Const canvas = document.getElementById(‘gameCanvas’);

Const ctx = canvas.getContext(‘2d’);

Canvas.width = 800;

Canvas.height = 600;

Let player = {

X: canvas.width / 2 – 20,

Y: canvas.height – 60,

Width: 40,

Height: 40,

Speed: 5,

Bullets: []

};

Let enemies = [];

Let score = 0;

Let gameOver = false;

Document.addEventListener(‘keydown’, movePlayer);

Document.addEventListener(‘keyup’, stopPlayer);

Function movePlayer(e) {

If (e.key === ‘ArrowLeft’) player.movingLeft = true;

If (e.key === ‘ArrowRight’) player.movingRight = true;

If (e.key === ‘ ‘) fireBullet();

}

Function stopPlayer(e) {

If (e.key === ‘ArrowLeft’) player.movingLeft = false;

If (e.key === ‘ArrowRight’) player.movingRight = false;

}

Function fireBullet() {

Player.bullets.push({ x: player.x + player.width / 2 – 5, y: player.y, width: 5, height: 10, speed: 7 });

}

Function spawnEnemy() {

Let size = Math.random() \* 30 + 20;

Enemies.push({

X: Math.random() \* (canvas.width – size),

Y: -size,

Width: size,

Height: size,

Speed: Math.random() \* 2 + 1

});

}

Function update() {

If (gameOver) return;

// Move player

If (player.movingLeft && player.x > 0) player.x -= player.speed;

If (player.movingRight && player.x + player.width < canvas.width) player.x += player.speed;

// Move bullets

Player.bullets.forEach(bullet => (bullet.y -= bullet.speed));

// Move enemies

Enemies.forEach(enemy => (enemy.y += enemy.speed));

// Remove bullets and enemies that are off-screen

Player.bullets = player.bullets.filter(bullet => bullet.y > 0);

Enemies = enemies.filter(enemy => enemy.y < canvas.height);

// Check for bullet-enemy collisions

Player.bullets.forEach((bullet, bulletIndex) => {

Enemies.forEach((enemy, enemyIndex) => {

If (bullet.x < enemy.x + enemy.width &&

Bullet.x + bullet.width > enemy.x &&

Bullet.y < enemy.y + enemy.height &&

Bullet.y + bullet.height > enemy.y) {

// Remove bullet and enemy on collision

Player.bullets.splice(bulletIndex, 1);

Enemies.splice(enemyIndex, 1);

Score++;

}

});

});

// Check for enemy-player collisions

Enemies.forEach(enemy => {

If (enemy.x < player.x + player.width &&

Enemy.x + enemy.width > player.x &&

Enemy.y < player.y + player.height &&

Enemy.y + enemy.height > player.y) {

gameOver = true;

}

});

// Spawn new enemies

If (Math.random() < 0.02) spawnEnemy();

}

Function draw() {

Ctx.clearRect(0, 0, canvas.width, canvas.height);

// Draw player

Ctx.fillStyle = ‘lightblue’;

Ctx.fillRect(player.x, player.y, player.width, player.height);

// Draw bullets

Ctx.fillStyle = ‘yellow’;

Player.bullets.forEach(bullet => ctx.fillRect(bullet.x, bullet.y, bullet.width, bullet.height));

// Draw enemies

Ctx.fillStyle = ‘red’;

Enemies.forEach(enemy => ctx.fillRect(enemy.x, enemy.y, enemy.width, enemy.height));

// Draw score

Ctx.fillStyle = ‘white’;

Ctx.font = ‘20px Arial’;

Ctx.fillText(`Score: ${score}`, 10, 30);

If (gameOver) {

Ctx.fillStyle = ‘white’;

Ctx.font = ‘40px Arial’;

Ctx.fillText(‘Game Over’, canvas.width / 2 – 100, canvas.height / 2);

}

}

Function gameLoop() {

Update();

Draw();

If (!gameOver) requestAnimationFrame(gameLoop);

}

gameLoop();