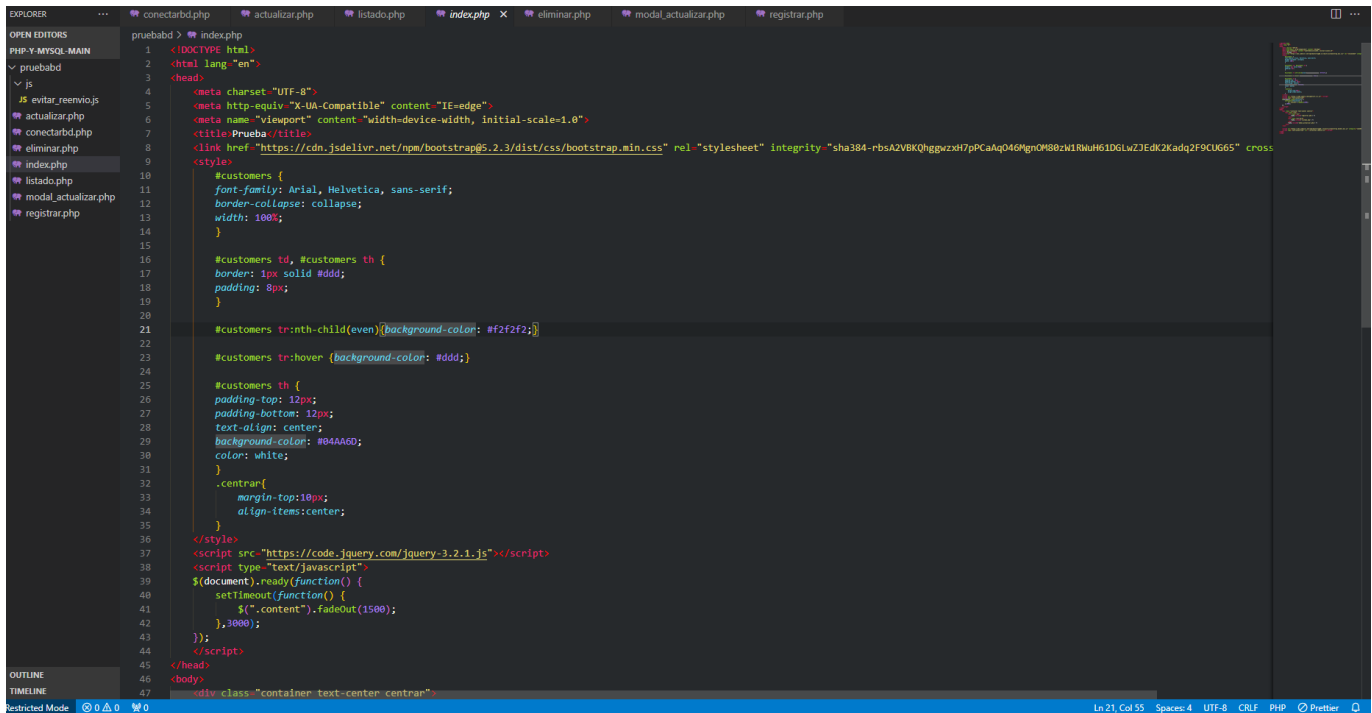


EXAMEN PROGRAMADOR



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Prueba</title>
8   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQhggwzzH7pCaAq046MgnOM80zW1RwuH61DGLwZ3EdK2Kadq2F9CU665" crosss
9   <style>
10     #customers {
11       font-family: Arial, Helvetica, sans-serif;
12       border-collapse: collapse;
13       width: 100%;
14     }
15
16     #customers td, #customers th {
17       border: 1px solid #ddd;
18       padding: 8px;
19     }
20
21     #customers tr:nth-child(even){background-color: #f2f2f2;}
22
23     #customers tr:hover {background-color: #ddd;}
24
25     #customers th {
26       padding-top: 12px;
27       padding-bottom: 12px;
28       text-align: center;
29       background-color: #04AA6D;
30       color: white;
31     }
32     .centrar{
33       margin-top:10px;
34       align-items:center;
35     }
36   </style>
37   <script src="https://code.jquery.com/jquery-3.2.1.js"></script>
38   <script type="text/javascript">
39     $(document).ready(function() {
40       setTimeout(function() {
41         $(".content").fadeOut(1500);
42       },3000);
43     });
44   </script>
45 </head>
46 <body>
47   <div class="container text-center centrar">
```

Index.php:

HTML: El código HTML define la estructura de la página. Se especifica el lenguaje del documento y se configuran las metaetiquetas para asegurar la compatibilidad con diferentes dispositivos y navegadores.

Bootstrap: Se incluye Bootstrap, un popular framework CSS que facilita el diseño responsivo y atractivo. En este caso, se usa Bootstrap 5.2.3 para el diseño de la página, con su hoja de estilo enlazada en el <head>.

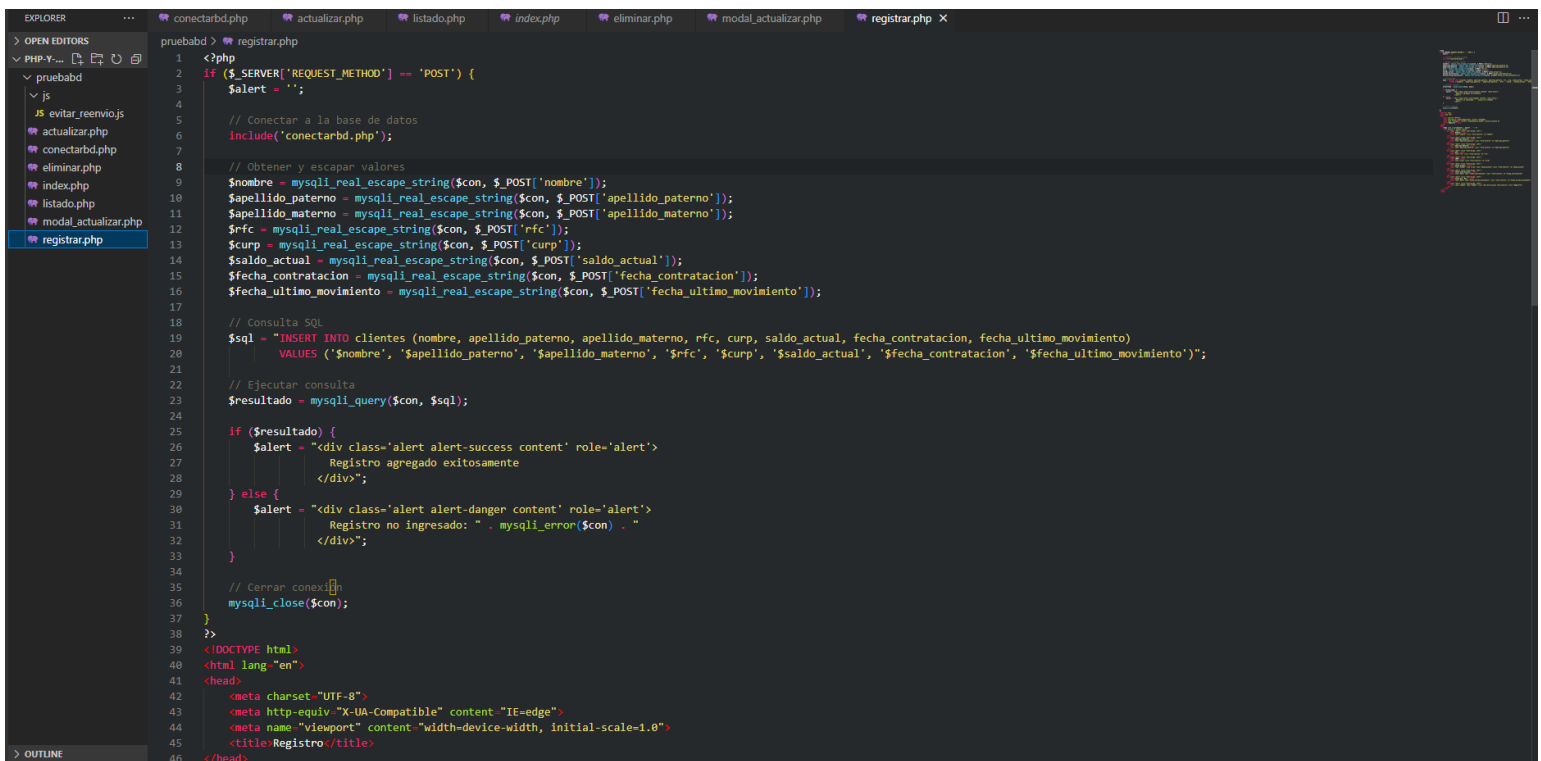
CSS: Se aplican estilos personalizados para la tabla con el ID customers, configurando el formato de celdas, bordes y colores de fondo para mejorar la presentación de datos tabulares. También se define una clase. centrar para centrar y alinear el contenido.

JavaScript y jQuery: Se utiliza jQuery para manejar eventos en la página. El script asegura que el contenido con la clase content se desvanecerá suavemente después de 3 segundos. jQuery se carga desde una CDN externa.

PHP: El código PHP se usa para incluir otros archivos PHP (registrar.php, listado.php, y modal_actualizar.php), lo que permite modularizar y organizar mejor el contenido de la página.

Bootstrap JavaScript: Se incluye el archivo JavaScript de Bootstrap al final del <body>, lo que permite utilizar componentes interactivos del framework, como modales y alertas.

JavaScript adicional: También se enlaza un archivo JavaScript personalizado (evitar_reenvio.js), que probablemente maneje la lógica para evitar el reenvío de formularios.



```
1 <?php
2 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
3     $alert = '';
4
5     // Conectar a la base de datos
6     include('conectarbd.php');
7
8     // Obtener y escapar valores
9     $nombre = mysqli_real_escape_string($con, $_POST['nombre']);
10    $apellido_paterno = mysqli_real_escape_string($con, $_POST['apellido_paterno']);
11    $apellido_materno = mysqli_real_escape_string($con, $_POST['apellido_materno']);
12    $rfc = mysqli_real_escape_string($con, $_POST['rfc']);
13    $curp = mysqli_real_escape_string($con, $_POST['curp']);
14    $saldo_actual = mysqli_real_escape_string($con, $_POST['saldo_actual']);
15    $fecha_contratacion = mysqli_real_escape_string($con, $_POST['fecha_contratacion']);
16    $fecha_ultimo_movimiento = mysqli_real_escape_string($con, $_POST['fecha_ultimo_movimiento']);
17
18    // Consulta SQL
19    $sql = "INSERT INTO clientes (nombre, apellido_paterno, apellido_materno, rfc, curp, saldo_actual, fecha_contratacion, fecha_ultimo_movimiento)
20    VALUES ('$nombre', '$apellido_paterno', '$apellido_materno', '$rfc', '$curp', '$saldo_actual', '$fecha_contratacion', '$fecha_ultimo_movimiento')";
21
22    // Ejecutar consulta
23    $resultado = mysqli_query($con, $sql);
24
25    if ($resultado) {
26        $alert = "<div class='alert alert-success content' role='alert'>
27        Registro agregado exitosamente
28        </div>";
29    } else {
30        $alert = "<div class='alert alert-danger content' role='alert'>
31        Registro no ingresado: " . mysqli_error($con) . "
32        </div>";
33    }
34
35    // Cerrar conexión
36    mysqli_close($con);
37 }
38 >
39 <!DOCTYPE html>
40 <html lang="en">
41 <head>
42     <meta charset="UTF-8">
43     <meta http-equiv="X-UA-Compatible" content="IE=edge">
44     <meta name="viewport" content="width=device-width, initial-scale=1.0">
45     <title>Registro</title>
46 </head>
```

Registrar.php:

Este código es un pequeño script PHP que se encarga de registrar información en una base de datos cuando un usuario envía un formulario. Primero, el código verifica si el formulario ha sido enviado utilizando el método POST. Si es así, realiza varias acciones:

Conexión a la base de datos: Incluye un archivo llamado conectarbd.php, que probablemente contiene la información necesaria para conectar con la base de datos.

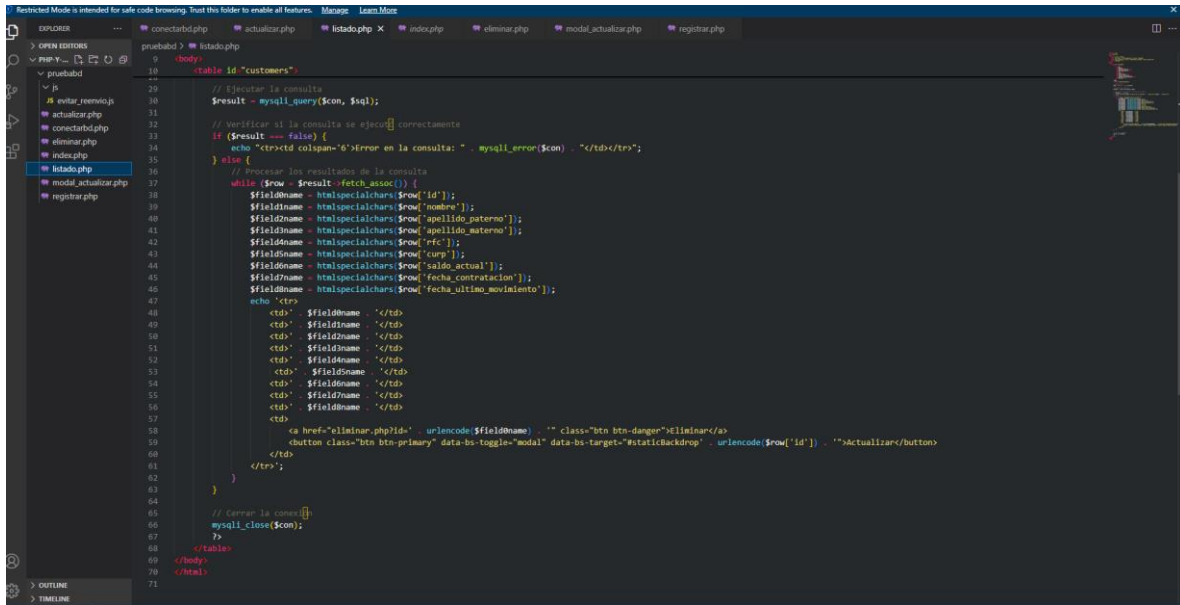
Procesamiento de datos: Recoge los datos enviados a través del formulario, como nombre, apellidos, RFC, CURP, saldo actual, fecha de contratación y fecha del último movimiento. Todos estos datos se "escapan" para evitar problemas de seguridad (como ataques de SQL injection) usando `mysqli_real_escape_string`.

Inserción de datos: Luego, construye una consulta SQL para insertar estos datos en la tabla clientes de la base de datos. Ejecuta esta consulta y verifica si se ha ejecutado correctamente.

Mensajes de alerta: Dependiendo del resultado de la consulta, muestra un mensaje de éxito o de error al usuario. Los mensajes se muestran usando alertas que se estilizan con clases de Bootstrap, aunque no se muestra el código de Bootstrap en este fragmento.

Cierre de conexión: Finalmente, cierra la conexión con la base de datos para liberar recursos.

El código también incluye un formulario HTML en la misma página. Este formulario permite al usuario ingresar los datos necesarios. Cuando el usuario envía el formulario, los datos se envían al mismo script PHP que se encarga de procesarlos y almacenarlos en la base de datos.



```
1 <body>
2     <table id="customers">
3
4         <!-- Ejecutar la consulta -->
5         $result = mysqli_query($con, $sql);
6
7         <!-- Verificar si la consulta se ejecutó correctamente -->
8         if ($result === false) {
9             <!-- Mostrar mensaje de error -->
10            echo "<tr><td colspan='6'>Error en la consulta: " . mysqli_error($con) . "</td></tr>";
11        } else {
12            <!-- Recorrer los resultados de la consulta -->
13            while ($row = mysqli_fetch_assoc()) {
14                <!-- Crear las etiquetas de la fila -->
15                $fieldName = htmlspecialchars($row["id"]);
16                $fieldName = htmlspecialchars($row["nombre"]);
17                $fieldName = htmlspecialchars($row["apellido_paterno"]);
18                $fieldName = htmlspecialchars($row["apellido_materno"]);
19                $fieldName = htmlspecialchars($row["rfc"]);
20                $fieldName = htmlspecialchars($row["curp"]);
21                $fieldName = htmlspecialchars($row["saldo_actual"]);
22                $fieldName = htmlspecialchars($row["fecha_contratacion"]);
23                $fieldName = htmlspecialchars($row["fecha_ultimo_movimiento"]);
24
25                <tr>
26                    <td> $fieldName </td>
27                    <td> $fieldName </td>
28                    <td> $fieldName </td>
29                    <td> $fieldName </td>
30                    <td> $fieldName </td>
31                    <td> $fieldName </td>
32
33                    <td>
34                        <a href="eliminar.php?id=" . urlencode($fieldName) . ">Eliminar</a>
35                        <button class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#staticBackdrop" . urlencode($row['id']) . ">Actualizar</button>
36                    </td>
37                </tr>
38            }
39
40            <!-- Cerrar la consulta -->
41            mysqli_close($con);
42        }
43    </table>
44 </body>
45 </html>
```

Listado.php:

Estructura HTML: La página está estructurada con HTML básico. En el <head>, se definen las meta etiquetas para la codificación de caracteres y la compatibilidad con dispositivos, y se establece el título de la página.

Tabla de Clientes: En el <body>, se crea una tabla con el ID customers. La tabla tiene encabezados para los diferentes campos de datos de los clientes, como nombre, apellidos, RFC, CURP, saldo actual, y fechas.

PHP para Datos Dinámicos:

Conexión a la Base de Datos: Incluye un archivo conectarbd.php que se encarga de conectar la página con la base de datos.

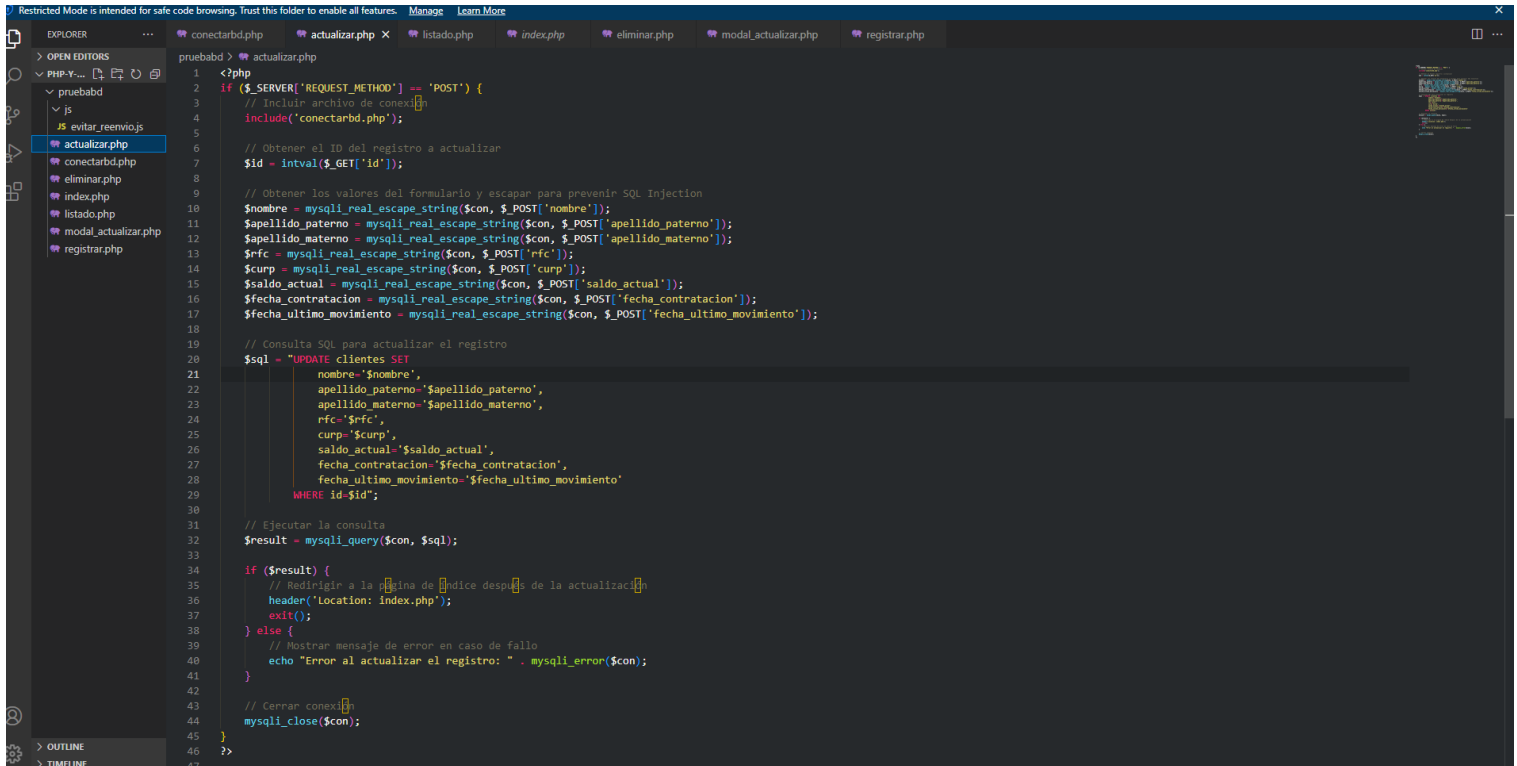
Consulta SQL: Se ejecuta una consulta SQL para obtener todos los registros de la tabla clientes.

Mostrar Datos: Si la consulta se ejecuta correctamente, se recorren los resultados y se crean filas en la tabla HTML para cada cliente. Cada campo se muestra en una celda de la tabla. Además, se muestran dos botones en cada fila:

Eliminar: Un enlace que lleva a eliminar.php para eliminar el cliente seleccionado.

Actualizar: Un botón que activa un modal (ventana emergente) para actualizar la información del cliente. El ID del cliente se pasa como parámetro.

Errores y Cierre de Conexión: Si hay un error en la consulta SQL, se muestra un mensaje en la tabla. Finalmente, se cierra la conexión con la base de datos para liberar recursos.



```
1 <?php
2 if ($_SERVER["REQUEST_METHOD"] == "POST") {
3     // Incluir archivo de conexión
4     include("conectarbd.php");
5
6     // Obtener el ID del registro a actualizar
7     $id = intval($_GET['id']);
8
9     // Obtener los valores del formulario y escapar para prevenir SQL Injection
10    $nombre = mysqli_real_escape_string($con, $_POST['nombre']);
11    $apellido_paterno = mysqli_real_escape_string($con, $_POST['apellido_paterno']);
12    $apellido_materno = mysqli_real_escape_string($con, $_POST['apellido_materno']);
13    $rfc = mysqli_real_escape_string($con, $_POST['rfc']);
14    $curp = mysqli_real_escape_string($con, $_POST['curp']);
15    $saldo_actual = mysqli_real_escape_string($con, $_POST['saldo_actual']);
16    $fecha_contratacion = mysqli_real_escape_string($con, $_POST['fecha_contratacion']);
17    $fecha_ultimo_movimiento = mysqli_real_escape_string($con, $_POST['fecha_ultimo_movimiento']);
18
19    // Consulta SQL para actualizar el registro
20    $sql = "UPDATE clientes SET
21        nombre='$nombre',
22        apellido_paterno='$apellido_paterno',
23        apellido_materno='$apellido_materno',
24        rfc='$rfc',
25        curp='$curp',
26        saldo_actual='$saldo_actual',
27        fecha_contratacion='$fecha_contratacion',
28        fecha_ultimo_movimiento='$fecha_ultimo_movimiento'
29        WHERE id=$id";
30
31    // Ejecutar la consulta
32    $result = mysqli_query($con, $sql);
33
34    if ($result) {
35        // Redirigir a la página de índice después de la actualización
36        header("Location: index.php");
37        exit();
38    } else {
39        // Mostrar mensaje de error en caso de fallo
40        echo "Error al actualizar el registro: " . mysqli_error($con);
41    }
42
43    // Cerrar conexión
44    mysqli_close($con);
45 }
46 >
```

Actualizar.php:

Verificación del Método de Solicitud: Primero, el script comprueba si el formulario ha sido enviado utilizando el método POST. Esto asegura que el código solo se ejecute cuando se envíen datos desde el formulario.

Conexión a la Base de Datos: Incluye un archivo llamado conectarbd.php, que probablemente contiene la lógica para conectar a la base de datos.

Obtención del ID y Datos del Formulario:

ID del Cliente: Obtiene el ID del cliente a actualizar desde la URL (\$_GET['id']) y lo convierte en un entero para evitar problemas de seguridad.

Datos del Formulario: Captura los datos enviados desde el formulario (nombre, apellidos, RFC, CURP, saldo actual, fechas) y los "escapa" para protegerse contra inyecciones SQL usando `mysqli_real_escape_string`.

Consulta SQL de Actualización: Crea una consulta SQL para actualizar el registro del cliente en la tabla clientes, estableciendo los nuevos valores en los campos correspondientes.

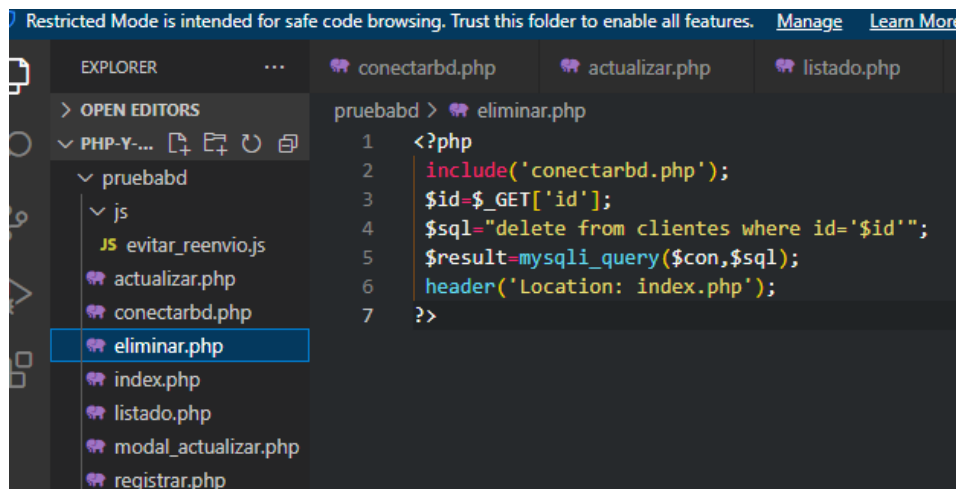
Ejecución y Resultado:

Ejecución: Ejecuta la consulta SQL con `mysqli_query`.

Redirección: Si la actualización es exitosa, redirige automáticamente al usuario a la página `index.php`, que probablemente muestra el listado de clientes actualizado.

Error: Si ocurre un error durante la actualización, muestra un mensaje de error con detalles.

Cierre de Conexión: Finalmente, cierra la conexión con la base de datos para liberar recursos.



```
1 <?php
2 include('conectarbd.php');
3 $id=$_GET['id'];
4 $sql="delete from clientes where id='$id'";
5 $result=mysqli_query($con,$sql);
6 header('Location: index.php');
7 ?>
```

Eliminar.php:

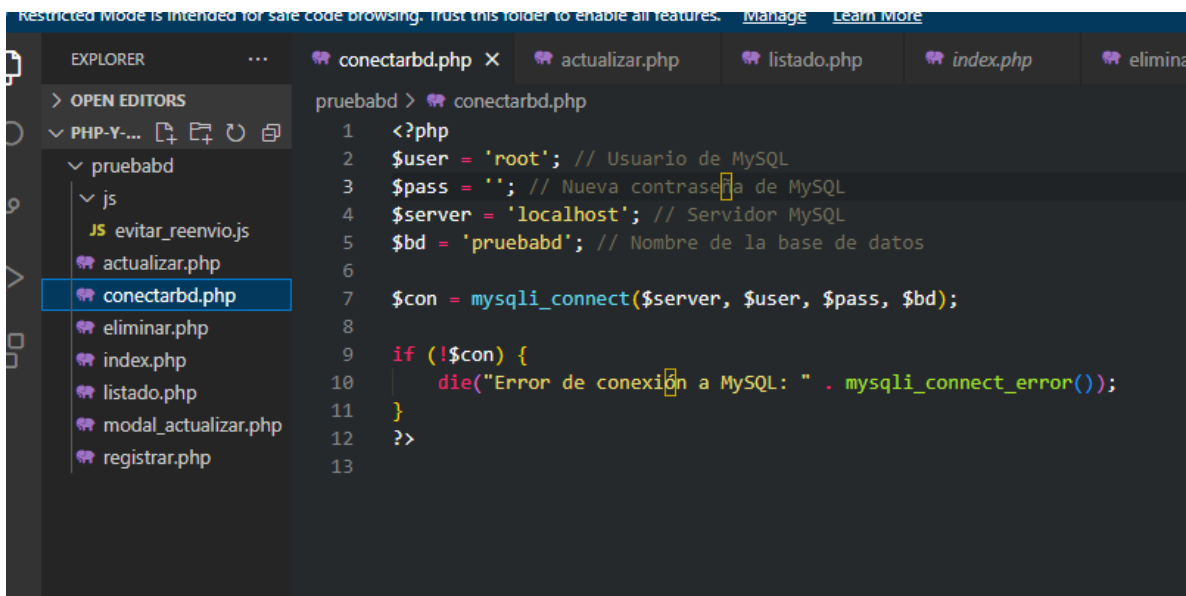
Conexión a la Base de Datos: Primero, incluye el archivo conectarbd.php, que contiene la lógica para conectar con la base de datos.

Obtener el ID: Luego, obtiene el ID del cliente que se va a eliminar desde la URL (\$_GET['id']). Este ID se usa para identificar el registro específico que se quiere borrar.

Consulta SQL para Eliminación: Crea una consulta SQL que elimina el registro de la tabla clientes donde el id coincide con el valor obtenido.

Ejecutar la Consulta: Ejecuta la consulta SQL usando mysqli_query. Esto realiza la eliminación en la base de datos.

Redirección: Después de ejecutar la consulta, redirige al usuario a index.php, que normalmente muestra el listado de clientes actualizado.



```
1 <?php
2 $user = 'root'; // Usuario de MySQL
3 $pass = ''; // Nueva contraseña de MySQL
4 $server = 'localhost'; // Servidor MySQL
5 $bd = 'pruebabd'; // Nombre de la base de datos
6
7 $con = mysqli_connect($server, $user, $pass, $bd);
8
9 if (!$con) {
10     die("Error de conexión a MySQL: " . mysqli_connect_error());
11 }
12 ?>
13
```

Conectarbd.php:

Definición de Credenciales: Se establecen las variables necesarias para la conexión a la base de datos:

\$user: El nombre de usuario de MySQL (en este caso, 'root').

\$pass: La contraseña de MySQL (vacía en este ejemplo, pero en un entorno real deberías usar una contraseña segura).

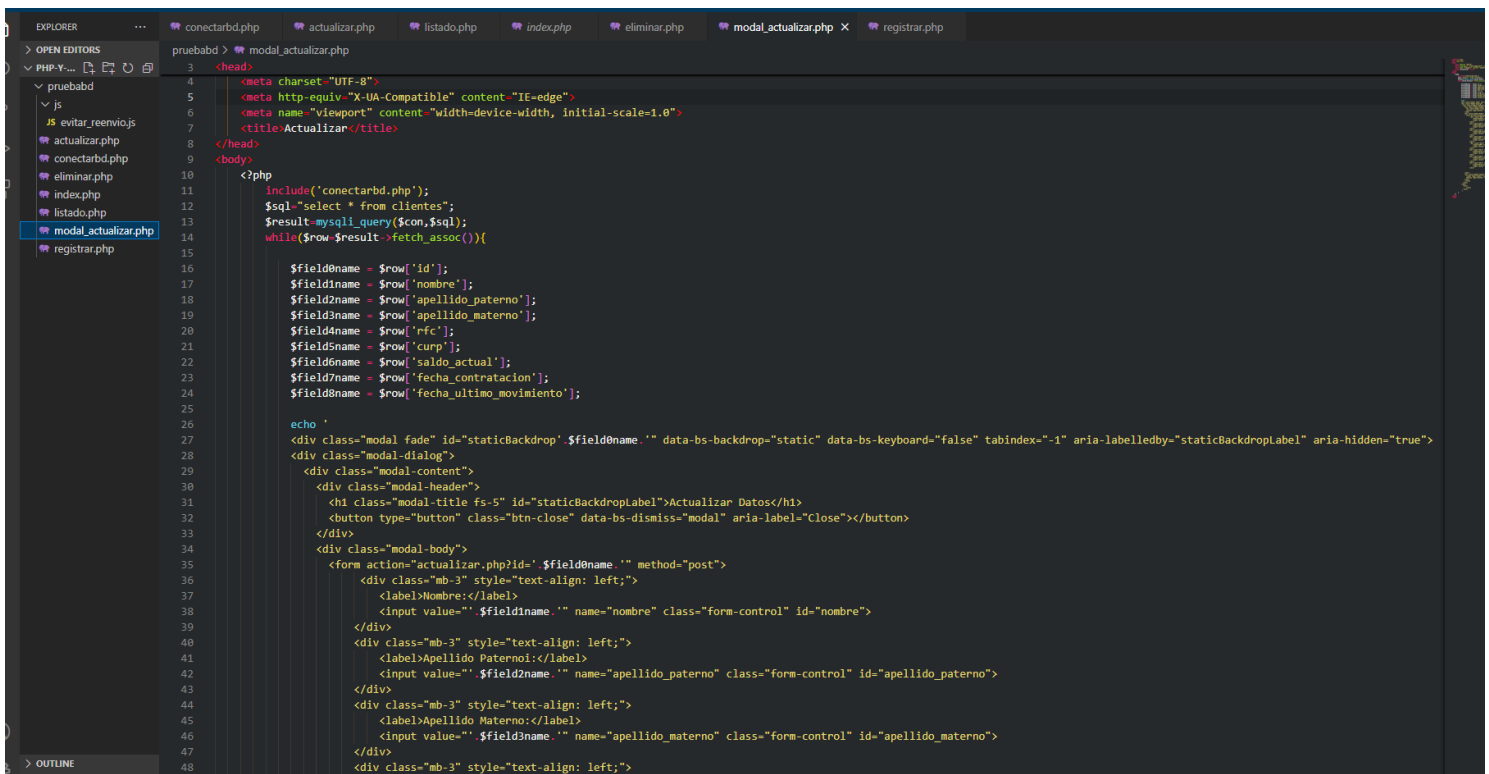
\$server: El servidor donde está alojada la base de datos (en este caso, 'localhost' que significa que la base de datos está en el mismo servidor que el script).

\$bd: El nombre de la base de datos a la que quieres conectarte ('pruebabd').

Conexión a la Base de Datos: Usa la función `mysqli_connect` para intentar conectar con la base de datos utilizando las credenciales proporcionadas.

Verificación de la Conexión:

Si la conexión falla (\$con es false), el script termina con un mensaje de error que incluye detalles sobre el problema de conexión (`mysqli_connect_error()`).



```
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <title>Actualizar</title>
8 </head>
9 <body>
10 <?php
11     include('conectarbd.php');
12     $sql="select * from clientes";
13     $result=mysqli_query($con,$sql);
14     while($row=$result->fetch_assoc()){
15
16         $field0name = $row['id'];
17         $field1name = $row['nombre'];
18         $field2name = $row['apellido_paterno'];
19         $field3name = $row['apellido_materno'];
20         $field4name = $row['rfc'];
21         $field5name = $row['curp'];
22         $field6name = $row['saldo_actual'];
23         $field7name = $row['fecha_contratacion'];
24         $field8name = $row['fecha_ultimo_movimiento'];
25
26         echo '
27         <div class="modal fade" id="staticBackdrop'.$field0name.'" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="staticBackdropLabel" aria-hidden="true">
28         <div class="modal-dialog">
29             <div class="modal-content">
30                 <div class="modal-header">
31                     <h1 class="modal-title fs-5" id="staticBackdropLabel">Actualizar Datos</h1>
32                     <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
33                 </div>
34                 <div class="modal-body">
35                     <form action="actualizar.php?id='.$field0name.'" method="post">
36                         <div class="mb-3" style="text-align: left;">
37                             <label>Nombre:</label>
38                             <input value="'.$field1name.'" name="nombre" class="form-control" id="nombre">
39                         </div>
40                         <div class="mb-3" style="text-align: left;">
41                             <label>Apellido Paterno:</label>
42                             <input value="'.$field2name.'" name="apellido_paterno" class="form-control" id="apellido_paterno">
43                         </div>
44                         <div class="mb-3" style="text-align: left;">
45                             <label>Apellido Materno:</label>
46                             <input value="'.$field3name.'" name="apellido_materno" class="form-control" id="apellido_materno">
47                         </div>
48                         <div class="mb-3" style="text-align: left;">
```

Modal Actualizar.php:

Estructura HTML: Empieza con la estructura básica de un documento HTML, incluyendo metaetiquetas para la codificación de caracteres, compatibilidad con navegadores, y un título.

PHP para Conexión y Consulta:

Incluir Conexión: Incluye el archivo conectarbd.php para conectar a la base de datos.

Consulta SQL: Ejecuta una consulta SQL (SELECT * FROM clientes) para obtener todos los registros de la tabla clientes.

Generación de Modales:

Recorrer Resultados: Usa un bucle while para recorrer cada registro devuelto por la consulta.

Datos del Cliente: Extrae los campos del cliente (como id, nombre, apellido_paterno, apellido_materno, rfc, curp, saldo_actual, fecha_contratacion, y fecha_ultimo_movimiento) y los asigna a variables.

Crear Modales:

Estructura del Modal: Para cada cliente, genera un modal utilizando HTML y las clases de Bootstrap. Cada modal tiene un ID único basado en el id del cliente (id="staticBackdrop'.\$field0name.'") para asegurarse de que los modales no se confundan entre sí.

Formulario de Actualización: Dentro del modal, hay un formulario que permite actualizar los datos del cliente. Cada campo del formulario está pre-llenado con la información actual del cliente.

Botones: Hay dos botones en el modal: uno para guardar los cambios (Guardar Cambios) y otro para cerrar el modal (Cerrar). El botón de guardar envía los datos del formulario al script actualizar.php con el id del cliente como parámetro en la URL.

Cerrar PHP y HTML: El bucle while y el bloque PHP terminan, cerrando así la generación dinámica de los modales.

VISTAS DE LA APP WEB

localhost / 127.0.0.1 / prueba: x x Prueba

localhost/xampp/PHP-y-Mysql-main/pruebad/index.php

Paletools v24.0.1

Todos los marcadores

Nombre:

Apellido Paterno:

Apellido Materno:

RFC:

CURP:

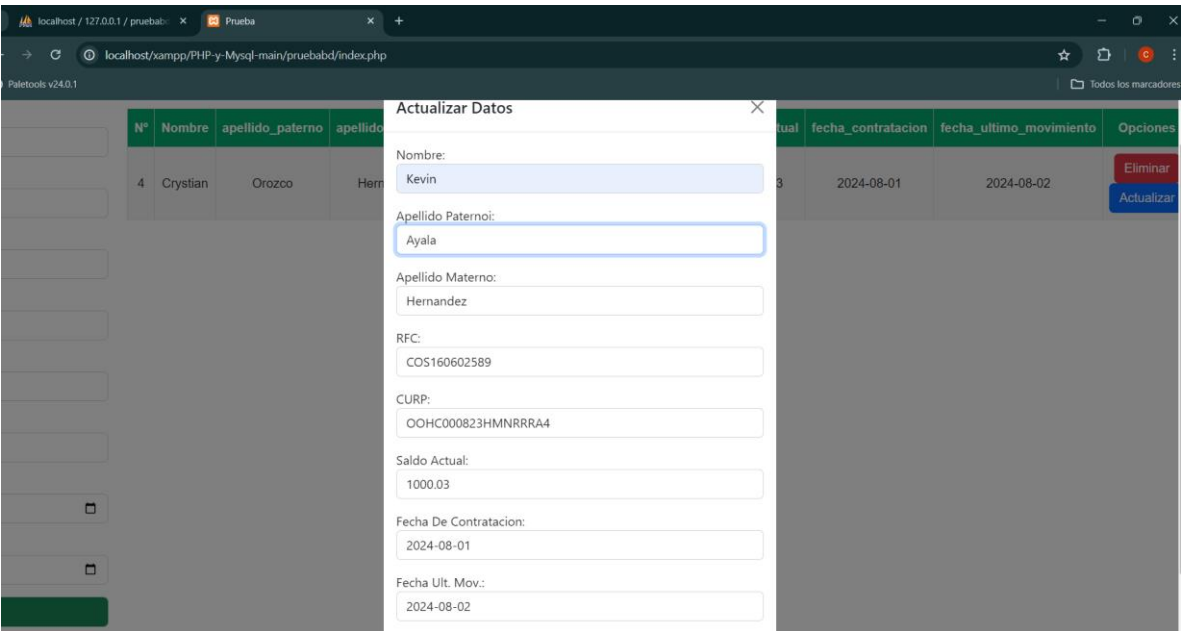
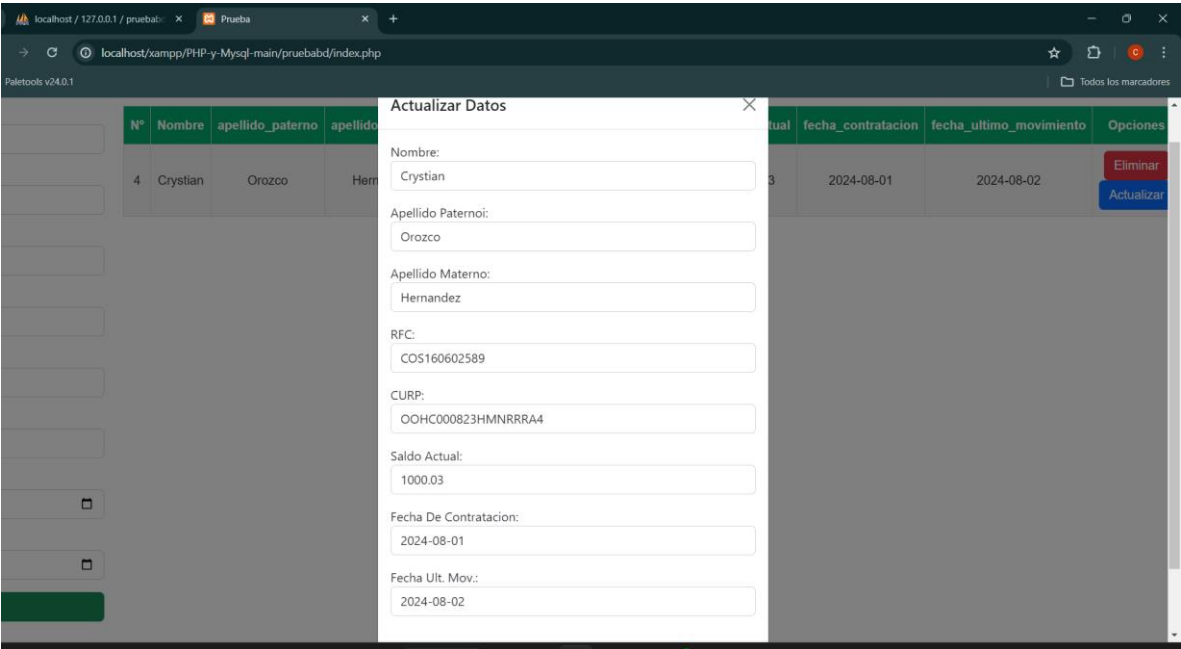
Saldo Actual:

Fecha Contratacion:

Fecha Ult. Mov.:

Registrar

N°	Nombre	apellido_paterno	apellido_materno	rfc	curp	saldo_actual	fecha_contratacion	fecha_ultimo_movimiento
----	--------	------------------	------------------	-----	------	--------------	--------------------	-------------------------



localhost / 127.0.0.1 / pruebab: x Prueba x +

localhost/xampp/PHP-y-Mysql-main/pruebabd/index.php

Paletools v24.0.1

Todos los marcadores

N°	Nombre	apellido_paterno	apellido_materno	rfc	curp	saldo_actual	fecha_contratacion	fecha_ultimo_movimiento	Opciones
4	Kevin	Ayala	Hernandez	COS160602589	OOHC000823HMNRRRA4	1000.03	2024-08-01	2024-08-02	Eliminar Actualizar

localhost / 127.0.0.1 / pruebab: x Prueba x +

localhost/phpmyadmin/index.php?route=/sql&server=1&db=pruebabd&table=clientes&pos=0

Paletools v24.0.1

Todos los marcadores

phpMyAdmin

Reciente Favoritas

No existen tablas favoritas

tablas favoritas

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Operaciones Disparadores

Mostrando filas 0 - 0 (total de 1, La consulta tardó 0.0002 segundos)

SELECT * FROM `clientes`

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Opciones

	id	nombre	apellido_paterno	apellido_materno	rfc	curp	saldo_actual	fecha_contratacion	fecha_ultimo_movimiento
<input type="checkbox"/>	4	Kevin	Ayala	Hernandez	COS160602589	OOHC000823HMNRRRA4	1000.03	2024-08-01	2024-08-02

Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar Exportar

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Operaciones sobre los resultados de la consulta

Imprimir Copiar al portapapeles Exportar Mostrar gráfico Crear vista

Consola

01:52 a.m.