

# **Unitat Didàctica 07 :**

## **Formularis**



# Fonaments sobre formularis.

- Un formulari HTML es definix mitjançant l'element **<form></form>** :
  - ▣ **method**: Indica si el navegador ha d'enviar una sol·licitud de tipus GET o POST.
  - ▣ **action**: Indica l'URL a la qual enviar el formulari.
  - ▣ **enctype**: La manera de codificar les dades quan s'envien al servidor.
  - ▣ **reset()** y **submit()** ens permet resetear i enviar el formulari des del script.
- Quan enviem un formulari utilitzem els botons creats per a això :  
**<input type="submit" value="send">** **<input type="button" value="send">**
- En l'element **type="submit"** l'esdeveniment **submit** s'executa abans donant-nos l'oportunitat de validar les dades abans de permetre l'enviament.

Si necessitem cancel·lar l'enviament de dades recordem : **event.preventDefault()** o tornar **false**
- En l'element **type="button"** es llança l'esdeveniment **click** i no s'envien les dades.

Si necessitem enviar les dades haurem de fer-ho amb el mètode **formulari.submit()**
- Es convenient deshabilitar el formulari o el botó enviar una vegada es vagen a enviar les dades per a evitar que l'usuari polse dos vegades el botó d'enviar  
**formulari.disabled = true**

# Accedir a la informació d'un formulari

- Es necessari obtindre una referència a l'elemento <form>
  - Utilitzar el mètode getElementById() passant-li l'id del formulari.

```
var oForm = document.getElementById("form1");
```

- També es pot fer referència a l'objecte **forms** per a la seua **posició en l'array** o per el seu atribut **name** :

```
var oForm = document.forms[0];  
var oOtherForm = document.forms["formX"];
```

- Per a accedir als elements d'un formulari es poden utilitzar mètodes natius DOM o també podem utilitzar la propietat **elements** de l'objecte **forms** ( **És un array que només torna els elements input**)

- Es pot fer referència a l'objecte **elements** per a la seua **posició en l'array** o per el seu atribut **name** :

```
var oFirstField = oForm.elements[0]  
var oTextbox1 = oForm.elements["textbox1"];
```

- També per el mètode abreviat encara que desapareixerà en el futur

```
var oTextbox1 = oForm.textbox1;
```

# Accedir a la informació d'un formulari

- Per a accedir a la informació d'un element :

- input type="text" , type="password", textarea => **.value**
- input type="radio", type="checkbox" => **.checked**    **.value**
- select => **.options** → Array amb els elements option

**.selectedIndex** → Indica la posició de l'option seleccionat (la primera posició si es multiselecció) i -1 en cas contrari. (Es pot modificar el valor)

- Cada <option> afig les següents propietats i mètodes :
  - **index**: Índex de l'opció dins de l'array.
  - **label**: L'attribut label
  - **selected**: Booleà que indica si ha sigut seleccionada.
  - **text**: Text de l'opció.
  - **value**: La propietat value.

# Accedir a la informació d'un formulari

- `input type="file"` => **.files** → array d'objectes File

Propietats :

- **.length** : Nombre d'elements de files
- Per a cada element :
  - **.name** : Nom de l'element File
  - **.lastModified, .lastModifiedDate** : Data de l'última modificació del fitxer File
  - **.size** : Tamany del fitxer File
  - **.type** : Tipus MIME del fitxer File
  - **.webkitRelativePath** : Ruta relativa de l'URL del File

<Exemple 5.3>

# Esdeveniments camps de formulari :

- L'esdeveniment **blur** es produïx quan el camp perd el focus. Existix el mètode **blur()**.
- L'esdeveniment **focus** es produïx quan el camp rep el focus. Existix el mètode **focus()**.
- L'esdeveniment **change** es produïx quan el camp perd el focus i el contingut ha canviat

<Exemple 5.4>

- L'esdeveniment **select** es produïx quan l'usuari selecciona un text. Existix el mètode **select()**
  - \* HTML5 proporciona dos propietats (selectionStart y selectionEnd) que ens proporcionen els límits del text seleccionat.
  - \* **SelectionRange**(indiceInicial,IndiceFinal) ens permet seleccionar part del text.

# Validació de formulari :

- HTML 5 proporciona attributs i elements input que realitzen validacions de manera automàtica.

[https://www.w3schools.com/tags/tag\\_input.asp](https://www.w3schools.com/tags/tag_input.asp)

- Exemples d'attributs : max, min, maxlength, required, pattern...
- Exemples d'elements input : type="number", type="email", type="date", type="url", type="color", ...
  - S'ha de comprovar la compatibilitat entre els navegadors
- On no es puga arribar amb validació HTML5, recorrerem a Javascript i podem utilitzar aquests mètodes i attributs :

**.checkValidity()** : Torna true si les validacions HTML5 es complixen

**.validationMessage** : Missatge d'error que ens proporciona el navegador

**.validity.ValidityProperty** : Indica si no es complix la validació per culpa de ValidityProperty

- Llista de properties : [https://www.w3schools.com/js/js\\_validation\\_api.asp](https://www.w3schools.com/js/js_validation_api.asp)

# Expressions regulars :

- Les expressions regulars són patrons que s'utilitzen per a fer coincidir combinacions de caràcters en cadenes de text.
  - Per a comprovar una expressió regular es pot utilitzar el mètode **test** `re.test(cadena)` `re` → expressió regular
  - L'expressió regular es definix entre barres inclinades `re = /.../`
  - Es pot consultar el fitxer :
    - `cheat-sheet-regular-expressions.pdf`
    - Exemples que poden ser d'utilitat:
      - `[A-Za-z]{2,15}` : Conté només lletres entre 2 i 15 caràcters
      - `[0-9]{9}` : Conté exactament 9 dígitos.

Més informació en :

[https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Regular\\_Expressions#crear\\_una\\_expressi%C3%B3n\\_regular](https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Regular_Expressions#crear_una_expressi%C3%B3n_regular)