

03

Desenvolupament Web en entorn Client

Cicle: Desenvolupament d'aplicacions web

Curs 2024-2025

©José Masip Alonso

1. Introducció

En aquest document aprendrem modificar propietats de css, crear un joc senzill, utilitzar cookies i programació orientada a objectes.

2. Contingut

2.1. Modificar propietats css.

Per a accedir a una propietat de css cal accedir a l'atribut style de l'element.

```
document.getElementById("id").style.propietatcss
```

Cal que aquesta estiga definida en el style de l'element

```
<element id="ele" name='nom'  
style="position:absolute;top:100px;left:100px;background-color:cyan">
```

Podem definir els estils sobre l'element, en el head o en un arxiu extern. Davant aquesta diversitat de possibilitats algunes voltes no ens mostra el valor de la propietat definida.

```
window.getComputedStyle(ELEMENT)
```

Aquest mètode obté totes les actuals (computades) propietats de CSS amb els seus valors de l'element HTML.

L'estil computat es calcula després d'aplicar els múltiples orígens per a definir estils (element, full extern, head, estil per defecte del navegador).

```
window.getComputedStyle(identificador) -----> undefined
```

```
window.getComputedStyle(document.getElementById("identificador"))
```

```
-----> vector amb propietats
```

```
window.getComputedStyle(document.getElementById("identificador")).style
```

```
-----> propietat no definida
```

```
window.getComputedStyle(document.getElementById("identificador")).atribut-css
```

```
-----> OK
```

Exemples:

```
window.getComputedStyle(document.getElementById("capa1")).top
```

```
window.getComputedStyle(document.getElementById("capa1")).getPropertyValue("top")
```

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <title>Javascript</title>
  <meta name="description" content="descripcio">
  <meta name="author" content="Jose Masip">
<script language="JavaScript">
  function mostrar()
  {
    document.getElementById("capa1").style.visibility="visible";
  }
  function ocultar()
  {
    document.getElementById("capa1").style.visibility="hidden";
  }
</script>
</head>
```

```
<body>
  <div id="capa1"
style="position:absolute;width:100px;height:100px;top:500px;left:100px;background-
color:cyan">Capa 1</div>
  <input type="button" value="mostrar" onClick="mostrar()">
  <input type="button" value="ocultar" onClick="ocultar()"> <br>
</body>
</html>
```

2.2. canvas

L'etiqueta canvas permet dibuixar, animar i processar imatges, etc.

Genera un espai rectangular buit en la pàgina, en el qual mostrarem els resultats d'executar la nova funcionalitat

```
<canvas id="lienzo" width="500" height="300">
  El navegador no soporta aquest element canvas
</canvas>
```

```
function iniciar()
{
    canvas=document.getElementById('lienzo');
    //obindre el context sobre el que podem dibuixar ---
    """"""taula de pixels""""
    lienzo=canvas.getContext('2d');

    //dibuixar un rectangle -----fillRect(x, y, ancho,
alto)

    lienzo.fillRect(110,110,10,10);

    // color
    lienzo.fillStyle="#000099";
    lienzo.fillRect(150,150,60,40);

    //afegir text
    lienzo.font="bold 24px verdana, sans-serif";
    lienzo.textAlign="center";
    lienzo.fillText("Missatge", 100,60);

    //afegir una imatge
    //drawImage(imagen, x, y)
    //drawImage(imagen, x, y, ancho, alto)
    var imagen=new Image();
```

```
    imagen.src="./ima/banana.gif";  
    lienzo.drawImage(imagen,40,0,40,40);  
}
```

|

Missatge



Si volem donar moviment a un element.

```
var lienzo2=null, canvas2=null;  
var x=50,y=50;  
function iniciar2()  
{  
    canvas2=document.getElementById('lienzo2');  
    lienzo2=canvas2.getContext('2d');  
  
    run();  
}  
  
function run()  
{  
    setTimeout(run, 17);  
    act();  
    paint(lienzo2);  
}  
  
function act()  
{  
    if(x>canvas2.width)  
        x=0;  
    else  
        x+=2;
```

```
}  
  
function paint(lienzo2)  
{  
    lienzo2.fillStyle='#0f0';  
    lienzo2.fillRect(x,y,10,10);  
}
```



Genera una linea ja que mou el quadrat de lloc sense eliminar l'anterior. Cal eliminar o esborrar tot cada volta.

Modificar el codi així:

```
function paint(lienzo2)  
{  
    lienzo2.fillStyle="#112299";  
    lienzo2.fillRect(0,0,canvas2.width,canvas2.height); // pinte fondo blau  
    lienzo2.fillStyle='#0f0';  
    lienzo2.fillRect(x,y,10,10);  
}
```

2.3. cookies

Les cookies permeten emmagatzemar informació de l'usuari a les pàgines web. És guarden a l'equip del client.

A les cookies podem guardar informació de l'usuari per a quan torne a visitar la pàgina.

La cookie es guarda en format nom_variable=valor.

```
function crear(){
    document.cookie="galleta=ieeee";
}

function llegir(){
    alert (document.cookie);
}
```

Podem fer que la cookie caduque afegint l'atribut expires.

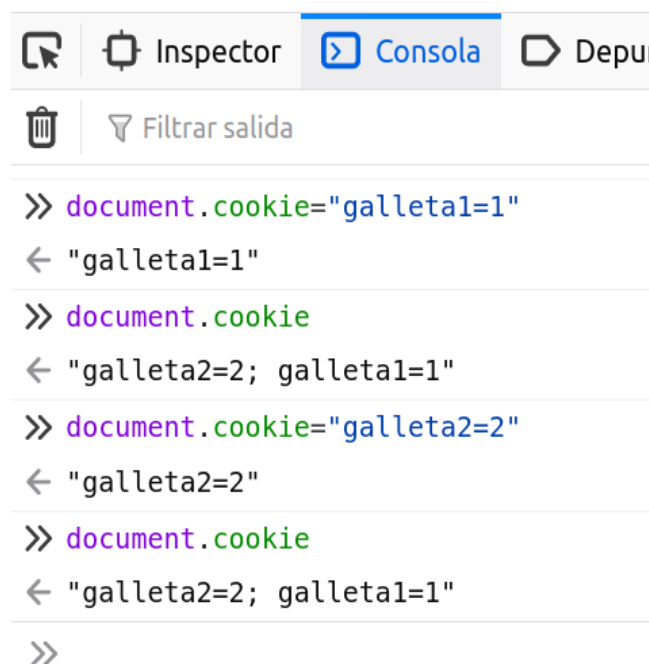
```
document.cookie="galleta=ieeee;expires=data a la que caduca"

var datacad= new Date();
datacad= new Date(datacad.getTime()+1000*60*5); //5minuts en ms
document.cookie="galleta=ee;expires="+datacad.toGMTString()+";";
```

Com podem guardar 2 cookies distintes? Com accedim a elles?

Com eliminem una cookie?

Per a guardar 2 cookies simplement les guardarem en distint nom.



Si ens fixem en exemple anterior podem deduir que les cookies es mostren com a un string. Per tant, per a accedir a aquestes cal obtindre els valors separant aquest string en parts. Si dividim la cadena pels ; conseguirem les distintes cookies. Posteriorment dividirem aquestes per = per a separar el nom de la cookie del seu valor.

Per a eliminar una cookie simplement la tornem a crear amb una data actual a aquest moment. Així la creem caducada i la eliminem.

2.4. HTML5 Localstorage

Amb localStorage, les aplicacions web poden emmagatzemar dades localment en el navegador de l'usuari.

Abans d'HTML5, dades d'aplicacions havien de ser emmagatzemada en les cookies, inclosos en cada petició de servidor. localStorage és més segura, i grans quantitats de dades poden ser emmagatzemats localment, sense afectar el rendiment web.

A diferència de les cookies, el límit d'emmagatzematge és molt gran (almenys 5 MB, cookies 4Kb) i la informació mai es transfereix al servidor (en cookies necessita transferir tota la informació).

L'emmagatzematge local és per domini. Totes les pàgines, d'un domini, poden emmagatzemar i accedir a les mateixes dades.

window.localStorage - magatzems de dades sense data de caducitat

code.sessionStorage - emmagatzema les dades per a un període de sessions (les dades es perden quan la llengüeta està tancat)


```
// guardar valor
localStorage.setItem("nom", "Rodrigo de la Fuente Luis");

// Obtindre valor
localStorage.getItem("nom");

//eliminar valor
localStorage.removeItem("nom");
```

2.5. Programació Orientada a Objectes

Les classes JavaScript són plantilles per a objectes JavaScript.

Utilitzem la paraula clau `class` per crear una classe. Amb `constructor()` definim l'objecte (aquest mètode es crida automàticament).

```
class NomClasse {
  constructor() {
    this.tribut=...;
    this.tribut2=...;
  }
  metode1(){...}
  metode2(){...}
}

let variable= new NomClasse();
variable.tribut2; //accedir a tribut
variable.metode1(); //cridar a un metode
```

Exemple de classe: client d'un banc.

Un client d'un banc té un compte bancari amb un saldo. Les operacions que pot fer sobre el compte es crear-lo (constructor), depositar i extraure diners.

```
class client {  
    constructor(nom,saldo) {  
        this.nom=nom;  
        this.saldo=saldo;  
    }  
  
    depositar(diners){  
        this.saldo=this.saldo+diners;  
    }  
  
    extraure(diners){  
        this.saldo=this.saldo-diners;  
    }  
}
```

this ----- fa referencia al objecte. Per tant, this.at fa referencia a l'atribut at de l'objecte que definim.

```
<script language="javascript">  
    var client1;  
    client1=new client('didac',1200);  
    document.write('Nom del client:'+client1.nom+'<br>');  
    document.write('Saldo actual:'+client1.saldo+'<br>');  
    client1.depositar(120);  
    document.write('Nou saldo---->'+client1.saldo+'<br>');  
    client1.extraure(1000);  
    document.write('Nou saldo---->'+client1.saldo+'<br>');  
</script>
```

Exemple complet

```
<!doctype html>  
<html lang="es">  
<head>  
    <meta charset="utf-8">  
    <title>titol</title>  
    <meta name="description" content="HTML5">  
    <meta name="author" content="Jose Masip">  
    <link rel="stylesheet" href="css/estilos.css">
```

```
<script language="javascript">
class client {
  constructor(nom,saldo) {
    this.nom=nom;
    this.saldo=saldo;
  }

  depositar(diners){
    this.saldo=this.saldo+diners;
  }

  extraure(diners){
    this.saldo=this.saldo-diners;
  }
}

var client1;
client1=new client('didac',1200);
document.write('Nom del client:'+client1.nom+'<br>');
document.write('Saldo actual:'+client1.saldo+'<br>');
client1.depositar(120);
document.write('Nou saldo---->'+client1.saldo+'<br>');
client1.extraure(1000);
document.write('Nou saldo---->'+client1.saldo+'<br>');
</script>
</head>
<body>

</body>
</html>
```

2.6. Propietats d'un event

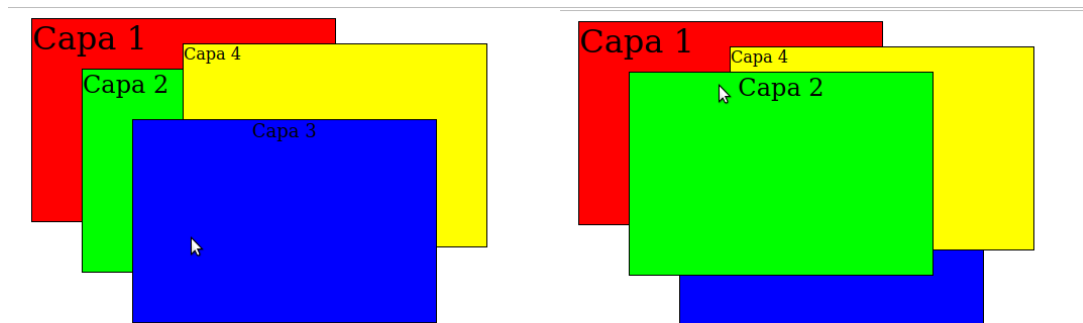
Podem detectar quina tecla hem polsat consultant propietats de l'event, coordenades en que hem provocat l'event, etc. Aquesta informació l'obtenim de l'objecte event.

```
event.keyCode  
event.charCode  
event.shiftKey  
event.clientX  
event.screenX
```

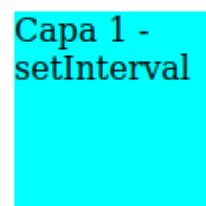
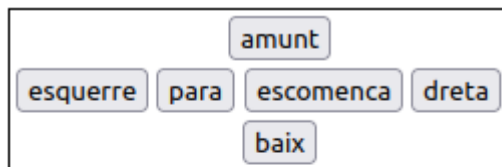
Recordar que cal definir un event sobre l'element (onkeypress, onkeydown,).

3. Activitats.

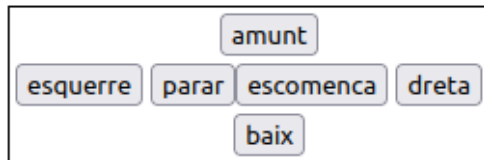
- E01. Crea una pàgina amb 4 capes que al posar-se damunt d'una d'elles la porte a la part superior (z-index)



- E02. Mou una capa amb setInterval.

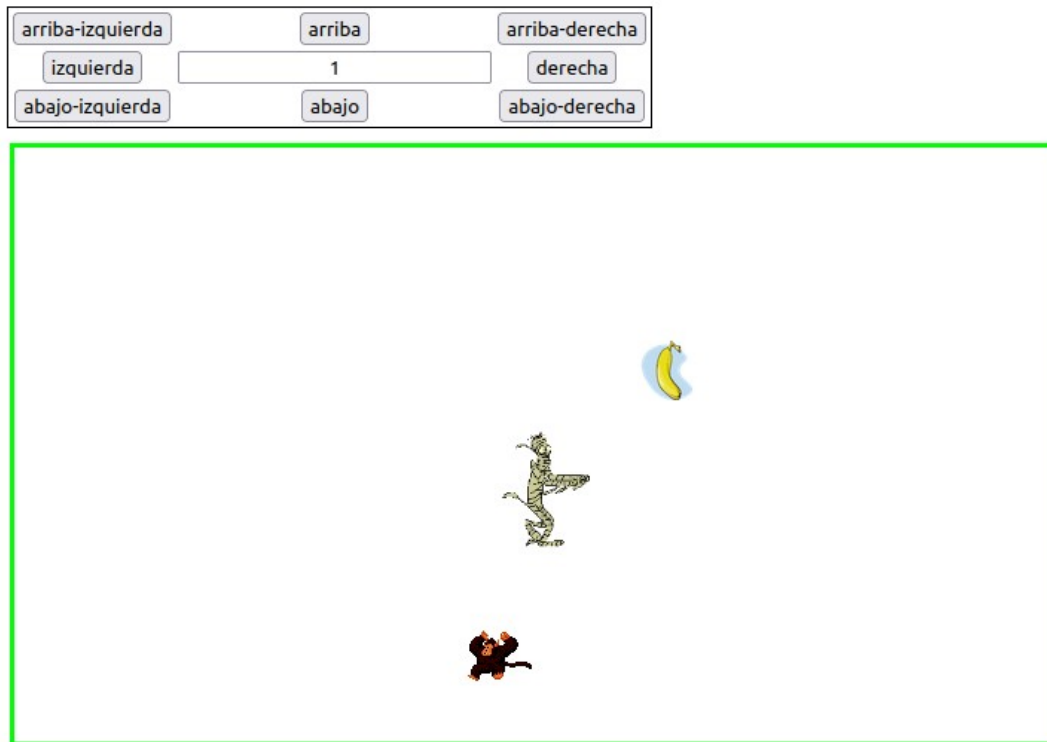


- E03. Mou una capa amb setTimeout.



Capa 1 -
setTimeout

- E04. Crea el joc següent (observa el video explicatiu):



- E05. Crea el joc anterior (amb canvas):
- E06. Crea una pàgina amb la següent funcionalitat:

Gestió d'un parking

client

data

codi

nom

matricula

marca

model

color

- És pretén registrar l'entrada del client amb el cotxe autoritzat.
- Per a evitar fer consultes a la base de dades cada volta que es canvia el valor del select, s'ha decidit descarregar totes les dades al carregar la pagina i emmagatzemar-les en el value del select separades per ";". **(RECORDAR: No podem fer php ni ajax encara)**
Ex: client;matricula;marca;model;color
- quan es seleccione un valor cal mostrar les dades en els distints camps.
- afegeix l'hora i data actual.
- Podem ocultar i mostrar els detalls del client
- existeix la possibilitat de buscar el client. Si existeix el que escrivim al camp, seleccionem eixa opció en el select. si no existeix mostrem l'error
- valida el formulari abans de registrar l'entrada al parking.

Codi del formulari

```
<form name="formulari">
  <table align="center">
    <tr>
      <td width="60">client</td>
      <td><select name="codi">
        <option value="-1">selecciona un codi de client...
        <option value="client1;matricula;marca;model;color">codiclient1
        <option value="client2;1234-wer;ford;mondeo;blau">codiclient2
        <option value="client3;4567-fgh;audi; a4;negre">codiclient3
        <option value="client4;0987-tyu;mercedes;merche;blanc">codiclient4
      </select>
      <a href=""></a>
      <input type="text" name="busca" size="25">

    </td>
  </tr>
  <tr>
    <td>data</td>
    <td><input type="text" name="data" size="55" disabled>
      
      
    </td>
  </tr>
</table>
</form>
```




```
</tr>
</table>
<div id="detall">
  <table align="center">
    <tr>
      <td width="60">codi</td>
      <td><input type="text" name="cod" size="66" disabled></td>
    </tr>
    <tr>
      <td>nom</td>
      <td><input type="text" name="nom" size="66" disabled></td>
    </tr>
    <tr>
      <td>matricula</td>
      <td><input type="text" name="matricula" size="66" disabled></td>
    </tr>
    <tr>
      <td>marca</td>
      <td><input type="text" name="marca" size="66" disabled></td>
    </tr>
    <tr>
      <td>model</td>
      <td><input type="text" name="model" size="66" disabled></td>
    </tr>
    <tr>
      <td>color</td>
      <td><input type="text" name="color" size="66" disabled></td>
    </tr>
  </table>
</div>

  <center><input type="button" name="boto" value="entrada parking"></center>
</form>
```

- E07. Crea un ahorcado:

ahorcado



paraula
s e e e

vides 4
missatges NO ESTA. Esperant nova lletra
lletra

sols apareix quan es guanya o es perd la partida

- Les paraules per al joc les traguem d'un vector de 8 posicions d'on es seleccionerà una aleatòriament per a cadascuna de les partides.
 - Les paraules son de longitud variable (max 15)
 - una lletra pot aparéixer mes d'una volta en la paraula.
 - paraules: moble, cua, novembre, porta, setembre, quadre, quatre, hipervincle.
 - al fallar 7 voltes el joc acaba i es dona la possibilitat de tornar a escomençar mostrant la paraula de la sol·lucio. es mostra en el camp vides, les opcions que et queden
 - Si existeix la lletra en alguna posicio, escriurela en la casella corresponent i es mostra un missatge en el camp felicitant al jugador. si no existeix la lletra, es resta una vida i s'indica l'error.
-
- E08. Crea una pàgina amb la següent funcionalitat:
 - crea un contador del número de voltes entres a la pagina en un minut.
 - Si no eixsteix la cookie, hi ha que crear-la.
 - si ja existeix, incrementar el número de visites.
 - Mostrar el número de visites actual dins d'un h1.

- E09. Crea una pàgina amb la següent funcionalitat:
 - crea una cistella de la compra amb cookies.
 - utilitza una (o mes) cookie per a emmagatzemar els productes i les unitats seleccionades per a la compra.
 - Incorpora un boto per a eliminar o buidar el carrito.
- E10. Adapta el joc del dossier 03 (mono) per a que funcione amb tecles.
- E11. Modifica l'exercici 07 amb aquesta funcionalitat:
 - No es pot repetir una lletra. Si he dit la «a» i aquesta ja havia sigut utilitzada restarà una vida.
 - Mostrar un camp d'ajuda on es mostren les lletres que ja hem dit.
 - Les lletres les detectem automaticament al polsar el teclat sobre qualsevol lloc de la pantalla.



4. Activitats entregables.

Entrega les següents activitats:

- E04 o E05
- E06
- E11

5. Bibliografia.

- <https://www.w3schools.com/js/>

6. Aclariments

No intentes copiar exercicis d'altres companys o internet. ni tans sols pegar una miradeta. Si que pots consultar la sintaxis d'una funció, etc. Però no et serveix per a res intentar buscar la solució.

És el major error que pot realitzar qui comença a programar. Sols et deixarà una falsa sensació d'aprenentatge i després no sabràs solucionar els problemes per tu mateix.

Cal intentar que el resultat que genere el vostre codi siga el mes paregut possible a l'exercici (si hi ha imatge d'exemple).

Com els Navegadors no son compatibles entre sí al 100% jo corregire els examens amb FIREFOX. Per tant, es recomanable utilitzar aquest per als nostres exercicis.

Crea uns arxius per a cada exercici o apartat. Així un error a un no afectarà a la resta.

Per a entregar els exercicis crear una carpeta amb el vostre nom i comprimiu aquesta (el nom de l'arxiu comprimit serà elteunom.zip o elteunom.tar.gz). Si entregueu arxius solts o arxius genèrics sense identificar el vostre nom puc confondre'm i no puntuar-vos algun exercici al no saber de qui és.

Cal entregar tots els arxius necessaris per a l'exercici (fotos, llibreries, etc).