

**DAW CFGS Desenvolupament d'aplicacions web**  
**Mòdul 6 bloc 1 – Desenvolupament web en entorn client**  
**UF1 – Sintaxi del llenguatge. Objectes predefinits del llenguatge**

**EAC1**  
**(Curs 2020–21 / 1r semestre)**

**Presentació i resultats d'aprenentatge**

Aquest exercici d'avaluació continuada (EAC) es correspon amb els continguts treballats a la unitat 1 Sintaxi del llenguatge. Objectes predefinits del llenguatge.

Els **resultats d'aprenentatge** que es plantegen són:

- Selecciona les arquitectures i tecnologies de programació sobre clients web, identificant i analitzant les capacitats i característiques de cadascuna.
- Escriu sentències simples, aplicant la sintaxi del llenguatge i verificant la seva execució sobre navegadors web.

**Criteris d'avaluació**

La puntuació màxima assignada a cada activitat s'indica a l'enunciat.

Els criteris que es tindran en compte per avaluar el treball de l'alumnat són els següents:

- Caracteritza i diferencia els models d'execució de codi al servidor i al client web.
- Identifica les capacitats i els mecanismes d'execució de codi dels navegadors web.
- Identifica i caracteritza els principals llenguatges relacionats amb la programació de clients web.
- Reconeix les particularitats de la programació de guions i els seus avantatges i desavantatges sobre la programació tradicional.
- Verifica els mecanismes d'integració dels llenguatges de marques amb els llenguatges de programació de clients web.
- Reconeix i avalua les eines de programació sobre clients web.
- Selecciona un llenguatge de programació de clients web en funció de les seves possibilitats.
- Utilitza els diferents tipus de variables i operadors disponibles en el llenguatge.
- Identifica els àmbits d'utilització de les variables.

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 1 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

- Reconeix i comprova les peculiaritats del llenguatge respecte a les conversions entre diferents tipus de dades.
- Utilitza mecanismes de decisió en la creació de blocs de sentències.
- Utilitza bucles i verifica el seu funcionament.

## Forma i data de lliurament

Per respondre les qüestions heu d'utilitzar aquest mateix fitxer. Elimineu els apartats *Presentació i resultats d'aprenentatge*, *Criteris d'avaluació* i *Forma i data de lliurament*.

Un cop finalitzat l'exercici d'avaluació continuada heu d'enviar el document des de l'apartat **M6B1 Lliurament EAC1** de l'aula, dins del termini establert. Tingueu en compte que el sistema no permetrà fer lliuraments després de la data i hora indicades.

El nom del fitxer serà el següent: **DAW\_M06B1\_EAC1\_Cognom1\_Inicial del cognom2.zip** Els cognoms s'escriuran sense accents. Per exemple, l'estudiant *Joan García Santos* posaria el següent nom al seu fitxer de l'EAC1: **DAW\_M06B1\_EAC1\_Garcia\_S.zip**.

Substituïu **Nom i cognoms** de la capçalera per les vostres dades personals.

El termini de lliurament finalitza a les **23:55 h** del dia **28/10/2020**. La proposta de solució de l'EAC es publicarà el dia **30/10/2020** i les qualificacions el dia **05/11/2020**.

## Enunciat

Per publicar la solució heu d'utilitzar aquest mateix model, eliminant els apartats *Presentació i resultats d'aprenentatge*, *Criteris d'avaluació* i *Forma i data de lliurament i Enunciat* i substituint *Nom i cognoms* pels vostres.

Cal lliurar el fitxer comprimit amb format .zip indicat que contindrà:

- Les respostes als dos primers exercicis en un fitxer que s'anomenarà:

**DAW\_M06B1\_EAC1\_Cognom1\_Inicia del cognom2.odt**

Aquest fitxer serà el mateix que conté aquest enunciat, però, amb les respostes als dos primers exercicis i eliminant els apartats que s'han indicat abans.

- Les respostes a la resta d'exercicis, que seran els fitxers amb codi JavaScript (.js) que es demanen a cada exercici.

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 2 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

**1 (1.5 punts)** Per a cada afirmació, indiqueu si és certa amb una 'C' o falsa, amb una 'F' en el requadre que acompanya a cada afirmació. Tingueu cura d'escriure-la correctament, atès que la correcció d'aquest test serà semiautomàtica.

Tingueu present que, perquè una afirmació sigui considerada certa, tota ella ha de ser-ho. Per exemple, l'afirmació «En aquest mòdul es treballa amb JavaScript. JavaScript és un llenguatge compilat.» caldria considerar-la falsa, ja que, tot i que és certa la primera part, no ho és la segona.

Cada resposta encertada suma 0,075 punts. Cada resposta errònia resta 0,075 punts. En cap cas l'exercici sumarà menys de zero punts.

Afirmació	Cert (C) / Fals (F)
"1" == 1 dóna cert	C
El codi JavaScript integrant en codi HTML s'etiqueta amb <script> i </script>	C
L'ordre alert("<b>hola</b>") mostra una finestra amb el text hola en negreta	F
a == b compara tant el valor com el tipus d'a i b	F
A JavaScript "2"*3 és 6	C
Es pot utilitzar un bucle for per recórrer un desplegable	C
En una aplicació web el protocol de comunicacions és SMTP	F
El fet de que JavaScript s'executi en el client li afegeix un plus de seguretat	F
Per comunicar-se entre sí, dues finestres independents han d'utilitzar localStorage	C
Per saltar línia quan escrivim en una pàgina HTML utilitzarem \n	F
Una aplicació web té arquitectura client-servidor	C
L'expressió "El barroc té l'origen al segle " + 16 es correcta	C
Les dades de localStorage s'envien sempre al servidor quan es recarrega la pàgina	F
Javascript sols té variables boolean int double i array	F
A JavaScript és obligatori declarar les variables	F
Amb javascript no es pot canviar una imatge del document HTML	F
Podem canviar el valor d'un checkbox des de JavaScript	C
La tendència actual és separar el codi en dos blocs: contingut i funcionalitat	C
Les expressions a == b i a === b sempre donen el mateix resultat	F
A JavaScript, "2"==2 és true i "2"===2 és false	C

## ATENCIÓ:

- Per fer els següents exercicis cal tenir el fitxer **DAW\_M06B1\_EAC1\_codi\_2021S1.zip** ja descomprimit. En els següents enunciats sempre parlarem directament de les carpetes que conté.
- Tots els programes amb què treballareu (excepte el de l'exercici 2 i el de l'annex) s'han fet procurant separar al màxim el codi HTML i el codi JavaScript. Per aquest motiu, al fitxer HTML només hi ha una referència a un fitxer .js, captures a events que criden a funcions d'aquest fitxer .js i, en un d'ells, una crida directa a una d'aquestes funcions. En el bloc 2 del mòdul s'estudiaran recursos per, inclús, eliminar del codi HTML aquestes crides a funcions.

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 3 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

- Als exercicis 3 a 6 us trobareu amb fitxers .html **que no heu de modificar** i que contenen una referència a un fitxer .js. La vostra tasca consistirà a escriure aquest fitxer .js de manera que, almenys, contingui les funcions que són cridades des del fitxer .html. On es consideri adient se us donaran ajuts per facilitar-vos la confecció d'aquest fitxer.
- L'exercici es corregirà amb el navegador Firefox de Mozilla, ja que és el que es recomana per treballar amb el campus.
- Els missatges d'errors els donareu utilitzant la funció `alert()` de JavaScript.
- Recordeu que tant les errades sintàctiques del codi com les errades d'execució apareixen a la consola del depurador. A l'exercici 2 s'explica com accedir-hi.

## 2. (1.5 punts) Treball amb el depurador

La finalitat de l'exercici és familiaritzar-se amb el depurador que porta integrat el vostre navegador web. Assolir aquest objectiu us ajudarà força a fer la resta d'exercicis. La manera d'accedir a aquest depurador varia d'un navegador a un altre i entre versions del mateix navegador.

A continuació teniu un parell d'exemples:

- A Mozilla Firefox, s'hi accedeix fent clic al símbol (a la part dreta de la barra del navegador) i, després, seleccionant les opcions: *Desenvolupador* → *Depurador*. Veureu que la pantalla es divideix en 2 i que, a la part de sota, apareix el codi de la pàgina web (columna central), els diferents fitxers (columna de l'esquerra) i una columna a la dreta que permet visualitzar el contingut de les variables (si aquesta darrera columna no aparegués, podeu fer que es mostri fent clic a la icona, que apareix a la dreta de la barra del depurador). Per últim, a la part esquerra de la barra del depurador veureu unes icones per aturar l'execució o fer-la pas a pas i, a sobre d'aquesta barra, una pestanya que ens permet veure la consola. Podem posar també punts d'interrupció (o *breakpoints*) fent clic a una línia del codi. Els punts d'interrupció se suprimeixen fent-hi clic a sobre.
- A Google Chrome, s'hi accedeix fent clic al símbol (a la part dreta de la barra del navegador) i, després, seleccionant les opcions *Més eines* → *Eines per a desenvolupadors*. Ens apareix a la dreta un requadre amb aquestes eines. Cal triar la pestanya *Sources* perquè puguem navegar pels diferents fitxers de l'aplicació. Veureu que el requadre es divideix en quatre, que d'esquerra a dreta i de dalt a baix són: explorador de fitxers, contingut del fitxer seleccionat, informació de depuració i icones per aturar l'execució o fer-la pas a pas i, per últim, observació de variables (podem posar-hi variables per veure el seu valor). La manera de posar punts d'interrupció (o *breakpoints*) és similar a com es fa a Mozilla Firefox.

### Es demana:

**a)** Obriu amb el navegador el fitxer **depurador.html**, a continuació, obriu el depurador, poseu un punt d'interrupció a la línia 14 (`perimetre=2*Math.PI*radi;`) del fitxer `js/depurador.js`. A continuació, escriviu un número vàlid al quadre de text, i feu clic al botó «Calcula». L'execució s'aturarà en aquesta línia 14. En aquest moment, visualitzeu el contingut de la variable **àrea**, feu una captura de pantalla i enganxeu-la aquí. Abans de fer la captura doneu a la finestra una mida suficient perquè es vegi tot el rellevant, però no excessiva (sinó, en posar la captura al requadre és possible que no es vegi bé).

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 4 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020



## b) Expliqueu per a què serveix el “toFixed(4)” que apareix a la línia 15.

Formata un número emprant notació de punt fixe. La sintaxis de la funció és la següent:

`numObj.toFixed([dígits])`

On el paràmetre opcional *dígits* són els número de dígits que apareixen després del punt decimal. En el nostre cas arrodoneix a 4 decimals. Si no s'especifica cap paràmetre, s'assumeix que és 0. Aquesta funció retorna una cadena.

En la captura de pantalla es pot veure el valor de la variable *area* **abans** i **després** d'aplicar la funció.



	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 5 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

### 3. (1.5 punts) Programa introductori

En aquest exercici completareu un programa web molt senzill que respon a la següent pantalla:

## Treball amb radiobuttons

Numero

Acció: ☐ Doble ☐ Triplica

En fer clic al botó Executa, el programa primer ha de comprovar:

- Que hi hagi alguna cosa escrita a la caixa, i que sigui un número.
- Que hi hagi seleccionada alguna acció.

Si no es compleixen aquestes condicions, el programa mostrarà unes alertes indicant l'error. Si es compleixen, el programa multiplicarà per 2 o per 3 el número introduït, i mostrarà el resultat al paràgraf «resultat» del fitxer inicial.html.

En fer clic al botó Neteja, el programa neteja el quadre de text, desmarca les accions, i esborra el missatge del paràgraf «resultat».

El fitxer .html ja està fet. Es troba a la carpeta **inicial**. No es pot modificar.

**Es demana:** escriure el fitxer **inicial/js/inicial.js**, que contindrà una implementació correcta de les funcions *executa()* i *neteja()*, que són cridades des del fitxer *inicial.html*. Lògicament, si ho creieu necessari, el fitxer pot contenir més funcions. Haureu d'afegir el fitxer **inicial.js** al fitxer comprimit que entregareu.

#### Ajut:

Per saber si un radiobutton que té un identificador "exemple", està clicat, farem:

```
if(document.getElementById("exemple").checked) {...}
```

Podeu modificar la propietat checked de manera molt similar:

```
document.getElementById("exemple").checked=false;
```

En el següent enllaç trobareu un vídeo que us pot ajudar a comprendre el funcionament de l'exercici.

<https://www.loom.com/share/915eae62706c41aa8b2d476f35c284d5>

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 6 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

## 4. (1.5 punts) Treball amb desplegable

En aquest exercici l'objectiu és aprendre a treballar amb desplegable, i correspon la següent pantalla:

### Treball amb desplegable

Text

Valor

### Introducció als desplegable

Com sabreu, els desplegable en HTML tenen la següent estructura:

```
<select id="provincies">  
  <option value="25"> Lleida </option>  
  <option value="08"> Barcelona </option>  
  <option value="17"> Girona </option>  
  <option value="43"> Tarragona </option>  
</select>
```

Aquest exemple, ens mostraria un desplegable amb 4 opcions (Lleida-Barcelona-Girona-Tarragona) que l'usuari podria escollir. No obstant, pel programador, sovint és més interessant el valor (value) que el text que mostrem a l'usuari. Fixeu-vos com en l'exemple del requadre, el valor i el text no tenen per què ser el mateix.

### Enunciat

Així doncs el que es demana en aquest exercici és que programeu tot el funcionament d'un desplegable amb JavaScript.

El fitxer .html ja està fet, s'anomena **desplegable.html** i no es pot modificar.

**Es demana:** escriure el fitxer **desplegables/js/desplegables.js**, que contindrà una implementació correcta de les funcions: Concretament:

- **afegir():** ha d'afegir un element al final del desplegable. El text que ha d'aparèixer és el que l'usuari escriu al camp de text amb identificador *text*, i el valor que ha de tenir l'element del desplegable és el que l'usuari indiqui al camp amb identificador *valor*.
- **eliminar():** ha d'eliminar l'element seleccionat del desplegable.
- **buidar():** ha d'eliminar tots els elements del desplegable.
- **visualitzar():** ha de mostrar a la divisió (<div>) amb identificador "resultat" el text i el valor de l'element seleccionat del desplegable.
- **afegirControlant():** aquesta funcionalitat és molt semblant a la d'afegir, però tenint en compte que no es permeten valors duplicats. Així doncs, quan afegim amb aquest mètode, primer haurem de comprovar que no existeixi cap element amb el mateix valor al desplegable.

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 7 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

- En cas que existeixi un element amb el mateix valor, mostrarem un missatge d'error.
- En cas que no n'existeixi cap, afegireu l'element al desplegable.

**Caldrà que inclogueu** al comprimit del lliurament el fitxer desplegable.js esmentat abans.

**En el següent enllaç trobareu un vídeo que us pot ajudar a comprendre el funcionament de l'exercici.**

<https://www.loom.com/share/254262fa997c4c4f8e924462015efe6d>

**Ajut:**

- **No cal controlar res de les caixes de text. Suposarem que sempre ens introdueixen un text i un valor.**
- Si partim del desplegable d'exemple anomenat províncies, que es troba a l'inici d'aquest exercici, tenim:
  - Per agafar el desplegable: `var x=document.getElementById("provincies");`
  - Per afegir un nou element al desplegable:
    - s'ha de crear un element primer
      - `var element=document.createElement("option")`
    - s'ha d'assignar el text i el valor a les propietats `text` i `value` de `element`
    - s'ha d'afegir al desplegable aquest nou element:
      - `x.add(element)`
  - Per saber quin és l'index seleccionat: `x.selectedIndex`
  - Per saber quants elements hi ha, `x.length`
  - Per eliminar un element: `x.remove(index)`. (Index és un numero entre 0 i `x.length-1`)
  - Per agafar un element: `x.options[index]`, on `index` és un número entre 0 i `x.length-1`
  - Per agafar el text o valor d'un element: `x.options[index].text` o `x.options[index].value`;
  - Per buidar la llista, i fer el mètode `afegirControl`, haureu de recórrer tots els elements del desplegable amb un bucle.

## 5. (2 punts) Treball amb localStorage

Completeu una aplicació web que simuli un editor de text. Quan obriu el fitxer, trobareu una pantalla com la següent:

## Editor de textos

Text:

Mida font:  Negreta: ☐

Cursiva: ☐

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 8 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020



Per la correcta realització de l'exercici, teniu els següents fitxers amb l'enunciat:

- editor.html: pàgina HTML. **No es pot modificar.**
- js/editor.js: És el document que heu de modificar i entregar. Hi trobareu detallades les funcions a implementar, amb comentaris que us ajudin a implementar-les.

Es demana que implementeu les 5 funcions associades als botons nou, afegir text, guardar text, recuperar text, i esborrar text. Han de fer el següent:

- **nou()**: simula la funció de crear un document nou. Això és, buida el camp amb identificador *text*, i buida la divisió (<div>) amb identificador *resultat*.
- **afegir()**: Afegeix un nou paràgraf a la divisió resultat. Això significa que s'ha d'afegir a la divisió amb identificador *resultat*, el text que l'usuari escrigui al camp amb identificador *text*, sense esborrar el que hi havia. Si us fixeu, hi ha un camp anomenat *font*, un anomenat *negreta*, i un anomenat *cursiva*, que es detallen a continuació:
  - camp **font**: hi ha d'haver un número que indica la mida de la font que volem utilitzar en el paràgraf que estem afegint. Si no s'indica número, la font serà de mida 12.
  - Camp **negreta**: indica si volem que el paràgraf que estem afegint aparegui en negreta.
  - Camp **cursiva**: indica si volem que el paràgraf que estem afegint aparegui en cursiva.
- **guardar()**: guarda el text de la divisió *resultat* al localStorage..
- **recuperar()**: recupera el text guardat del localStorage, i el situa a la divisió *resultat*. En cas que no hi hagi res guardat al localStorage, mostra una alerta, i no modifica la divisió *resultat*.
- **esborrar()**: esborra el localStorage..

**Es demana:** completeu el fitxer **js/editor.js**, que contindrà una implementació correcta de les funcions que es demanen al mateix fitxer. No cal que controleu errors als camps de text.

**Caldrà que inclogueu** al comprimit del lliurament el fitxer **editor.js** esmentat abans.

**En el següent enllaç trobareu un vídeo que us pot ajudar a comprendre el funcionament de l'exercici.**

<https://www.loom.com/share/729bda8cfa974eb59c16acab8f348042>

## Ajuts:

- Necessiteu una galleta que no caduqui. Com l'especificació de les galletes no proporciona aquesta possibilitat, caldrà gravar les galletes perquè expirin, per exemple, d'aquí a 100 anys.
- Per donar format al paràgraf que s'afegeix cada cop que cliquem al botó afegir, podeu utilitzar l'atribut *style*, donant-li els següents valors segons el que estigui marcat:
  - font-weight: bold;
  - font-style: italic;
  - font-size: 12px;

Un exemple complet del que hauríeu d'afegir, si ens han entrat el text "hola", una mida 16, i ens han marcat negreta i cursiva, seria:

**<p style="font-size: 12px;font-weight: bold;font-style: italic;">hola</p>**

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 9 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

**6. (2 punts) Treball amb galetes (cookies)**

En aquest exercici, creareu una web que simularà una casa domòtica, treballant amb **checkbox**, **imatges** i **cookies**. Aquesta casa constarà de tres habitacions, en les quals es podrà decidir si volem la calefacció oberta o no. La web quedarà així:



**Calefacció domòtica**

Habitacio1: ☐

Habitacio2: ☐

Habitacio3: ☐

Actualitza    Desa    Recupera    reset

Per la correcta realització de l'exercici, teniu els següents fitxers amb l'enunciat:

- domotica.html: pàgina HTML. **No es pot modificar.**
- css/domotica.css: **No es pot modificar.** CSS associat al document HTML per dibuixar els requadres que contenen les imatges.
- Img/: En aquest directori trobareu 2 imatges amb extensió **png**. Aquestes imatges s'anomenen **fred.png** i **calor.png**.
- js/domotica.js: És el document que heu de modificar i entregar. Hi trobareu detallades les quatre funcions a implementar, amb comentaris que us ajudin a implementar-les,

Es demana que implementeu les 4 funcions associades als botons actualitza, desa, recupera, i reset. Han de fer el següent:

- **actualitza()**: comprova si els checkboxes estan marcats o no, i assigna la imatge calor.png a cada habitació on el checkbox estigui marcat, i la imatge fred.png a les habitacions on els checkbox no estan marcats.
- **desa()**: crea tres *cookies* on hi emmagatzema si els checkbox estan marcats o no. Mostra una alerta indicant que s'han desat les dades i crida al mètode actualitza.
- **recupera()**: recupera les *cookies*, deixa marcats o desmarcats els checkbox segons estigui indicat a les cookies i crida al mètode actualitza. En cas que l'element que vulgueu recuperar sigui *null*, haureu de considerar que el checkbox no està marcat.
- **reset()**: esborra totes les dades de les cookies (les fa caducar). Deixa tots els checkbox sense marcar, mostra una alerta, i crida al mètode actualitza.

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 10 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

**Es demana:** completeu el fitxer **js/domotica.js**, que contindrà una implementació correcta de les funcions que es demanen al mateix fitxer.

**Caldrà que inclogueu** al comprimit del lliurament el fitxer **domotica.js** esmentat abans.

**En el següent enllaç trobareu un vídeo que us pot ajudar a comprendre el funcionament de l'exercici.**

<https://www.loom.com/share/cf9c545af92244de9e12ef7a45840b3d>

## Ajuts:

- Per actualitzar una imatge d'un element `img` anomenat "exemple", hauríem de fer:  
`document.getElementById("exemple").src="img/foto.jpg";`
- Necessitareu que les galetes no caduquin. Com l'especificació de les galetes no proporciona aquesta possibilitat, caldrà gravar les galetes perquè expirin, per exemple, d'aquí a 10 anys.

## Annex:

Molts exercicis necessiten llegir i modificar dades d'un formulari. El següent codi us pot ajudar en aquesta tasca.

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <title>Formulari</title>
</head>
<body onload="mostra_dades()">
  <h1>Tria un model de cotxe</h1>
  <form>
    Preu: <input type="number" id="preu" value="5" onkeyup="mostra_dades()"/>
    <br/><br/>
    Color:
    <select id="tipus_color" onchange="mostra_dades()">
      <option value="color_vermell" selected="selected">Vermell</option>
      <option value="color_groc">Groc</option>
      <option value="color_blau">Blau</option>
    </select>
    <p id="missatge">&nbsp;</p>
    <button type="button" onclick="canvia_dades();">Canvia dades</button>
  </form>
</script>
```

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 11 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020

```
function mostra_dades(){
    var preu=parseInt(document.getElementById("preu").value);
    var element = document.getElementById("tipus_color");
    var tipus_color = element.options[element.selectedIndex].value;
    var msg="Preu: "+preu+" Tipus color:"+tipus_color ;
    document.getElementById("missatge").innerHTML=msg;
}
```

```
function canvia_dades(){
    document.getElementById("preu").value=100;
    var element = document.getElementById("tipus_color");
    element.selectedIndex=2;
    document.getElementById("missatge").style.fontSize="40px";
    mostra_dades();
}
</script>
</body>
</html>
```

	Codi: I71	Exercici d'avaluació contínua 1	Pàgina 12 de 12
	Versió: 01	DAW_M06B1_EAC1_Enunciat_2021S1	Lliurament: 28/10/2020