

Primero nos creamos una base de datos y una sencilla tablas (yo por equivocación la he llamado usuarios):

```
MariaDB [ajax]> select * from usuarios;
+-----+-----+
| id | link |
+-----+-----+
| 1 | deportes.jpg |
| 2 | bebidas.jpg |
| 3 | revista.jpg |
+-----+-----+
3 rows in set (0,001 sec)
```

Como podemos ver tiene varios datos insertados, esto son las rutas de las imágenes (están al mismo nivel que el html).

Como en la anterior práctica de Ajax, deberemos comunicarnos de forma asíncrona con el servidor. Cada botón es un formulario con un campo hidden de la función deseada a realizar (deportes, revista, etc):

```
<form action="procesar_datos.php" method="post" id="form_deportes">
  <input type="hidden" name="deportes">
  <div class="opciones">
    <button id="btnDeportes" name="deportes" value="deportes">deportes</button>
  </div>
</form>

<form action="procesar_datos.php" method="post" id="form_bebidas">
  <input type="hidden" name="bebidas">
  <div class="opciones">
    <button id="btnBebidas" name="bebidas" value="bebidas">bebidas</button>
  </div>
</form>

<form action="procesar_datos.php" method="post" id="form_revistas">
  <input type="hidden" name="revistas">
  <div class="opciones">
    <button id="btnRevistas" name="revistas" value="revistas">revistas</button>
  </div>
</form>
```

Y cada formulario tiene su método en el script:

```

btnRevistas.addEventListener('click', function (event) {
  //Para que no nos lleve a otro lado cuando damos a enviar.
  event.preventDefault();

  //usamos FormData para formatear los datos del formulario a algo que se pueda procesar
  const formattedFormData = new FormData(formRevistas);

  //pasamos los datos formateados a una funcion que los procesará
  postData(formattedFormData);
})

async function postData(formattedFormData){
  // en fetch especificamos el archivo php que captura los datos enviados
  const response = await fetch('procesar_datos.php',{
    // el metodo a usar
    method: 'POST',
    // los datos a ser enviados
    body: formattedFormData
  });

  // data contendra la respuesta del servidor
  const data = await response.text();
  // cambiamos la url de la imagen
  logoImg.src = data;
}

```

Entonces, el funcionamiento es el siguiente, el archivo php recoge la información enviada como contexto (el formulario en concreto) y según lo que recibe, ejecuta y devuelve una cosa u otra. Si recibe el formulario de deportes, nos va a devolver una imagen de deportes, así con todas:

```

//deportes (id=1)
if(isset($_POST['deportes'])) {
  $id = 1;
  $stmt = $mbd->prepare("SELECT id, link FROM usuarios WHERE id = :id;");
  $stmt->bindParam(':id', $id);
  $stmt->execute();

  foreach ($stmt as $key => $fila) {
    echo $fila['link'];
  }
}

//bebidas (id=2)
if(isset($_POST['bebidas'])) {
  $id = 2;

```

Como resultado obtendríamos este HTML dinámico y asíncrono:

# Logos

deportes

bebidas

revistas

# Logos

deportes

bebidas

revistas



```
<div id="resultados"> flex
  
</div>
```

# Logos

deportes

bebidas

revistas



# Logos

deportes

bebidas

revistas



Con peticiones Ajax y consultas a bases de datos.