

A large, solid yellow graphic element that starts as a thin diagonal line from the top center and widens into a large triangular shape covering the right half of the slide.

**VALIDACIONES**

## Obtener el formulario

El primer objetivo, será obtener el formulario y luego agregar un comportamiento cuando queramos capturar el evento.

```
{} window.addEventListener("load", function(){  
    let formulario = document.querySelector("form.reservation")  
    //Agregamos el evento para atraparlo  
    formulario.addEventListener("submit", function(event){  
        //Form recién atrapado :)  
    });  
});
```

## Validando un campo vacío

Primero y principal, dentro del evento, detendremos el envío de formulario con **event.preventDefault()** .

Luego, obtendremos nuestro **input** con **querySelector** para que finalmente preguntemos si el campo está vacío.

```
{  
  event.preventDefault();  
  let campoNombre = document.querySelector("input.nombre");  
  if(campoNombre.value == ""){  
    alert("El campo nombre no debe estar vacío");  
  }  
}
```

## Array de errores

Nuestro siguiente paso es crear un **array** para acumular estos errores y cambiar nuestra lógica, es decir, si el array no está vacío, entonces prevengo el envío de formulario, caso contrario, el formulario se enviará.

```
let errores = [];  
let campoNombre = document.querySelector("input.nombre");  
if(campoNombre.value == ""){  
    errores.push("El campo nombre está vacío");  
}  
if(errores.length > 0){  
    event.preventDefault();  
}
```

{}

## Mostrando errores en HTML

Nuestro siguiente paso es crear una etiqueta **<div>** dentro de nuestro html donde mostraremos el listado de errores. Para ello, la colocaremos por arriba de nuestro formulario sumándole una etiqueta **<ul>** para indicar el inicio de un listado.

```
{  
  <div class="errores">  
    <ul>  
      <!--Aca irán los errores-->  
    </ul>  
  </div>  
}
```

## Mostrando errores con Javascript

Por último, en nuestro archivo JavaScript, recorreremos con un **for**, el array de errores para que sean mostrados dentro del **<div> <ul>**.

```
let ulErrores = document.querySelector("div.errores ul");
for (let i = 0; i < errores.length; i++) {
    ulErrores.innerHTML += "<li>" + errores[i] + "</li>";
}
```