Gmail Mayank Raj <mayankrajnit.0405@gmail.com>

sanke programs

1 message

Mayank Raj <21mayankraj12d@gmail.com> Thu, May 25, 2023 at 12:50 AM
To: mayankrajnit.0405@gmail.com

```
<!DOCTYPE html>
<html>
<head>
 <title>Snake Game</title>
 <body>
 <div class="body">
 <div><h1>"Sequence Safari"</h1></div>
 <div id="scoreBox"><h5>Score: 0</h5></div>
 <div id="board"></div>
 </div>
</body>
 <style>
 #game-board {
 width: 575px;
 height: 575px;
 border: 1px solid #000;
 position: relative;
 background: linear-gradient(rgb(173, 186, 173),rgb(206, 206, 26));
 }

 .snake {
 background-color: purple;
 border: .25vmin solid white;
 border-radius: 12px;
 width: 20px;
 height: 20px;
 position: absolute;
 }

 .food {
 background: linear-gradient(red, purple);
 border: .25vmin solid black;
 border-radius: 8px;
 width: 20px;
 height: 20px;
 position: absolute;

}
#scoreBox{
 position: absolute;
```

```
  top: 9px;
  right: 200px;
  font-size: 39px;
  font-weight: bold;
  font-family: 'New Tegomin', serif;
 }

 </style>
</head>
<body>
 <div id="game-board"></div>

 <script>
 // Game variables
 var boardSize = 24;
 var snake = [{ x: 10, y: 10 }];
 var food = { x: 15, y: 10 };
 var direction = "right";
 var score=0;

 // Function to draw the game board
 function drawBoard() {
 var board = document.getElementById("game-board");
 board.innerHTML = "";

 // Draw snake
 snake.forEach(function(segment) {
 var snakeElement = document.createElement("div");
 snakeElement.className = "snake";
 snakeElement.style.left = segment.x * boardSize + "px";
 snakeElement.style.top = segment.y * boardSize + "px";
 board.appendChild(snakeElement);
 });

 // Draw food
 var foodElement = document.createElement("div");
 foodElement.className = "food";
 foodElement.style.left = food.x * boardSize + "px";
 foodElement.style.top = food.y * boardSize + "px";
 board.appendChild(foodElement);
 }

 // Function to handle keyboard input
 document.onkeydown = function(event) {
 switch (event.keyCode) {
 case 37: // Left arrow
 direction = "left";
 break;
 case 38: // Up arrow
 direction = "up";
```

```
break;
case 39: // Right arrow
direction = "right";
break;
case 40: // Down arrow
direction = "down";
break;
}
};

// Function to update the game state
function update() {
// Move snake
var head = Object.assign({}, snake[0]);
switch (direction) {
case "left":
head.x--;
break;
case "up":
head.y--;
break;
case "right":
head.x++;
break;
case "down":
head.y++;
break;
}
score= score+1;
snake.unshift(head);

// Check for collision with food
if (head.x === food.x && head.y === food.y) {
// Generate new food
food.x = Math.floor(Math.random() * boardSize);
food.y = Math.floor(Math.random() * boardSize);
score=score+1;
} else {
// Remove tail segment
snake.pop();
}

// Check for collision with walls or self
if (
head.x < 0 ||
head.y < 0 ||
head.x >= boardSize ||
head.y >= boardSize ||
checkCollision(head)
) {
```

```
clearInterval(gameLoop);
alert("Game Over!! Play Again.");
return;
}

// Redraw the game board
drawBoard();
}

// Function to check collision with self
function checkCollision(segment) {
for (var i = 1; i < snake.length; i++) {
if (snake[i].x === segment.x && snake[i].y === segment.y) {
return true;
}
}
return false;
}

// Start the game loop
var gameLoop = setInterval(update, 100);
</script>
</body>
</html>
```