cuando se trata del inicio de sesión pero insisto, lo hago para mostrarte algo diferente ya verás el uso del **id** en el siguiente archivo que llamas “**ventas_panel.php**” ¿te acuerdas donde lo viste?, si es ese el que abre código javascript.

El código del archivo es el siguiente:

```
1 <html>
2 <head>
3 <title>:: PANEL DE VENTAS ::</title>
4 </head>
5
6 <body>
7 <a href=<?php echo "ventas.php?id=".$_GET['id']; ?>>VENTAS</a><br />
8 <a href=<?php echo "reportes.php?id=".$_GET['id']; ?>>REPORTE DE VENTA POR
  PRODUCTO</a>
9 </body>
10 </html>
```

Este código es muy simple y sirve para redireccionarte a las ventas o a los reportes mandando el **id** del usuario identificado. En la línea 7 y 8 puedes ver que hacemos uso del **\$_GET** para acceder al valor mandado por la página anterior en ese mismo método y para que ese dato siga vigente en la siguiente página lo mandamos nuevamente por la barra de direcciones.

Es hora del script que va a ejecutar una venta y el código responsable de ello está en el archivo que se llama **ventas.php** y su código es el siguiente:

```
1 <?php
2 include ("funciones/funciones_mysql.php");
3 conectar();
4 extract($_POST);
5 ?>
6 <html>
7 <head>
8 <title>:: VENTAS ::</title>
9 </head>
10
```

```
11 <body>
12 <form name="ventas" method="post" action="<?php echo $PHP_SELF; ?>">
13 Buscar Por:<br />
14 Nombre:
15 <select name="prod_nombre">
16 <?php $sql = "SELECT * FROM productos";
17     $result = mysql_query($sql) or die ("ERROR AL CARGAR PRODUCTOS
18     ".mysql_error());
19     while ($campo = mysql_fetch_array($result))
20         echo "<option value='$campo[codigo]'$>$campo[nombre]</option>";
21 ?>
22 </select>
23 C&acute;digo:
24 <select name="prod_codigo">
25 <?php $sql = "SELECT * FROM productos";
26     $result = mysql_query($sql) or die ("ERROR AL CARGAR CODIGOS ".mysql_error());
27     while ($campo = mysql_fetch_array($result))
28         echo "<option value='$campo[codigo]'$>$campo[codigo]</option>";
29 ?>
30 </select>
31 Cantidad:<input name="cantidad" type="text" size="2" /><br>
32 <input type="hidden" name="id_cajero" value="<?php echo $_GET['id']; ?>">
33 <input type="submit" name="agregar" value="Agregar" />
34 </form>
35 <?php
36     if (isset ($prod_codigo) || isset ($prod_nombre)){
37         $fecha = date ("d/m/y h:m:s");
38         !empty($prod_nombre) ? $codigo = $prod_nombre : $codigo = $prod_codigo;
39         $sql = "INSERT INTO ventas VALUES
40         (null,'$fecha','$codigo',$cantidad,$id_cajero)";
41         mysql_query ($sql) or die ("ERROR AL REGISTRAR VENTA ".mysql_error());
42         $sql = "SELECT productos.precio,productos.nombre,ventas.* FROM
43         productos,ventas WHERE ventas.id_cajeros = $id_cajero
44         AND productos.codigo = ventas.codigo_producto";
45         $result = mysql_query($sql) or die ("ERROR AL IMPRIMIR VENTA ACTUAL
46         ".mysql_error());
47         echo
48         "<table><tr><td>Producto</td><td>Cantidad</td><td>Precio</td><td>Total</td>";
49         $gt = 0;
50         while($campo = mysql_fetch_array($result)){
51             echo
52             "<tr><td>$campo[nombre]</td><td>$cantidad</td><td>$campo[precio]</td><td>".$camp
53             o['precio']*$cantidad."</td></tr>";
```

```
48          $gt = $gt + ($campo['precio']*$cantidad);
49      }
50      echo "</table><br /><center><strong>TOTAL:
<u>$gt</u></strong></center><br>
          <a href='ventas_panel.php?id=$id_cajero'>Realizar Otra
Venta</a>";
51  }
51 ?>
52 </body>
53 </html>
```

Si has seguido este tutorial al pie de la letra y realizando los códigos por ti mismo ya debes de entender la mayor parte de este código, **ya que lo único nuevo de este script es la consulta a la tabla productos y a la tabla de ventas en una sola sentencia sql**. Permíteme explicarte esto con pausa.

Así que primero veamos la sentencia en cuestión:

```
SELECT productos.precio, productos.nombre, ventas.* FROM productos,ventas WHERE
ventas.id_cajeros = $id_cajero AND productos.codigo = ventas.codigo_producto
```

Esta sentencia se encuentra en la línea 42 del programa. Lo que se esta haciendo en esta consulta es obtener de la tabla productos el precio y el nombre, así como de la tabla ventas todos sus campos (eso es lo que indica el *) de los registros relacionados en ambas tablas por el campo **codigo_producto**(tabla ventas) **y código** (tabla productos), donde las ventas tenga el mismo **id_cajeros**.

Sé que esta medio confusa la explicación por eso pongo esta otra explicación para ver cual queda mas clara: *"la consulta es para obtener el nombre y el precio de los productos que aparecer en la venta que ha realizado el usuario identificado y marcado por su id"*.

Si aún se te dificulta entender esta consulta lo mejor es que realices tu mismo el ejemplo y veas que resultado tiene este código y esa será la mejor explicación que puedes tener de este código.

Recuerda que puedes buscar mas información sobre este tema en el blog:
<http://AprendeAProgramarEnPHPya.com/blog>.

Ahora es turno de revisar el código que se va a encargar de imprimir los reportes y que al igual que el de las ventas uno de los principales objetivos es mostrarte como se utiliza la consulta a dos tablas en una misma sentencia de SQL.

El código en cuestión esta en el archivo llamado **reportes.php**:

```
1 <?php
2 include ("funciones/funciones_mysql.php");
3 conectar();
4
5 echo <<<HTML
6     <html>
7     <head>
8     <title> :: REPORTE DE VENTA POR PRODUCTO :: </title>
9     </head>
10
11     <body>
12     <form name='producto' method='post' action='$PHP_SELF' >
13     Selecciona el producto:
14     <select name='producto'>
15 HTML;
16
17 $sql = "SELECT nombre,codigo FROM productos ORDER BY nombre ASC";
18 $result = mysql_query($sql) or die ("ERROR AL CARGAR LOS PRODUCTOS
19 ".mysql_error());
20 while ($campo = mysql_fetch_array($result))
21     echo "<option value='$campo[codigo]'">$campo[nombre]</option>";
22 echo <<<HTML
23     </select>
24     <input type='submit' name='enviar' value='Ver Reporte' />
25     </form>
26 HTML;
27 extract($_POST);
28 if (isset($producto)){
29 $sql = "SELECT productos.nombre AS np,productos.precio,cajeros.nombre,ventas.fecha,
30     ventas.cantidad FROM productos,cajeros,ventas WHERE
31     ventas.codigo_producto='$producto' AND
32     productos.codigo = ventas.codigo_producto AND cajeros.id = ventas.id_cajeros ";
33 $result = mysql_query($sql) or die ("ERROR AL BUSCAR VENTAS ".mysql_error());
34 $gt=0;
35 while ($campo = mysql_fetch_array($result)){
36     echo $gt == 0 ? "<strong>PRODUCTO: <u>$campo[np]</u></strong>"
37     <table
38     border='1'><tr><td>CAJERO</td><td>FECHA</td><td>TOTAL</td></tr>":0;
39     echo "<tr><td>$campo[nombre]</td><td>$campo[fecha]</td><td>".
40     $campo['precio']*$campo['cantidad']. "</td></tr>";
41     $gt = $gt + ($campo['precio']*$campo['cantidad']);
42 }
43 echo "</table><strong>TOTAL: <u>$gt</u></strong><br><a
44 href='ventas_panel.php?id=$_GET[id]'">VOLVER AL PANEL DE VENTAS</a>";
45 }
46 echo <<<HTML
```

```
41    </body>
42    </html>
43 HTML;
44
45 ?>
```

Al igual que el código anterior encargado de la venta, este código debe ser entendido si has realizado paso a paso este tutorial y lo mas destacado o diferente de este código es la consulta SQL que obtiene datos de tres tablas distintas. Esta es la sentencia SQL:

```
SELECT productos.nombre AS np, productos.precio, cajeros.nombre, ventas.fecha,
      ventas.cantidad
FROM productos,cajeros,ventas
WHERE ventas.codigo_producto='$producto'
      AND
      productos.codigo = ventas.codigo_producto
      AND
      cajeros.id = ventas.id_cajeros
```

Como puedes ver en la cláusula de **FROM** se esta aplicando el filtro de **SELECT** a tres tablas (productos, cajeros y ventas) donde las condiciones son código_producto de la tabla ventas sea igual al producto para el cual se está solicitando el reporte y que de la tabla productos obtenga los registros que se relacionan con el campo de código_producto de la tabla ventas y que la tabla cajeros donde el campo id este relacionado con la tabla de ventas por medio del campo id_cajeros. No entendí ...

Hay va otra explicación distinta haber si queda mas claro: “*se va a obtener de la tabla productos el nombre y el precio, de la tabla cajeros el nombre y de la tabla ventas la fecha y la cantidad, de la venta de un producto*”.

Espero se haya comprendido alguna de las dos explicaciones, y perdona que sea tan insistente en esto pero si quieres comprender mejor lo que acabo de hacer en esa consulta, pues lo mejor es que realices el ejemplo tu mismo para que veas el resultado de dicha consulta y así puedas comprender mejor lo que acabo de explicar, recuerda mi único objetivo de todo esto que hago para ti es que aprendas a programa en php de forma fácil y sencilla.

Ya solo falta que le pongas tu toque personal con un poco de maquetación, es decir una presentación agradable y con esto queda concluido este tutorial, recuerda como lo he venido manejando desde un principio, NO VOY A PONER los archivos para su descarga excepto para aquellos que ya adquirieron o que adquieran el curso (recibirán en su email un link para la descarga).

Espero de corazón que este material sea de mucha utilidad para ti, porque la verdad lo estaba realizando para venderlo en 17usd, pero no, creo que este conocimiento básico va a servir de mucho para aquellos (sobre todo estudiantes) que desean seriamente aprender a

programar en php en poco tiempo, así que si permites pedirte un favor es el siguiente: comparte este material con el mayor número de amigos posibles, pero solo para aquellos que sepan valorar el esfuerzo de un material didáctico realizado con el único objetivo de que aprendas a programar en php.

Recuerda este es mi blog: <http://AprendeAprogramarEnPHPya.com/blog>, hay estaré publicando artículos continuamente.

Soy Salvador López, mexicano, licenciado en informática y profesor de programación cualquier asunto, comentario y/o recomendación será bien recibida en el e-mail: SalvadorL@aprendeaprogramarenphpya.com.