



En esta práctica estaremos desarrollando un CRUD sencillo utilizando el lenguaje de programación php.

CRUD es un acrónimo que representa las cuatro operaciones básicas que se pueden realizar en una base de datos o sistema de gestión de información (Create, Read, Update and Delete).

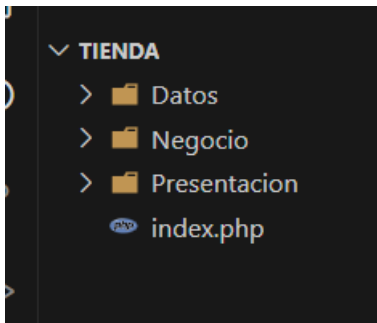
Para poder realizar esta práctica debes descargar el archivo de la base de datos proporcionada en la plataforma e importarla en tu gestor de base de datos mysql.

Una vez importada la base de datos, procedemos a la práctica.

### Paso 1

Creamos la estructura del proyecto, en este ejemplo trabajaremos con la arquitectura de capas.

La estructura debe ser la siguiente:



### Paso 2: configurar la conexión a la base de datos

Dentro de la carpeta Datos creamos un archivo llamado **conf.php** el cual contendrá el siguiente código:



```
Datos > conf.php > Conf > connect()
1  <?php
2
3  2 references | 2 implementations
4  Class Conf {
5      2 references
6      private $server;
7      2 references
8      private $user;
9      2 references
10     private $password;
11     2 references
12     private $database;
13     4 references
14     private $connection;
15     2 references
16     private $result_query;
17
18     0 references | 0 overrides
19     public function __construct() {
20         $this->server = "127.0.0.1"; // IP del localhost, tambien puedes poner localhost
21         $this->user = "root"; // tu usuario, root por defecto
22         $this->password = ""; // tu contrasenia
23         $this->database = "dbtienda"; // nombre de la base de datos
24     }
25
26     // funcion que nos permite conectarnos al servidor y a su vez a la base de datos
27     1 reference | 0 overrides
28     protected function connect() {
29         @$this->connection = mysqli_connect(
30             $this->server,
31             $this->user,
32             $this->password,
33             $this->database)
34             or die("error". mysqli_connect_error());
35         return $this->connection;
36     }
37 }
```

```
// funcion para cerrar la conexion
0 references | 0 overrides
protected function disconnect() {
    return mysqli_close($this->connection);
}

// funcion para ejecutar las consultas
0 references | 0 overrides
public function query($query) {
    $this->result_query = mysqli_query($this->connect() , $query)
        or die("error". mysqli_error($this->connection));
}
```

### Paso 3: Crear las clases



Dentro de la carpeta Negocio crear un archivo llamado **categorías.php** e ingresar el siguiente código:

```
categorias.php X
Negocio > categorias.php > Categorías > list_categories()
1  <?php
2
3  require_once 'Datos/conf.php';
4
5  1 reference | 0 implementations
class Categorías extends Conf {
6
7      // listamos todas las categorías existentes
8      0 references | 0 overrides
    public function list_categories(){
9          $query = "SELECT * FROM categorías";
10         return mysqli_fetch_all($this->exec_query($query), MYSQLI_ASSOC);
11     }
12
13 }
```

Ahora dentro de la misma carpeta creamos **productos.php**

```
Negocio > productos.php > Productos
1  <?php
2
3  require_once 'Datos/conf.php';
4
5  2 references | 0 implementations
class Productos extends Conf {
6      // Atributos de la clase los cuales representan
7      // los campos de la tabla en la base de datos
8      3 references
    public $name;
9      3 references
    public $description;
10     3 references
    public $categoríe;
11
12     // Metodo para listar los productos
13     1 reference | 0 overrides
    public function list_products() {
14         $query = "SELECT * FROM productos";
15         return mysqli_fetch_all($this->exec_query($query), MYSQLI_ASSOC);
16     }
17
18     // Metodo para agregar
19     0 references | 0 overrides
    public function add(){
20         $query = "INSERT INTO productos (
21             name,
22             description,
23             id_categoríe
24         )
25         VALUES (
26             '". $this->name. "',
27             '". $this->description. "',
28             '". $this->categoríe. "'
29         )";
30
31         return $this->exec_query($query);
32     }
```



```

34 // Metodo para actualizar
    1 reference | 0 overrides
35 public function update($id){
36     $query = "UPDATE productos SET
37         name = '". $this->name."',
38         description = '". $this->description."',
39         id_categorie = '". $this->categorie."',
40         WHERE id = '". $id."' ";
41
42     return $this->exec_query($query);
43 }
44
45 // Metodo para eliminar
    0 references | 0 overrides
46 public function delete($id){
47     $query = "DELETE FROM productos WHERE id = '". $id."' ";
48     return $this->exec_query($query);
49 }
50
51 // Metodo para obtener producto por id
    1 reference | 0 overrides
52 public function get_producto($id){
53     $query = "SELECT * FROM productos
54         WHERE id = '". $id."' ";
55     return mysqli_fetch_all($this->exec_query($query), MYSQLI_ASSOC);
56 }
57 }

```

#### Paso 4: Configuración de la navegación

En el archivo **index.php** ubicado en raíz. Y agregamos las CDN de Bootstrap

```

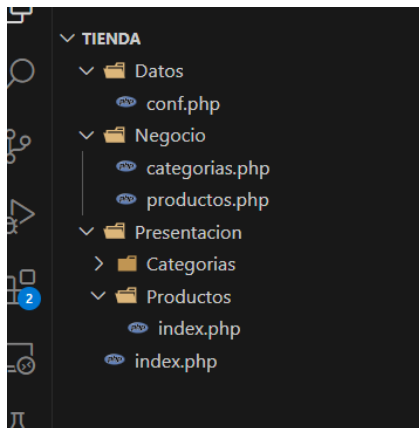
index.php
1  <?php
2
3  // estructura para definir el modulo al cual navegara el usuario
4  switch ( @$_GET["mod"] ) {
5      //dejaremos la direccion por defecto ya que solamente tenemos un modulo
6      default:
7          $module = "Presentacion/Productos/index.php";
8  }
9  ?>
10
11 <!DOCTYPE html>
12 <html lang="en">
13 <head>
14     <meta charset="UTF-8">
15     <meta name="viewport" content="width=device-width, initial-scale=1.0">
16     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
17     <title>Tienda</title>
18 </head>
19 <body>
20
21 <div class="container">
22     <?php
23         require_once $module;
24     ?>
25 </div>
26
27 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"
28 </body>
29 </html>
30

```



Ahora dentro de la carpeta presentación creamos una carpeta llamada Productos y dentro de esta creamos un archivo llamado **index.php**

La estructura debería ser la siguiente



En el archivo **Presentacion/Productos/index.php** pondremos el siguiente código

```
1  <?php
2
3  $form_valid = false;
4
5  switch ( @$_GET["form"] ) {
6      case 'li':
7          // Redireccionamos al archivo en donde se listaran los productos
8          $form = 'Presentacion/Productos/list.php';
9          $form_valid = true;
10         break;
11
12         case 'ad':
13             // redireccionamos al mantenimiento de agregar
14             $form = 'Presentacion/Productos/create.php';
15             $form_valid = true;
16             break;
17
18         case 'up':
19             $form = 'Presentacion/Productos/update.php';
20             $form_valid = true;
21             break;
22
23         case 'Presentacion/Productos/delete.php';
24             $form_valid = true;
25             break;
26
27         default:
28             $form_valid = false;
29             break;
30     }
31
32     if ( $form_valid ) {
33         require_once $form;
34     }
35     else{
36         header('Location: error404.php');
37     }
```



### Paso 5: Crear el diseño

Dentro de la carpeta **Presentacion/Productos** crear un archivo **list.php**

```
Presentacion / Productos / list.php
1  <?php
2
3  // importamos los archivos con las clases
4  require_once 'Negocio/categorias.php';
5  require_once 'Negocio/productos.php';
6
7  // instanciamos las clases
8  $productos = new Productos();
9  $categorias = new Categorias();
10
11  ?>
12
13  <div class="card m-auto mt-5 p-3" style="width: 800px">
14      <h3>
15          <button class="btn btn-success"><i class="bi bi-plus-circle me-1"></i>Agregar</button>
16      </h3>
17      <table class="table">
18          <thead class="thead-dark">
19              <tr>
20                  <th scope="col">Producto</th>
21                  <th scope="col">Descripcion</th>
22                  <th scope="col">Categoria</th>
23                  <th scope="col">Controles</th>
24              </tr>
25          </thead>
26          <tbody>
27              <?php
28              foreach ($productos->list_products() as $prod) <
29                  // obtener la categoria
30                  foreach ($categorias->get_categoria($prod['id_categoria'] as $cat)<
31                      ?>
32                      <tr>
33                          <td><?= $prod['name']; ?></td>
34                          <td><?= $prod['description']; ?></td>
35                          <td><?= $cat['name']; ?></td>
36                          <td>
37                              <button type="button" onclick="editar(<?= $prod['id']; ?>);" class="btn btn-info">
38                                  <i class="bi bi-pencil"></i>
39                              </button>
40                              <button type="button" onclick="eliminar(<?= $prod['id']; ?>);" class="btn btn-danger">
41                                  <i class="bi bi-trash"></i>
42                              </button>
43                          </td>
44                      </tr>
45              </tbody>
46          </table>
47      </div>
48  <?php
49
50  <script>
51      function editar(id) <
52          document.location.replace('index.php?mod=&form=up&id=' + id);
53      >
54      function eliminar(id)<
55          document.location.replace('index.php?mode=&form=del&id=' + id);
56      >
57  </script>
```

```
44      </tr>
45      </tbody>
46      </table>
47      </div>
48      <?php
49
50      <script>
51          function editar(id) <
52              document.location.replace('index.php?mod=&form=up&id=' + id);
53          >
54          function eliminar(id)<
55              document.location.replace('index.php?mode=&form=del&id=' + id);
56          >
57      </script>
```



Crear un nuevo archivo llamado **update.php**

```
Presentation > Productos > update.php
1  <?php
2  // importamos los archivos con las clases
3  require_once 'Negocio/categorias.php';
4  require_once 'Negocio/productos.php';
5
6  $categorias = new Categorias();
7  $productos = new Productos();
8
9  //obtenemos el registro del producto
10 foreach ($productos->get_producto($_GET['id']) as $producto) {
11
12 }
13 ?>
14
15 <div class="card m-auto mt-5 p-3" style="width: 600px;">
16   <h3>Editar producto</h3>
17   <form method="post" action="">
18     <div class="row">
19       <div class="col-3">
20         <button type="button" onClick="location.replace('index.php?mod=&form=li');"
21           class="btn btn-danger">Cancelar</button>
22       </div>
23     </div>
24     <div class="row justify-content-center mt-4">
25       <div class="col-5">
26         <label for="">Nombre:</label>
27         <input type="text" class="form-control" name="name" value="{?= $producto['name']; ?}">
28       </div>
29       <div class="col-5">
30         <label for="">Categoria:</label>
31         <select class="form-control" name="categoria">
32           <?php
33             foreach ($categorias->list_categories() as $categoria) {
34               ?>
35               <option value="{?= $categoria['id']; ?}">
36                 {?= $categoria['name']; ?>
37             </option>
38           </select>
39           <?php
40             ?>
41           </div>
42     </div>
43   </div>
44   <div class="text-center mt-4">
45     <button type="submit" class="btn btn-primary">Actualizar</button>
46   </div>
47 </div>
```



```
42         </div>
43     </div>
44     <div class="row justify-content-center mt-3">
45         <div class="col-10">
46             <label for="">Descripción:</label>
47             <textarea class="form-control" name="description">= $producto['name']; ?&gt;&lt;/textarea&gt;
48         &lt;/div&gt;
49     &lt;/div&gt;
50     &lt;div class="row mt-3 ms-5 mb-3"&gt;
51         &lt;div class="col-3"&gt;
52             &lt;button type="submit" class="btn btn-success"&gt;Guardar&lt;/button&gt;
53         &lt;/div&gt;
54     &lt;/div&gt;
55 &lt;/form&gt;
56 &lt;/div&gt;
57
58 &lt;?php
59
60 if ($_SERVER["REQUEST_METHOD"] == "POST") {
61     $productos-&gt;name = $_POST["name"];
62     $productos-&gt;description = $_POST["description"];
63     $productos-&gt;category = $_POST["category"];
64
65     //ejecutamos el mantenimiento de agregar
66     if ($productos-&gt;update($_GET['id'])) {
67         echo "&lt;script&gt;location.replace('index.php?mod=&amp;form=li')&lt;/script&gt;";
68     } else {
69         echo "error";
70     }
71 }</pre
```

Ahora volvemos a **list.php** y nos ubicamos abajo en el botón Agregar

Realizamos los siguientes cambios al botón

```
<div class="card m-auto mt-5 p-3" style="width: 800px">
    <h3>
        <button type="button" class="btn btn-success" onClick="location.replace('index.php?mod=&form=ad')">
            <i class="bi bi-plus-circle me-1"></i> Agregar
        </button>
    </h3>
    <table class="table">
```

Ahora crear un archivo llamado **create.php** en la misma carpeta **Presentacion/Productos**

Y agregar el siguiente código:





```

Presentacion > Productos > create.php
1  <?php
2  // importamos los archivos con las clases
3  require_once 'Negocio/categorias.php';
4  require_once 'Negocio/productos.php';
5
6  $categorias = new Categorias();
7  $productos = new Productos();
8
9  ??
10
11 <div class="card m-auto mt-5 p-3" style="width: 600px;">
12   <h3>Agregar producto</h3>
13   <form method="post" action="">
14     <div class="row">
15       <div class="col-3">
16         <button type="button" onClick="location.replace('index.php?mod=&form=li');"
17           class="btn btn-danger">Cancelar</button>
18       </div>
19     </div>
20     <div class="row justify-content-center mt-4">
21       <div class="col-5">
22         <label for="">Nombre:</label>
23         <input type="text" class="form-control" name="name">
24       </div>
25       <div class="col-5">
26         <label for="">Categoria:</label>
27         <select class="form-control" name="categorie">
28           <?php
29             foreach ($categorias->list_categorias() as $categoria) {
30               ??
31               <option value="<?= $categoria['id']; ??">
32                 <?= $categoria['name']; ?>
33               </option>
34             <?php
35           }
36           ??
37         </select>

```

```

36     ??
37   </select>
38 </div>
39 </div>
40 <div class="row justify-content-center mt-3">
41   <div class="col-10">
42     <label for="">Descripcion:</label>
43     <textarea class="form-control" name="description"></textarea>
44   </div>
45 </div>
46 <div class="row mt-3 ms-5 mb-3">
47   <div class="col-3">
48     <button type="submit" class="btn btn-success">Guardar</button>
49   </div>
50 </div>
51 </form>
52 </div>
53
54 <?php
55
56 if ($_SERVER['REQUEST_METHOD'] == "POST") {
57   $productos->name = $_POST["name"];
58   $productos->description = $_POST["description"];
59   $productos->categorie = $_POST["categorie"];
60
61   //ejecutamos el mantenimiento de agregar
62   if ($productos->add()) {
63     echo "<script>location.replace('index.php?mod=&form=li')</script>";
64   } else {
65     echo "error";
66   }
67 }

```



Y listo, ya podemos agregar y editar nuestros registros

**Reto de practica:**

- Implementar el mantenimiento de eliminar registros.
- Agregar un mensaje que diga “No hay registros existentes”, en caso de que no exista ningun producto en la base de datos.