

#index.html

colspan y rowspan: pueden ser usados para hacer que las celdas abarquen dos o más celdas en dirección horizontal o vertical, respectivamente.

<https://www.htmlquick.com/es/reference/tags/th.html>

```
<tr>
  <th scope="row" colspan="5">Cargando...</th>
</tr>
```

XMLHttpRequest(): Utilizado para enviar solicitudes HTTP.

https://developer.mozilla.org/es/docs/Web/API/XMLHttpRequest/Using_XMLHttpRequest

```
const xhttp = new XMLHttpRequest();
```

.send(): Envía la solicitud.

```
xhttp.send();
```

onreadystatechange: Listener. <https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest/onreadystatechange>

```
xhttp.onreadystatechange = function() {
```

readyState = 4: Tarea completada. <https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest/readyState>

status = 200: Código de respuesta de estado satisfactorio HTTP 200 OK que indica que la solicitud ha tenido éxito.

<https://developer.mozilla.org/es/docs/Web/HTTP/Status/200>

```
if (this.readyState == 4 && this.status == 200) {
```

Swal.fire: Alertas dulces para indicar en los "textBox" que ingresar. <https://sweetalert2.github.io/>

focusConfirm: false: Establézcalo en falso si desea enfocar el primer elemento en el orden de tabulación en lugar del botón "Confirmar" de forma predeterminada.

preConfirm: La función a ejecutar antes de confirmar, puede ser asíncrona (devolución de promesa) o sincronizada.

```
function showUserCreateBox() {
  Swal.fire({
    title: 'Crear Usuario',
    html:
      '<input id="id" type="hidden">' +
      '<input id="fname" class="swal2-input" placeholder="Nombre">' +
      '<input id="lname" class="swal2-input" placeholder="Apellido">' +
      '<input id="username" class="swal2-input" placeholder="Username">' +
      '<input id="email" class="swal2-input" placeholder="Correo">',
    focusConfirm: false,
    preConfirm: () => {
      userCreate();
    }
  })
}
```

setRequestHeader: establece el valor de un encabezado de solicitud HTTP. <https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest/setRequestHeader>

stringify: convierte un objeto o valor de JavaScript en una cadena de texto JSON.

https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/JSON/stringify

Swal.fire(objects['message']): Mensaje de la API.

```
const xhttp = new XMLHttpRequest();
xhttp.open("POST", "https://www.mecallapi.com/api/users/create");
xhttp.setRequestHeader("Content-Type", "application/json;charset=UTF-8");
xhttp.send(JSON.stringify({
  "fname": fname, "lname": lname, "username": username, "email": email,
  "avatar": "https://www.mecallapi.com/users/cat.png"
}));
xhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    const objects = JSON.parse(this.responseText);
    Swal.fire(objects['message']);
    loadTable();
  }
};
```