

Tehnici Web

CURSUL 12

Semestrul I, 2018-2019
Carmen Chirita

<https://sites.google.com/site/fmitehniciweb/>

CANVAS

Elementul HTML `<canvas>` este folosit pentru a desena grafica, cu ajutorul JavaScript.

Elementul `<canvas>` este doar un container pentru grafică (este necesar un script pentru a desena grafica).

Canvas are mai multe metode pentru a desena linii, figuri, cercuri, text și inserare de imagini.

CANVAS

```
#canvas {border: 2px solid black  
body {background-color: red;}
```

```
<canvas id="canvas" width="500" height="400">  
Continut alternativ </canvas>
```

```
canvas = document.getElementById("canvas");  
context = canvas.getContext("2d");
```



<https://developer.mozilla.org/en-US/docs/Web/API/HTMLCanvasElement>

<http://www.w3.org/TR/2010/WD-html5-20100624/the-canvas-element.html>

HTMLCanvasElement **metoda** toDataURL

Returnează un URL de date ce conține o reprezentare a imaginii în formatul specificat de parametrul de tip (implicit png).

canvas.toDataURL(type, encoderOptions)

// type poate fi: image. png // "image/jpeg", "image/webp"
//encoderOptions: calitatea imaginii (intre 0-1)

```
<body>  
<div id="imgcanvas">  
  <h1>Imagine creata cu canvas </h1>  
</div>  
</body>
```

Canvas-ul poate fi folosit ca o imagine bitmap

```
var url = canvas.toDataURL();  
var newImg = document.createElement("img");  
newImg.src = url;  
var newImgParent = document.getElementById("imgcanvas");  
newImgParent.appendChild(newImg);
```

Canvas: fillStyle și fillRect()

fillStyle //stilul folosit în interiorul formelor
poate fi culoare /gradient/ patern

fillRect(x,y,width,height) //deseneaza un dreptunghi

```
var canvas = document.getElementById("canvas");  
var ctx = canvas.getContext("2d");
```

```
ctx.fillStyle = "yellow";  
ctx.fillRect(50, 20, 100, 100);
```

```
<style type="text/css" >  
#canvas {border: 2px solid black;}  
</style>
```

```
<body>  
<h1> Canvas </h1>  
<canvas id="canvas" width="300" height="200"> alt</canvas>  
</body>
```

Canvas

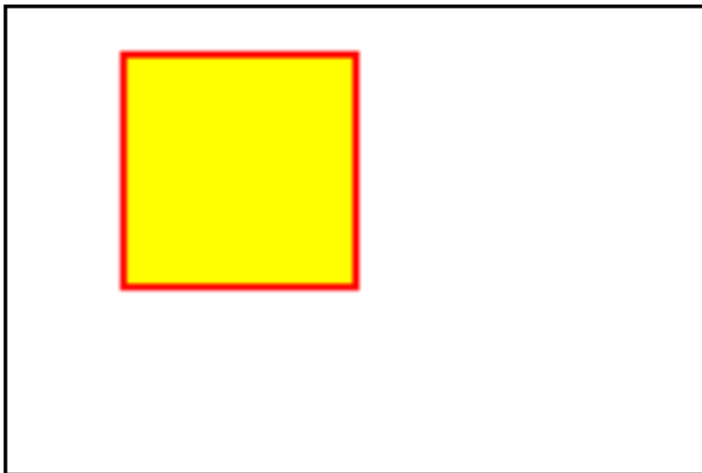


Stroke (conturul) si Fill (continutul)

```
ctx.strokeStyle = "red";  
ctx.lineWidth = 3;  
ctx.strokeRect(50, 20, 100, 100);  
  
ctx.fillStyle = "yellow";  
ctx.fillRect(50, 20, 100, 100);
```

strokeStyle
lineWidth
fillStyle

Canvas



strokeRect(x,y,width,height)
fillRect(x,y,width,height)
clearRect(x,y,width,height)

(x,y) coltul stanga sus

Desenarea figurilor

`beginPath()` //incepe desenarea unei noi figuri

`closePath()` //inchide figura de la punctul curent
la punctul de plecare

Desenarea unei linii

`moveTo(x,y)` //coordonatele de inceput

`lineTo(x,y)` // coordonatele de sfarsit

`lineCap` //stilul capetelor liniei ("butt|round|square")

`stroke()` //deseneaza efectiv linia

Desenarea unei cerc

`beginPath()`

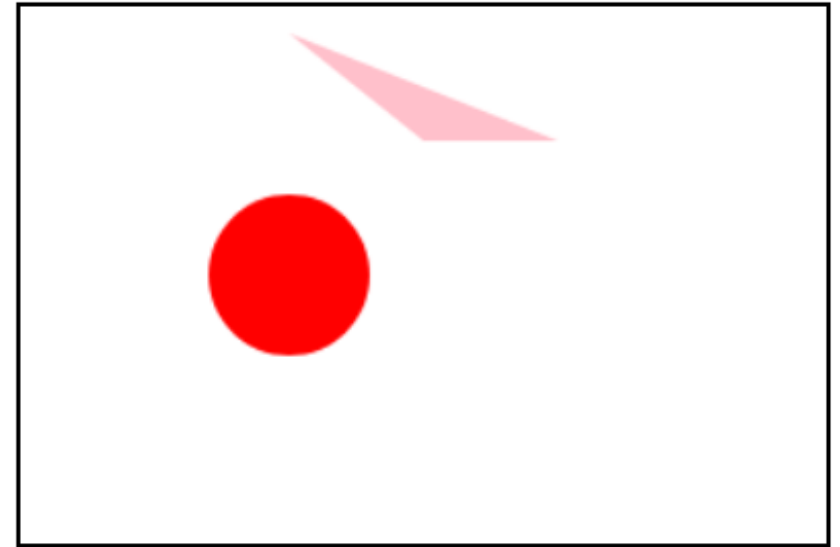
`arc(x,y,r,startangle,endangle)` //creeaza un cerc (arc de cerc)

`stroke()` /* `fill()` //deseneaza doar conturul sau
coloreaza si continutul

```
var canvas=document.getElementById("canvas");  
var ctx = canvas.getContext("2d");
```

```
ctx.beginPath();  
ctx.moveTo(100,10);  
ctx.lineTo(150,50);  
ctx.lineTo(200,50);  
ctx.lineTo(100,10);  
ctx.lineCap = "round";  
ctx.fillStyle = "pink";  
ctx.fill();  
ctx.closePath();
```

```
ctx.beginPath();  
ctx.arc(100, 100, 30, 0, 2* Math.PI);  
ctx.fillStyle = "red";  
ctx.fill();
```



```
<body>  
<h1> Canvas </h1>  
<canvas id="canvas" width="300" height="200"> alt</canvas>  
</body>
```

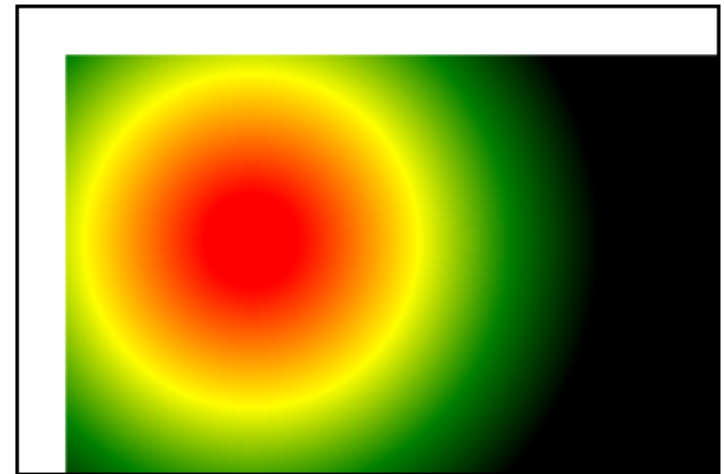

linearGradient, radialGradient

```
createLinearGradient(x0,y0,x1,y1)  
createRadialGradient(x0,y0,r0,x1,y1,r1)  
addColorStop()
```

```
var canvas = document.getElementById("canvas");  
var ctx = canvas.getContext("2d");
```

```
var fade = ctx.createLinearGradient(0, 0, 200, 200);  
fade.addColorStop(0, "red");  
fade.addColorStop(0.4, "yellow");  
fade.addColorStop(0.7, "green");  
fade.addColorStop(1.0, "black");
```

```
ctx.fillStyle = fade;  
ctx.fillRect(20, 20, 300, 300);
```



```
var fade = ctx.createRadialGradient(100,100,20,100,100,150);
```

Text și shadow

```
font  
strokeText(text,x,y)  
fillText(text,x,y)
```

```
shadowOffsetX  
shadowOffsetY  
shadowBlur  
shadowColor
```

```
ctx.shadowOffsetX = 10;  
ctx.shadowOffsetY = 10;  
ctx.shadowBlur = 4;  
ctx.shadowColor = "#666666";  
ctx.fillStyle = "green";  
//ctx.strokeStyle = "green";  
  
ctx.font = "italic 60px sans-serif";  
ctx.lineWidth = 1;  
ctx.fillText("Text Text",20, 105);  
//ctx.strokeText("Text Text",20, 105);
```



fill text



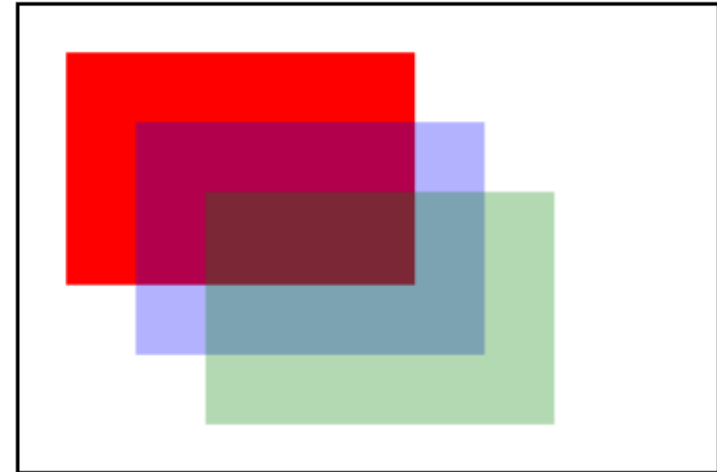
stroke text

CANVAS: globalAlpha si globalCompositeOperation

globalAlpha()
transparenta (intre 0-1=opac)

globalCompositeOperation
modul de amestecare a culorilor

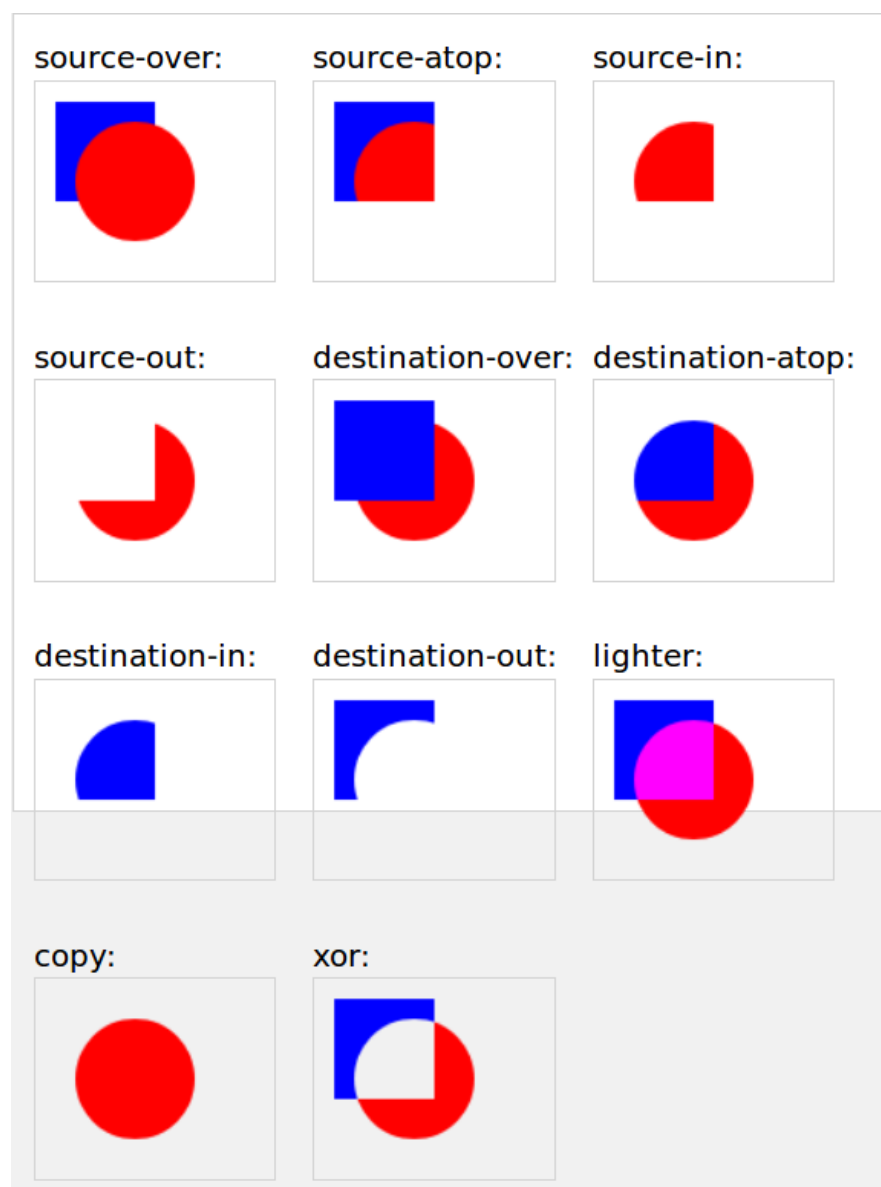
```
ctx.fillStyle = "red";  
ctx.fillRect(20, 20, 150, 100);  
  
ctx.globalAlpha = 0.3;  
ctx.fillStyle = "blue";  
ctx.fillRect(50, 50, 150, 100);  
ctx.fillStyle = "green";  
ctx.fillRect(80, 80, 150, 100);
```



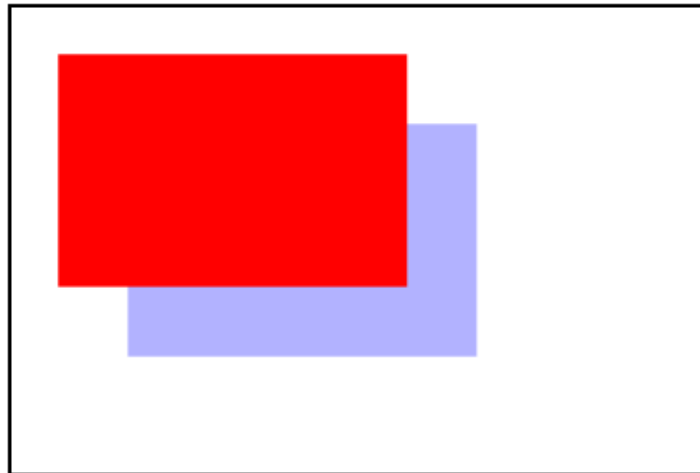
globalCompositeOperation
modul de amestecare a culorilor

specifica modul în
care o imagine
sursă (nouă)
este plasata pe o
imagine destinație
(existentă).

implicit:source-over



```
ctx.fillStyle = "red";  
ctx.fillRect(20, 20, 150, 100);  
  
ctx.globalAlpha = 0.3;  
ctx.globalCompositeOperation="destination-over";  
ctx.fillStyle = "blue";  
ctx.fillRect(50, 50, 150, 100);
```



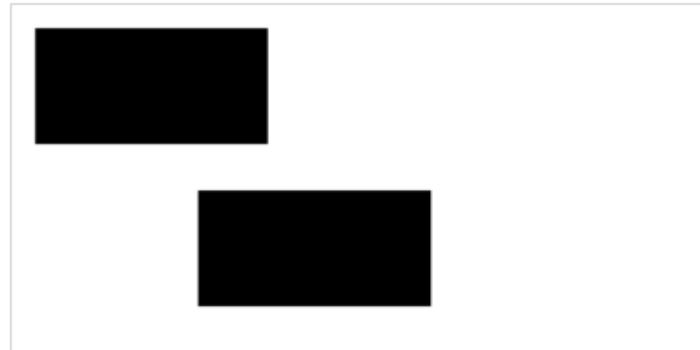
Transformari si salvarea contextului

```
ctx.translate(x,y)  
ctx.rotate(degrees*Math.PI/180)  
ctx.scale(factor)
```

se aplica tuturor
elementelor desenate

```
ctx.save()  
ctx.restore()
```

```
ctx.fillRect(10, 10, 100, 50);  
ctx.translate(70, 70);  
ctx.fillRect(10, 10, 100, 50);
```



Desenarea imaginilor

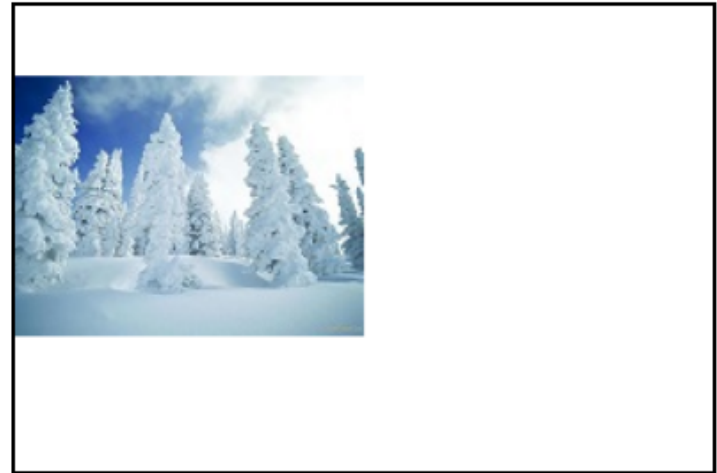
drawImage(image,x0,y0,width,height)

fisier.html

```
<canvas id="canvas" width="300" height="200"> alt</canvas>  
<p style = "display:none">  
  
</p>
```

fisier.js

```
var canvas = document.getElementById("canvas");  
var ctx = canvas.getContext("2d");  
var image = document.getElementById('image');  
ctx.drawImage(image, 0, 30);
```



Subiecte date la examen

S1. Sa se implementeze urmatoarea functionalitate:
la click pe orice paragraf se construiesc o lista cu continutul elementelor ****
care sunt copii ai paragrafului, lista fiind adaugata in document imediat dupa paragraf;
urmatoarele clickuri pe paragraf nu vor avea nici un efect.

Ex: daca p este

`<p id="pgf">Ion are o pisica, un catel, un papagal. </p>`
atunci lista va contine: **pisica, catel, papagal.**

Ion are o **pisica**, un **catel**, un **papagal**.

George are o **bicicleta** si un **scuter**.

Ana are **mere**, cateva **pere**, si multe **nuci**.

Vine **vacanta**, de Craciun voi primi **cadouri** si voi rezolva **exercitii** la **Tehnici Web** pentru **examen**.

A venit vacanta!

Ion are o **pisica**, un **catel**, un **papagal**.

- pisica
- catel
- papagal

George are o **bicicleta** si un **scuter**.

Ana are **mere**, cateva **pere**, si multe **nuci**.

- mere
- pere
- nuci

Vine **vacanta**, de Craciun voi primi **cadouri** si voi rezolva **exercitii** la **Tehnici Web** pentru **examen**.

- vacanta
- cadouri
- exercitii
- Tehnici Web
- examen

A venit vacanta!

liste.html

```
<script>
function copiiB(elem){ \creeaza lista cu continutul copiilor de tip b ai elementului elem
    var copii=elem.getElementsByTagName("b");
    var ul=document.createElement("ul");
    for(var i=0;i<copii.length;i++){
        var li=document.createElement("li");
        li.innerHTML=copii[i].innerHTML;
        ul.appendChild(li);
    }
    var frate=elem.nextElementSibling;
    document.body.insertBefore(ul,frate); \inserare lista imediat după elem
}

window.onload=function(){
    var colectie=document.getElementsByTagName("p"); \selectarea paragrafelor
    for(var i=0;i<colectie.length;i++){
        let k=i;
        colectie[i].nr=0; \proprietate noua care reține nr. clickului
        colectie[i].onclick = function(){ \clickul pe paragrafe cu apelarea functiei copiiB
            if(colectie[k].nr==0){
                copiiB(colectie[k]);
                colectie[k].nr=1;
            }
        }
    }
}

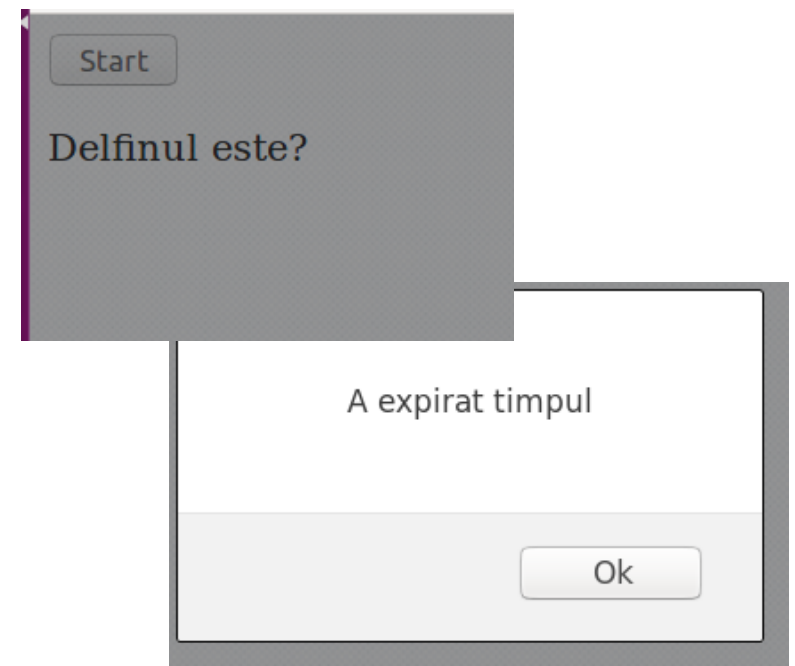
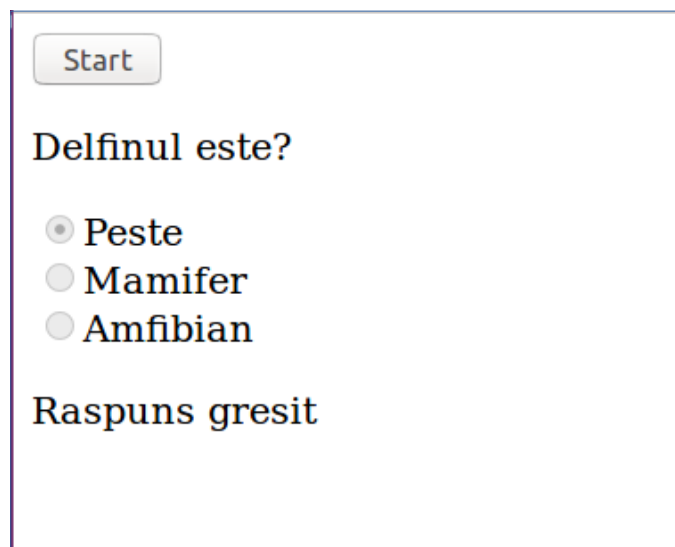
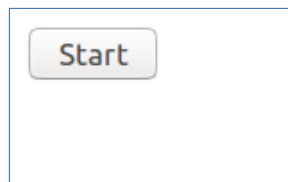
</script>
```

S2. Un fișier HTML conține: un `<button>` cu id-ul "start", un paragraf cu id-ul "răspuns", un paragraf cu id-ul "intrebare" care conține o întrebare, 3 butoane radio reprezentând variantele de răspuns, cu valorile "corect" sau "gresit".
Inițial întrebarea și butoanele radio nu sunt vizibile.

La click pe butonul "start" întrebarea și butoanele radio devin vizibile și utilizatorul are la dispoziție 5s pentru a selecta un răspuns.

Dacă utilizatorul a selectat un răspuns, în paragraful "răspuns" se afișează textul "Răspuns corect" sau "Răspuns gresit" în funcție de valoarea răspunsului.

După 5s, dacă nu s-a răspuns la întrebare, butoanele dispar și se afișează într-o fereastră alert "Timpul a expirat".



```

<script>
window.onload=function(){
    var intrebare=document.getElementById("intrebare");
    var forma=document.getElementById("form");
    var butoane=document.getElementsByTagName("input");

    document.getElementById("start").onclick=function(){ //click pe start
        intrebare.style.visibility="visible";
        forma.style.visibility="visible";

        var t=setTimeout(function(){
            document.body.removeChild(forma);
            alert("A expirat timpul");},5000);

        forma.onchange=function(){ //daca s-a selectat un buton
            for(var i=0;i<butoane.length;i++){
                butoane[i].disabled=true; //butoanele sunt dezactivate după
                selecția unui buton
                if(butoane[i].checked)
                    document.getElementById("raspuns").innerHTML=
                    "Raspuns"+ ' '+ butoane[i].value;
                \\generarea raspunsului
            }
            clearTimeout(t);
        }
    }
}

```

quizz.html

```

<body>
<button id="start">Start</button>
<p id="intrebare">Delfinul este?</p>
<form id="form">
<input type="radio" name="q1" value="gresit">Peste<br>
<input type="radio" name="q1" value="corect">Mamifer <br>
<input type="radio" name="q1" value="gresit">Amfibian<br>
</form>
<p id="raspuns"></p>
</body>

```

S1. Avem o pagina cu urmatorul body:

```
<ul id="imagini">  
  <li>img1.jpg</li>  
  .....  
  <li>imgk.jpg</li></ul>  
<button id="start">start</button>
```

La click pe butonul "start" numele fisierelor sunt inlocuite cu imaginile respective (src-ul fiind chiar numele fisierului) astfel: primul fisier este inlocuit cu imaginea care este afisata 3s, dupa 3s prima imagine este inlocuita cu numele fisierului, iar al doilea nume este inlocuit cu a doua imagine pe durata a 3s si asa mai departe.

- img1.jpg
- img2.jpg
- img3.jpg
- img4.jpg
- img5.jpg

start

- img1.jpg
- img2.jpg
- img3.jpg



-
- img5.jpg

start

imagini.html

```
<script>
```

```
window.onload=function(){
    document.getElementById("start").onclick=schimba;
    var img=document.getElementById("imagini").children;
    function imagine(i){ //creeaza o imagine și schimba fișierul i cu imaginea
        var sursa=img[i].innerHTML;
        var imagine=document.createElement("img");
        imagine.src=sursa;
        imagine.alt="eroare";
        imagine.fisier=sursa; //proprietate noua
        img[i].innerHTML="";
        img[i].appendChild(imagine);
    }
    function inapoi(i){ //schimba imaginea cu numele
                        fisierului
        var fisier=img[i].children[0].fisier;
        img[i].innerHTML=fisier;
    }

    function schimba(){
        imagine(0); //prima schimbare
        for(let i=1;i<=img.length;i++){
            setTimeout(function(){ //urmatoarele schimbări la fiecare 3s
                inapoi(i-1);
                if (i < img.length) imagine(i);},3000*i);
        }
    }
}</script>
```

```
<ul id="imagini">
    <li>img1.jpg</li>
    <li>img2.jpg</li>
    <li>img3.jpg</li>
    <li>img4.jpg</li>
    <li>img5.jpg</li>
</ul>
<button id="start">start</button>
```

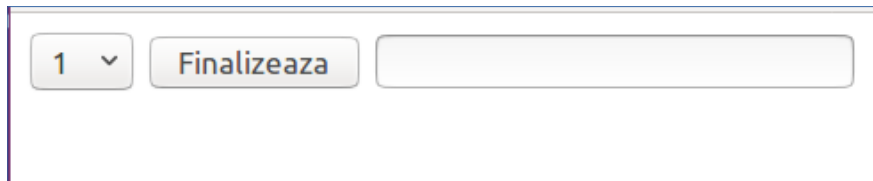
S2. Un fisier HTML contine un element `<select>` ale carui optiuni contin numerele de la 1 la 100, un `<button>` cu id-ul "finalizeaza", un element `<input>` cu id-ul "rezervari" si valoare initiala "" (sirul vid).

La selectarea optiunii n se genereaza n butoane care vor contine numerele de la 1 la n și vor avea initial culoarea verde.

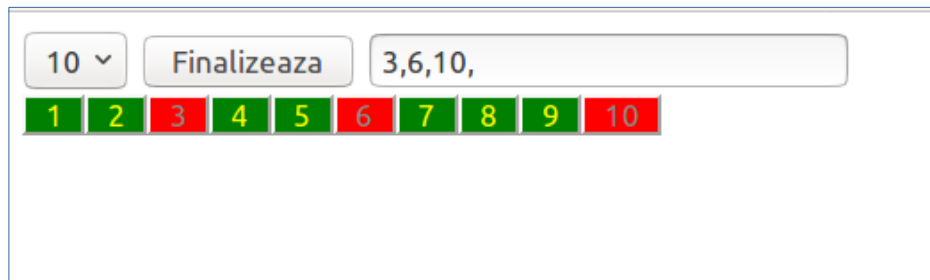
La click pe un buton verde, numarul biletului va fi retinut in input, butonul va deveni rosu, iar clickul nu va mai avea nici un efect.

La click pe butonul "finalizeaza", se va afisa alert cu biletele rezervate (continutul inputului), iar continutul inputului va fi setat la "".

Ex: daca locurile rezervate sunt 3, 50,7 atunci continutul input-ului va fi "3,50,7"



The initial user interface consists of a dropdown menu showing '1', a button labeled 'Finalizeaza', and an empty text input field.



After selecting '10' from the dropdown, the interface shows the dropdown with '10', the 'Finalizeaza' button, and an input field containing '3,6,10,'. Below the input field is a row of 10 buttons numbered 1 to 10. Buttons 1, 2, 4, 5, 8, and 9 are green, while buttons 3, 6, 7, and 10 are red.



An alert dialog box is displayed with the text '" 3,6,10, "' and an 'Ok' button at the bottom.

```
<script>
```

```
window.onload=function() {
```

```
    var sel=document.getElementById("op");
```

```
    var final=document.getElementById("finalizeaza");
```

```
    var inp=document.getElementById("rezervari");
```

```
    sel.onchange = function() { \la selectarea unei optiuni din select
```

```
        var n = sel.selectedIndex+1; \optiunea selectata
```

```
        for(var i=0; i<n;i++) { \creare butoane
```

```
            var bt=document.createElement("button");
```

```
            bt.innerHTML=i+1;
```

```
            document.body.appendChild(bt);
```

```
            bt.className="verde";
```

```
            bt.onclick=rezerva; \la click pe butoane se apeleaza functia rezerva
```

```
        }
```

```
    }
```

```
    function rezerva() {
```

```
        inp.value+=this.innerHTML + ",";
```

```
        this.className="rosu";
```

```
        this.disabled=true;
```

```
    }
```

```
    final.onclick = function(){
```

```
        alert(' ' + inp.value + ' ');
```

```
        inp.value="";
```

```
    }
```

```
}
```

```
</script>
```

```
.verde{background-color:green; color:yellow;}
```

```
.rosu{background-color:red;
```

```
<body>
```

```
<select id="op">
```

```
<option value="1">1</option>
```

```
<option value="2">2</option>
```

```
.....
```

```
</select>
```

```
<button id="finalizeaza">Finalizeaza</
```

```
button>
```

```
<input id="rezervari" value="">
```