



CodeCamp JavaScript

Algorithmics Brasov-Smart Kids



Ziua 5

Jocul Ping-Pong



Introducere



Obiectiv:

Construim un joc simplu de ping pong cu două platforme controlate de tastatură.

Introducere

Obiectivele zilei:

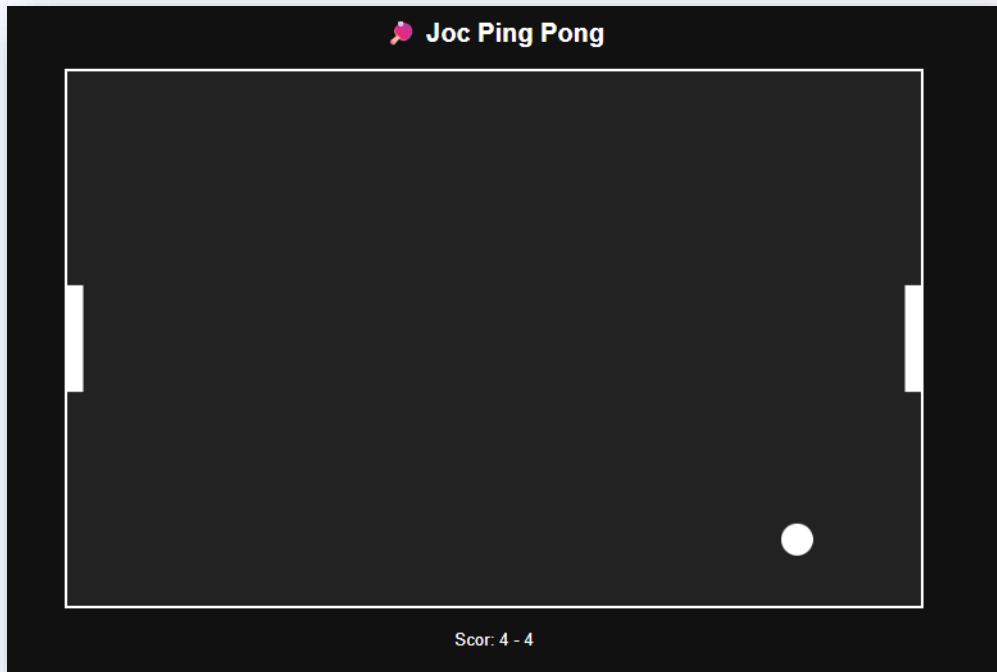
- Să folosim împreună HTML, CSS și JavaScript pentru a construi un joc complet
- Să controlăm obiecte cu tastele de pe tastatură (paletele jocului)
- Să aplicăm detectarea coliziunilor (mingea ricoșează din palete și pereți)
- Să afișăm și să actualizăm scorul în timp real

Structura proiectului

- ❑ **HTML** – zona de joc (canvas) + scorul
- ❑ **CSS** – stilizare pentru teren, palete și minge
- ❑ **JavaScript** – logica jocului:
 - mișcarea paletelor
 - mișcarea mingii
 - Coliziuni
 - scor

Ce vom realiza

Link pentru template-ul
proiectului CodePen:
<https://codepen.io/Violeta-the-sans/pen/myeGaKG>



HTML

👉 Avem o zonă de joc (canvas) și un scor afișat.

HTML

```
1 <h2>🍓 Joc Ping Pong</h2>
2   <canvas id="gameCanvas" width="800" height="500"></canvas>
3 <p>Scor: <span id="scoreLeft">0</span> - <span id="scoreRight">0</span>
4   </p>
5
```

CSS

👉 Facem ca jocul să arate ca un **teren de ping pong**.

* CSS

```
1 body {  
2   text-align: center;  
3   background: #111;  
4   color: #fff;  
5   font-family: Arial, sans-serif;  
6 }  
7  
8 canvas {  
9   background: #222;  
10  display: block;  
11  margin: 20px auto;  
12  border: 3px solid #fff;  
13 }
```


Setup JavaScript

👉 Avem variabile pentru palete, minge și scor.

📄 JS

```
1 // Setup canvas
2 const canvas = document.getElementById("gameCanvas");
3 const ctx = canvas.getContext("2d");
4
5 // Variabile pentru palete și minge
6 const paddleWidth = 15, paddleHeight = 100;
7 let leftY = canvas.height / 2 - paddleHeight / 2;
8 let rightY = canvas.height / 2 - paddleHeight / 2;
9
10 let ballX = canvas.width / 2, ballY = canvas.height / 2;
11 let ballSpeedX = 4, ballSpeedY = 4, ballSize = 15;
12
13 let scoreLeft = 0, scoreRight = 0;
```

Control palete

👉 Stânga: W/S

👉 Dreapta: ↑/↓

```
15 // Control palete
16 document.addEventListener("keydown", (e) => {
17     if (e.key === "w" && leftY > 0) leftY -= 20;
18     if (e.key === "s" && leftY < canvas.height - paddleHeight) leftY += 20;
19     if (e.key === "ArrowUp" && rightY > 0) rightY -= 20;
20     if (e.key === "ArrowDown" && rightY < canvas.height - paddleHeight) rightY += 20;
21 });
```

Mișcarea mingii & coliziuni

👉 Mingea se mișcă singură, ricoșează din pereți și palete, iar la gol se actualizează scorul.

```
function moveBall() {
  ballX += ballSpeedX;
  ballY += ballSpeedY;

  // Coliziune sus/jos
  if (ballY <= 0 || ballY + ballSize >= canvas.height) {
    ballSpeedY *= -1;
  }

  // Coliziune palete
  if (ballX <= paddleWidth && ballY + ballSize >= leftY && ballY <= leftY + paddleHeight) {
    ballSpeedX *= -1;
  }
  if (ballX + ballSize >= canvas.width - paddleWidth && ballY + ballSize >= rightY && ballY <= rightY + paddleHeight) {
    ballSpeedX *= -1;
  }

  // Goluri
  if (ballX < 0) { scoreRight++; resetBall(); }
  if (ballX > canvas.width) { scoreLeft++; resetBall(); }
}
```

Desenare

👉 Ștergem terenul și redesenăm paletele, mingea și scorul.

```
// Desenare elemente
61 function draw() {
62   ctx.clearRect(0, 0, canvas.width, canvas.height);
63
64   // Paleta stângă
65   ctx.fillStyle = "white";
66   ctx.fillRect(0, leftY, paddleWidth, paddleHeight);
67
68   // Paleta dreaptă
69   ctx.fillRect(canvas.width - paddleWidth, rightY, paddleWidth, paddleHeight);
70
71   // Minge
72   ctx.beginPath();
73   ctx.arc(ballX, ballY, ballSize, 0, Math.PI * 2);
74   ctx.fill();
75
76   // Actualizare scor
77   document.getElementById("scoreLeft").textContent = scoreLeft;
78   document.getElementById("scoreRight").textContent = scoreRight;
79 }
```

Resetare

👉 La gol, mingea revine în centru.

```
81 // Reset minge
82 function resetBall() {
83     ballX = canvas.width / 2;
84     ballY = canvas.height / 2;
85     ballSpeedX *= -1; // schimbă direcția
86 }
87
```

Buclo joclui

👉 Buclo rulează continuu, mută mingea, verifică coliziuni și redesenează terenul.

```
88 // Buclo principală
89 function gameLoop() {
90     moveBall();
91     draw();
92     requestAnimationFrame(gameLoop);
93 }
94
95 // Start joc
96 gameLoop();
97
```

Rezultatul final

- ✓ Două palete controlabile cu tastatura
- ✓ Mingea ricoșează din pereți și palete
- ✓ Scor afișat live
- 🏆 Joc simplu, dar complet



Recapitulare Finală

✨ Ce am învățat și construit în cele 5 zile





Recapitulare Finală

Ziua 1



Bazele JavaScript

- Variabile (let, const)
- Tipuri de date: numere, texte, true/false
- Condiții (if/else)
- DOM: cum „vorbit” cu pagina



Am făcut mini-exerciții simple



Recapitulare Finală

Ziua 2



Evenimente și animații

- Evenimente: click, tastatură, mouse
- Animații cu `setInterval` și `setTimeout`
- Obiecte care se mișcă pe ecran



Am construit mingea care ricoșează



Recapitulare Finală

Ziua 3



Funcții și Mini Pac-Man

- Funcții → cod refolosit
- Hărți 2D (matrice) pentru labirint
- Mișcarea jucătorului cu tastele
- Coliziuni și scor



Am făcut un **mini Pac-Man** cu fantomă



Recapitulare Finală

Ziua 4



Paint Interactiv

- Am folosit canvas pentru desen
- Alegere culoare și dimensiune pensulă
- Desen liber cu mouse-ul
- Buton pentru ștergere



Am creat propriul **Paint**



Recapitulare Finală

Ziua 5



Ping Pong

- Canvas = teren de joc
- Două palete controlate de tastatură
- Mingea ricoșează din pereți și palete
- Scor afișat în timp real



Am construit un **joc complet**

Felicitări! 🎉



👏 Bravo tuturor pentru efort!

V-ați descurcat excelent și acum sunteți **mini-programatori JavaScript** 🚀