<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width,initial-scale=1" />

<title>Endless Runner — Demo</title>

<style>

:root{--bg:#87ceeb;--ground:#2b6b2b;--ui:#ffffff}

html,body{height:100%;margin:0;font-family:Inter,Arial,Helvetica,sans-serif;background:var(--bg)}

#gameWrap{position:relative;width:100%;height:100vh;overflow:hidden}

canvas{display:block;background:linear-gradient(#87ceeb,#67b7e6);width:100%;height:100%}

#hud{position:absolute;left:12px;top:12px;color:#012;text-shadow:0 1px 0 #fff}

#hud .line{margin-bottom:6px;font-weight:700}

#controls{position:absolute;right:12px;top:12px;text-align:right}

button{padding:8px 12px;border-radius:8px;border:0;background:#fff;color:#222;font-weight:600;cursor:pointer;box-shadow:0 2px 6px rgba(0,0,0,0.15)}

#msg{position:absolute;left:50%;top:50%;transform:translate(-50%,-50%);background:rgba(0,0,0,0.6);color:#fff;padding:18px;border-radius:10px;display:none}

#leader{position:absolute;left:12px;bottom:12px;background:rgba(255,255,255,0.9);padding:8px;border-radius:8px;color:#000}

.small{font-size:13px;color:#333}

@media (max-width:600px){button{padding:6px 8px;font-size:14px}}

</style>

</head>

<body>

<div id="gameWrap">

<canvas id="c"></canvas>

<div id="hud">

<div class="line">Score: <span id="score">0</span></div>

<div class="line small">Coins: <span id="coins">0</span></div>

</div>

<div id="controls">

<button id="adBtn">Watch Ad → +10 coins</button>

<div style="height:6px"></div>

<button id="restartBtn">Restart</button>

</div>

<div id="leader">

<strong>Top:</strong>

<div id="toplist" class="small"></div>

</div>

<div id="msg"></div>

</div>

<script>

/\* ---------- Simple endless runner (single-file) ----------

- Canvas drawing (no assets)

- Coins + score

- Local leaderboard (localStorage)

- "Watch Ad" button simulates rewarded ad; replace with platform SDK call

-----------------------------------------------------------\*/

const canvas = document.getElementById('c');

const ctx = canvas.getContext('2d');

let W = canvas.width = Math.max(600, window.innerWidth);

let H = canvas.height = Math.max(360, window.innerHeight);

window.addEventListener('resize', ()=>{ W=canvas.width=window.innerWidth; H=canvas.height=window.innerHeight; });

/\* Game state \*/

let player = { x:80, y:0, w:48, h:48, vy:0, jump:-16, grounded:false };

let gravity = 0.9;

let obstacles = [], coins = [];

let scrollSpeed = 5;

let spawnTimer = 0, coinTimer = 0;

let score = 0, coinsCount = 0;

let running = true;

/\* UI refs \*/

const scoreEl = document.getElementById('score');

const coinsEl = document.getElementById('coins');

const msg = document.getElementById('msg');

const restartBtn = document.getElementById('restartBtn');

const adBtn = document.getElementById('adBtn');

const toplistEl = document.getElementById('toplist');

/\* Leaderboard via localStorage \*/

const LB\_KEY = 'erunner\_lb\_v1';

function loadLB(){ try{ return JSON.parse(localStorage.getItem(LB\_KEY)) || []; }catch(e){return [];} }

function saveScore(n){ if(!n) return; const lb = loadLB(); lb.push({score:n, time:Date.now()}); lb.sort((a,b)=>b.score-a.score); localStorage.setItem(LB\_KEY, JSON.stringify(lb.slice(0,10))); renderLB(); }

function renderLB(){ const lb = loadLB(); toplistEl.innerHTML = lb.length? lb.map((e,i)=>(`${i+1}. ${e.score}`)).join('<br>') : 'No scores yet'; }

renderLB();

/\* init \*/

player.y = H - 150;

player.vy = 0;

/\* helper collider \*/

function rectsOverlap(a,b){ return !(a.x + a.w < b.x || a.x > b.x + b.w || a.y + a.h < b.y || a.y > b.y + b.h); }

/\* spawn functions \*/

function spawnObstacle(){

const h = 40 + Math.random()\*40;

obstacles.push({ x: W + 20, y: H - 100, w: 30 + Math.random()\*30, h: 30 + Math.random()\*30 });

}

function spawnCoin(){

coins.push({ x: W + 20, y: H - 200 - Math.random()\*120, w:20, h:20 });

}

/\* input \*/

window.addEventListener('keydown', e=>{

if(e.code === 'Space' || e.code === 'ArrowUp') {

if(player.grounded){ player.vy = player.jump; player.grounded = false; }

}

});

canvas.addEventListener('touchstart', e=>{

if(player.grounded){ player.vy = player.jump; player.grounded = false; }

});

/\* main loop \*/

function update(){

if(!running) return;

// physics

player.vy += gravity;

player.y += player.vy;

if(player.y + player.h >= H - 90){ player.y = H - 90 - player.h; player.vy = 0; player.grounded = true; }

// spawn

spawnTimer += 1;

coinTimer += 1;

if(spawnTimer > 80 - Math.min(40, Math.floor(score/200))){ spawnObstacle(); spawnTimer = 0; }

if(coinTimer > 120){ spawnCoin(); coinTimer = 0; }

// move obstacles & coins

obstacles.forEach(o => o.x -= scrollSpeed + Math.min(3, Math.floor(score/500)));

coins.forEach(c => c.x -= scrollSpeed);

// remove offscreen

obstacles = obstacles.filter(o => o.x + o.w > -50);

coins = coins.filter(c => c.x + c.w > -50);

// collisions

for(let o of obstacles){

if(rectsOverlap(player, o)){

gameOver();

return;

}

}

for(let i = coins.length-1; i>=0; i--){

if(rectsOverlap(player, coins[i])){

coinsCount += 1;

score += 5;

coins.splice(i,1);

}

}

// increase score gradually

score += 0.5;

scoreEl.innerText = Math.floor(score);

coinsEl.innerText = coinsCount;

}

/\* draw \*/

function draw(){

// sky

ctx.fillStyle = '#87ceeb';

ctx.fillRect(0,0,W,H);

// ground

ctx.fillStyle = '#2b6b2b';

ctx.fillRect(0,H-90,W,90);

// player (rounded box)

ctx.fillStyle = '#e74c3c';

roundRect(ctx, player.x, player.y, player.w, player.h, 8, true, false);

// player eye

ctx.fillStyle = '#fff'; ctx.fillRect(player.x + player.w\*0.6, player.y + player.h\*0.2, 6,6);

// obstacles

ctx.fillStyle = '#2ecc71';

obstacles.forEach(o => roundRect(ctx, o.x, o.y - o.h, o.w, o.h, 4, true, false));

// coins

coins.forEach(c=>{

ctx.fillStyle = '#f1c40f';

ctx.beginPath(); ctx.arc(c.x + c.w/2, c.y + c.h/2, c.w/2, 0, Math.PI\*2); ctx.fill();

ctx.fillStyle = 'rgba(0,0,0,0.15)'; ctx.fillText('+', c.x + c.w/2 - 4, c.y + c.h/2 + 6);

});

// HUD shadow text

ctx.fillStyle = 'rgba(0,0,0,0.2)';

ctx.font = '18px Arial'; ctx.fillText('Score: ' + Math.floor(score), 16, 28);

if(!running){

ctx.fillStyle = 'rgba(0,0,0,0.6)';

ctx.fillRect(0,0,W,H);

ctx.fillStyle = '#fff'; ctx.font = '36px Arial';

const t = 'Game Over';

ctx.fillText(t, W/2 - ctx.measureText(t).width/2, H/2 - 10);

ctx.font = '18px Arial';

const s = 'Press Restart or Space to play again';

ctx.fillText(s, W/2 - ctx.measureText(s).width/2, H/2 + 22);

}

}

function roundRect(ctx,x,y,w,h,r,fill,stroke){

if (typeof r === 'undefined') r = 5;

ctx.beginPath();

ctx.moveTo(x+r,y);

ctx.arcTo(x+w,y,x+w,y+h,r);

ctx.arcTo(x+w,y+h,x,y+h,r);

ctx.arcTo(x,y+h,x,y,r);

ctx.arcTo(x,y,x+w,y,r);

ctx.closePath();

if(fill) ctx.fill();

if(stroke) ctx.stroke();

}

/\* game loop \*/

function loop(){

update();

draw();

requestAnimationFrame(loop);

}

/\* game over \*/

function gameOver(){

running = false;

saveHighscore(Math.floor(score));

showMsg('Game Over — Score: ' + Math.floor(score));

}

/\* restart \*/

function restartGame(){

obstacles = []; coins = [];

score = 0; coinsCount = 0;

player.y = H - 90 - player.h;

player.vy = 0; player.grounded = true; running = true;

hideMsg();

}

restartBtn.addEventListener('click', restartGame);

window.restartGame = restartGame;

/\* message \*/

function showMsg(t){

msg.style.display = 'block'; msg.innerText = t;

setTimeout(()=>{ msg.style.display='none'; }, 2200);

}

function hideMsg(){ msg.style.display='none'; }

/\* leaderboard storage \*/

function saveHighscore(n){

if(!n) return;

const lb = loadLB();

lb.push({score:n, ts:Date.now()});

lb.sort((a,b)=>b.score-a.score);

localStorage.setItem(LB\_KEY, JSON.stringify(lb.slice(0,10)));

renderLB();

}

/\* "watch ad" simulation — replace with platform SDK call on upload \*/

adBtn.addEventListener('click', ()=>{

// If you're on CrazyGames you can call CGAPI.showRewardedAd({...})

// Here we simulate an ad flow with a delay

adBtn.disabled = true; adBtn.innerText = 'Loading Ad...';

showMsg('Loading ad...');

setTimeout(()=>{

// simulate ad watched

coinsCount += 10;

showMsg('+10 coins awarded!');

adBtn.disabled = false; adBtn.innerText = 'Watch Ad → +10 coins';

coinsEl.innerText = coinsCount;

}, 1800);

});

/\* initial \*/

player.y = H - 90 - player.h;

player.grounded = true;

loop();

/\* render leaderboard initially on UI \*/

function renderLB(){ const lb = loadLB(); toplistEl.innerHTML = lb.length? lb.map((e,i)=>`${i+1}. ${e.score}`).join('<br>') : 'No scores yet'; }

renderLB();

</script>

</body>

</html>