



4 . Interacción con el usuario

- **Modificación de apariencia y comportamiento -**

Modificación de apariencia y comportamiento

Seleccionar un formulario

Disponemos del siguiente formulario:

```
<form id="miFormulario" name="miFormulario" action="siguiente.php">
```

getElementById()

```
let formulario = document.getElementById("miFormulario");
```

getElementsByName()

```
let formularios = document.getElementsByName("form");  
let primerform = formularios[0];
```

O en una sola línea

```
let primerform = document.getElementsByName("form")[0];
```



Modificación de apariencia y comportamiento

Seleccionar un formulario

Disponemos del siguiente formulario:

```
<form id="miFormulario" name="miFormulario" action="siguiente.php">
```

forms[0]

```
let formularios = document.forms; // Devuelve un array de formularios  
let primerform = formularios [0];
```

0 en una sola línea

```
let formularios = document.forms[0];
```



Modificación de apariencia y comportamiento

Seleccionar un formulario

Disponemos del siguiente formulario:

```
<form id="miFormulario" name="miFormulario" action="siguiente.php">
```

`forms["nombre"]`

```
let formularios = document.forms; //Devuelve un array de formularios  
let primerform = formularios ["miFormulario"];
```

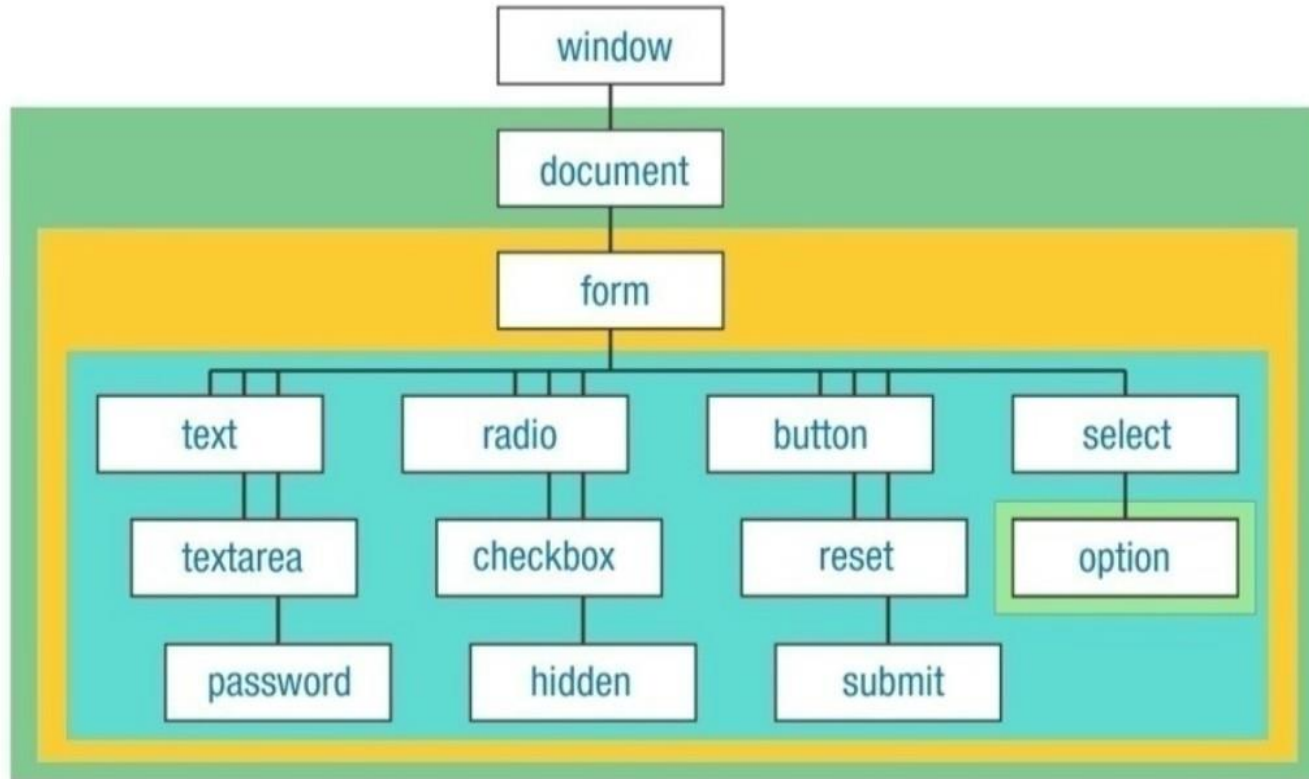
O en un sola línea

```
let formularios = document.forms["miFormulario"];
```



Modificación de apariencia y comportamiento

El formulario como contenedor de objetos



Modificación de apariencia y comportamiento

Selección de objetos de un formulario

form.elements[]

devuelve un *array* con todos los input de un formulario.

getElementByID()

devuelve un *elemento* que tenga un id determinado.

getElementsByTagName()

devuelve un *array* de elementos de un tipo.

Modificación de apariencia y comportamiento

Selección de objetos de un formulario

```
<!-- En este ejemplo: -->
<form id="formularioBusqueda" action="siguiente.php">
  <p>
    <input type="text" id="entrada" name="cEntrada">
    <input type="submit" id="enviar" name="enviar" value="Buscar...">
  </p>
</form>
```

```
// Para hacer referencia al input de tipo texto podemos hacerlo mediante:
document.getElementById("entrada")
document.formularioBusqueda.cEntrada
document.formularioBusqueda.elements[0]
document.forms["formularioBusqueda"].elements["cEntrada"]
document.forms["formularioBusqueda"].cEntrada
```



Modificación de apariencia y comportamiento

Objetos input de tipo texto

Es recomendable utilizar un **id** igual que el **name** para poder referirnos a ellos con `getElementById`
Podemos referirnos a cualquiera de sus propiedades para cambiarlas.

Ejemplo:

```
<input type="text" name="nombre" id="nombre"/>  
document.getElementById("nombre").value="Valor nuevo"; //Cambia el valor del texto
```

Objetos input de tipo checkbox

La propiedad **value** es el texto asociado al objeto.
El formulario solamente envía los datos si el checkbox está marcado.

Ejemplo:

```
<input type="checkbox" id="cantidad" name="cantidad" value="100"/>  
document.getElementById("cantidad").checked = true; //Marca el checked
```


Modificación de apariencia y comportamiento

Objetos input de tipo radio

Cuando utilizamos varios radiobutton con el mismo nombre, se crea un array de objetos input radio.

Podemos consultar el número de radiobutton con `formulario.gruporadio.length`;

Podemos consultar si un radio button está marcado con `formulario.gruporadio[num].checked` y devolverá `true` o `false`.

Ejemplo:

```
for (var i = 0; i < document.formulario.frutas.length; i++) {  
    if (document.formulario.frutas[i].checked) // Comprueba que es true  
        alert(document.formulario.actores[i].value);  
}
```

Objetos select

La propiedad `selectedIndex` devuelve el índice de la opción que ha sido seleccionada.

Cada opción tiene dos propiedades accesibles: `text` y `value` que indican el valor visible y el valor interno, respectivamente.

Ejemplo:

```
<option value="ZA">Zamora</option>  
formulario.nombreCampoSelect.options[n].text; // Devuelve Zamora  
formulario.nombreCampoSelect.options[n].value; // Devuelve ZA
```



Modificación de apariencia y comportamiento

Usando this

Utilizamos this para hacer referencia al objeto que llama a un evento.

Ejemplo:

```
<input type="text" name="nombre" id="nombre" value="Chema" onclick="identificar(this)" />

function identificar(objeto) {
    var nombre = objeto.name;
    var valor = objeto.value;
    var id = objeto.id;
    var tipo = objeto.type;
    alert(
        "Datos del objeto: " + nombre + " - " + valor + " - " + id + " - " + tipo
    );
}
```



END



prof.jduran@iesalixar.org