

INSTITUTO TECNOLÓGICO SUPERIOR DE CHICONTEPEC.

INGENIERIA EN SISTEMAS COMPUTACIONALES.

ASIGNATURA:

Programación Web.

ALUMNO:

Flor Hernández Cruz.

DOCENTE:

Ingeniero Efrén Flores Cruz.

UNIDAD 3:

Programación del lado del cliente.

TRABAJO:

Reporte de Pagina Web Local

SEMESTRE:

8°

Contenido

INTRODUCCION	3
DESARROLLO	4
CONCLUSION.....	10

INTRODUCCION

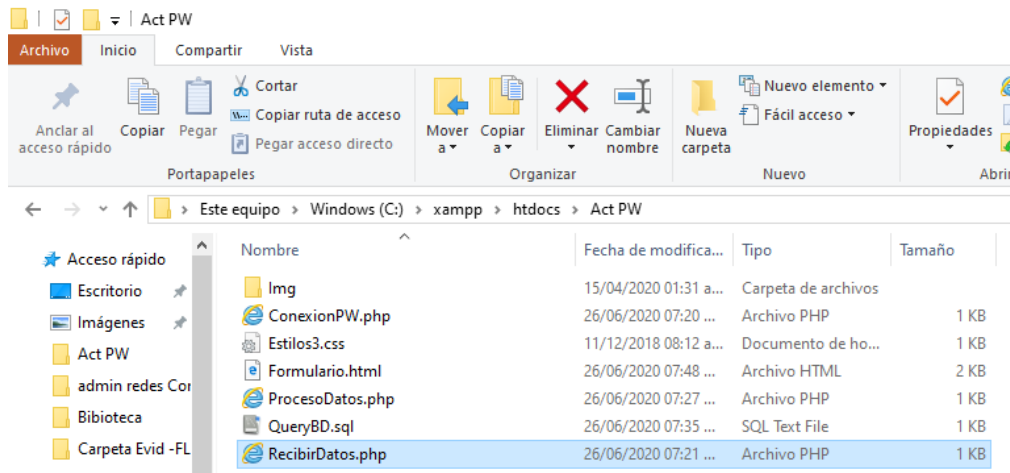
El presente trabajo es de la asignatura de programación web, muestra la creación de una página web de forma local, usando base de datos con ayuda del sistema gestor de base de datos MySQL.

Se muestran los archivos del index, conexión, procesodatos, recibir datos, la imagen que contiene, así como también un archivo de extensión sql, aquí esta guardada las instrucciones para la creación de la base de datos.

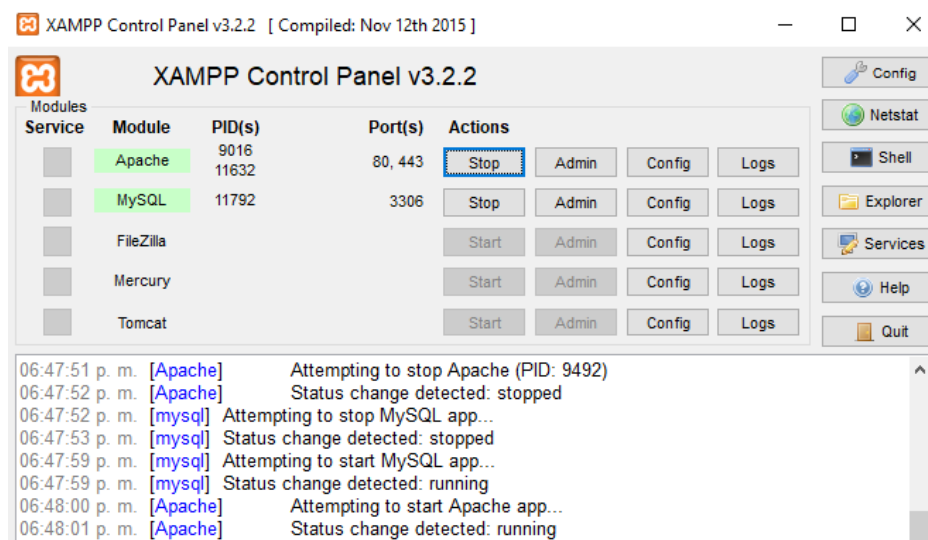
Los datos son almacenados en mi ordenador, se visualizan tipiendo la dirección IP local: 127.0.0.1 y se muestran todos los archivos que estén relacionados con PHP, programación web.

DESARROLLO

Los archivos para la creación de la página web deben ser almacenados en la siguiente ruta: C:\xampp\htdocs\Act PW esta ruta contiene los siguientes archivos: una carpeta de imágenes, archivos de conexión, formulario, proceso datos, un query de tipo sql, recibir datos.

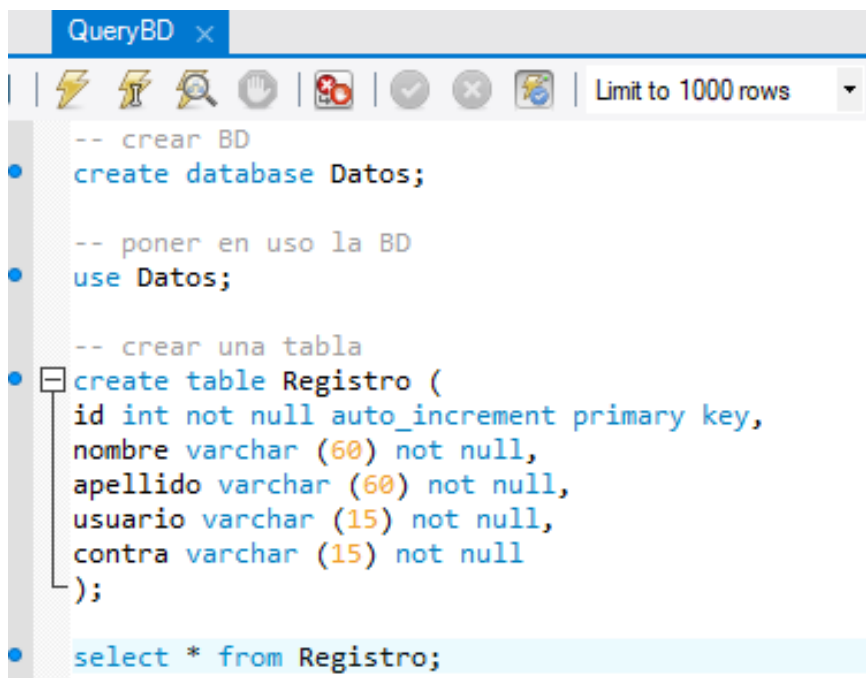


1. Se abre el panel de control de Xampp se inician los servidores de Apache, MySQL; cuando se inicia correctamente se muestra de color verde, de lo contrario se muestra el color amarillo y se envió un mensaje que no se ha establecido una conexión.



Creación de la base de datos

2. Se abre el sistema gestor de base de datos MySQL, se agregan las siguientes instrucciones. La base de datos se llama Datos, posteriormente se pone en uso con el comando use, la tabla se llama Registro y contiene los siguientes campos: el id, nombre, apellido, usuario, contraseña.



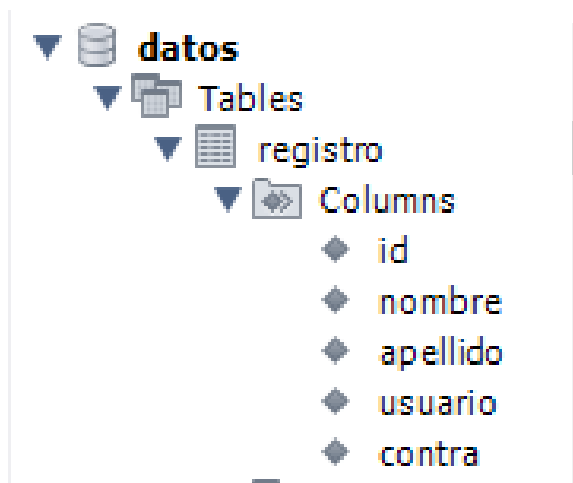
```
-- crear BD
create database Datos;

-- poner en uso la BD
use Datos;

-- crear una tabla
create table Registro (
  id int not null auto_increment primary key,
  nombre varchar (60) not null,
  apellido varchar (60) not null,
  usuario varchar (15) not null,
  contra varchar (15) not null
);

select * from Registro;
```

3. Se ejecutan las instrucciones y se verifica la base de datos, así como se muestra en la siguiente imagen.



Conexión de la base de datos

4. Se crea el archivo de ConexionPW.php asignando los parámetros de conexión: el tipo de servidor, el usuario, contraseña, nombre de la base de datos; conexión, posteriormente se verifica la conexión, si se ha realizado correctamente o si tiene una falla se envía un mensaje en el navegador.

```
RecibirDatos.php x ConexionPW.php x ProcesoDatos.php x QueryBD.sql x
1
2 <h3>creacion de una base de datos </h3>
3 <?php
4 // parametros de conexion con el gestor de base de datos
5 $servername="localhost";
6 $username="root";
7 $password="hernandez24";
8 $dbname="datos";
9
10 //Conexion
11 $conn = new mysqli($servername, $username, $password, $dbname);
12
13 //Verifica la conexion
14 if($conn->connect_error)
15 {
16     die("La conexion falló: " . $conn->connect_error);
17 }
18 else{
19     echo "La conexión se realizó satisfactoriamente";
20 }
21 //Se cierra la conexión
22 //$conn->close();
23 ?>
```

ProcesoDatos.php

5. Se crea el archivo de proceso datos. Se definen las variables estos deben coincidir con la base de datos, se agregan las etiquetas que hacen referencia a los campos.

```
ProcesoDatos.php x
1 <?php
2 //definen las variables
3 $Nombre = $Apellido = $Usuario = $Contra = " ";
4 //la cadena vacia se asegura de que las variables esten vacias
5 if ($_SERVER["REQUEST_METHOD"]=="POST"){
6     $Nombre = test_input(strtoupper($_POST["nombre"]));
7     $Apellido = test_input(strtoupper($_POST["apellido"]));
8     $Usuario = test_input(strtoupper($_POST["usuario"]));
9     $Contra = test_input(strtoupper($_POST["contra"]));
10 }
11 function test_input($data){
12     $data = trim($data);
13     $data = stripslashes($data);
14     $data = htmlspecialchars($data);
15     return $data;
16 }
17 ?>
```

Formulario de HTML

- Se crea el formulario con extensión de .HTML se crea el encabezado, se agregan los colores, el tipo de letra, el tamaño, las imágenes, posteriormente se agrega la acción de recibirDatos.php esto es para que al pulsar el botón de registrar se envíe los mensajes destinados en este archivo. Después sigue la creación de las etiquetas y las cajas de texto para los campos. Finalmente se agrega el botón de Registrar.

```

<!-- insertar titulo -->
<html>
<head>
</head>
<body bgcolor="#99CCFF">
<center><h1><font color="black" size="10" face="Times New Roman"> Formulario de Registro </font></h1>
 <br><br>
<font color="black">
<form action="RecibirDatos.php" method="POST">
<b>
<center> <label for="Nombre">Nombre <span><em></em></span></label>
<input type="text" name="nombre" placeholder="Escribe su nombre"> </center> <br>
<center> <label for="Apellido">Apellido <span><em></em></span></label>
<input type="text" name="apellido" placeholder="Escribe su apellido"> </center> <br>
<center> <label for="Usuario">Usuario <span><em></em></span></label>
<input type="text" name="usuario" placeholder="Escribe su usuario"> </center> <br>
<center> <label for="Contra">Password <span><em></em></span></label>
<input type="text" name="contra" placeholder="Escribe su Password"> </center> <br>
<center> <input class="form-btn" name="submit" type="submit" value="Registrar"/></center>
</b>
</form>
</body>
</html>

```

Recibir datos

- Se crea el archivo de Recibir datos.php en este archivo se agrega que se requiere el archivo de conexión, el de procesoDatos; después se agrega una instrucción SQL de inserción de datos, se vincula el query de sql; finalmente se accede a un ciclo de ir, es decir si los datos coinciden con los campos, se envía un mensaje de Registro exitoso, de lo contrario se envía un mensaje de error.

```








RecibirDatos.php x ProcesoDatos.php x
<?php
require("ConexionPW.php");
require("ProcesoDatos.php");
$sql = "INSERT Registro(Nombre, Apellido, Usuario, Contra)
VALUES ('".$Nombre."', '".$Apellido."', '".$Usuario."', '".$Contra."')";
if($conn-> query($sql) === TRUE)
{
    echo "Registro Exitoso";
}
else
{
    echo "Error " . $sql . "<br>" . $conn->connect_error;
}
// $conn->close();
?>

```

- Posteriormente se accede a los archivos de manera local host con la dirección IP 127.0.0.1 y el nombre de la carpeta donde están los archivos. Así como se muestra en la siguiente imagen.

← → ↻ ⓘ 127.0.0.1/Act%20PW/

Index of /Act PW

Name	Last modified	Size	Description
 Parent Directory		-	
 ConexionPW.php	2020-06-26 19:20	513	
 Estilos3.css	2018-12-11 08:12	827	
 Formulario.html	2020-04-15 00:51	1.1K	
 Img/	2020-04-15 01:31	-	
 ProcesoDatos.php	2020-04-18 14:02	550	
 Query_BD.sql	2020-06-26 19:11	425	
 RecibirDatos.php	2020-06-26 19:21	347	

Apache/2.4.25 (Win32) OpenSSL/1.0.2j PHP/7.1.6 Server at 127.0.0.1 Port 80

9. Se accede al archivo de formulario.html y se visualiza el diseño de la página web. Posteriormente se agregan los datos a las cajitas de texto y se da clic en el botón de registrar.

Formulario de Registro



Nombre

Apellido

Usuario

Password

10. Se envía el siguiente mensaje que indica que la conexión se realizó bien y que se registraron los datos correctamente.



creacion de una base de datos

La conexión se realizó satisfactoriamente Registro Exitoso

11. Se verifica en la base de datos de MySQL con el comando que aparece en la siguiente imagen: `select * from nombreDeLaTabla;`

En la siguiente imagen se muestran los datos registrados en el formulario.

15
16 • `select * from Registro;`

<

Result Grid | Filter Rows: | Edit: | Ex

	id	nombre	apellido	usuario	contra
	1	KARLA	HE	WE3	AMD12
	2	FLOR	HERNANDEZ	8N4	34D
	3	XOCHITL	MARQUEZ	FHDS	977HJKJ
	NULL	NULL	NULL	NULL	NULL

CONCLUSION

El trabajo de la creación de la página web usando base de datos me pareció un poco estresante debido a que no me salía correctamente la inserción de los datos, esto es debido a que no había escrito adecuadamente un campo en la base de datos; revise varias veces, hasta que volví a revisar línea por línea y detecte el error.

Esto es un claro ejemplo que por un error por más grande o pequeño que sea es complicado que se ejecuten los procesos.