

# Snake game.html

```
<!DOCTYPE html>
```

```
<html lang="en" dir="ltr">
```

```
<head>
```

```
  <meta charset="utf-8">
```

```
  <title>Snake Game JavaScript</title>
```

```
  <link rel="stylesheet" href="style.css">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <link rel="stylesheet"
```

```
    href="

```
    ss">
```


```

```
  <style>
```

```
    /* Import Google font */
```

```
    @import
```

```
    url("https://fonts.googleapis.com/css2?family=Open+Sans:wght@400;500;60
```

```
    0;700&display=swap");
```

```
    *{
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      box-sizing: border-box;
```

```
      font-family: 'Open Sans', sans-serif;
```

```
}
```

```
body{
```

```
  display: flex;
```

```
  align-items: center;
```

```
  justify-content: center;
```

```
  min-height: 100vh;
```

```
  background: #E3F2FD;
```

```
}
```

```
.wrapper{
```

```
  width: 65vmin;
```

```
  height: 70vmin;
```

```
  display: flex;
```

```
  overflow: hidden;
```

```
  flex-direction: column;
```

```
  justify-content: center;
```

```
  border-radius: 5px;
```

```
  background: #293447;
```

```
  box-shadow: 0 20px 40px rgba(52, 87, 220, 0.2);
```

```
}
```

```
.score{
```

```
    color: #B8C6DC;

    font-size: 1.5rem;

    text-align: center;

    padding: 10px 0;
}
```

```
.play-board{

    height: 100%;

    width: 100%;

    display: grid;

    background: #212837;

    grid-template: repeat(30, 1fr) / repeat(30, 1fr);

    position: relative;
}
```

```
.play-board.food{

    background: #FF003D;

    border-radius: 50%;

    box-shadow: 0 0 5px #FF003D;
}
```

```
.play-board.head{

    background: linear-gradient(to right, #4A90E2, #007ACC);
}
```

```
border-radius: 50%;  
  
position: relative;  
  
box-shadow: 0 4px 10px rgba(0, 0, 0, 0.5);  
  
}
```

```
.play-board.head.eye {  
  
  content: "";  
  
  position: absolute;  
  
  width: 20%;  
  
  height: 20%;  
  
  background: #FFFFFF;  
  
  border-radius: 50%;  
  
  box-shadow: 0 0 3px rgba(0, 0, 0, 0.3);  
  
}
```

```
.play-board.head.eye.left {  
  
  top: 20%;  
  
  left: 20%;  
  
}
```

```
.play-board.head.eye.right {  
  
  top: 20%;  
  
  right: 20%;  
  
}
```

```
}
```

```
.play-board.body{
```

```
background: linear-gradient(to right, #60CBFF, #3DB4FF);
```

```
border-radius: 50%;
```

```
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.3);
```

```
margin: 0;
```

```
}
```

```
.controls{
```

```
display: none;
```

```
justify-content: space-between;
```

```
}
```

```
.controls i{
```

```
padding: 25px 0;
```

```
text-align: center;
```

```
font-size: 1.3rem;
```

```
color: #B8C6DC;
```

```
width: calc(100% / 4);
```

```
cursor: pointer;
```

```
border-right: 1px solid #171B26;
```

```
}
```

```
@media screen and (max-width: 800px){

  .wrapper{

    width: 90vmin;

    height: 115vmin;

  }

  .controls{

    display: flex;

  }

  .controls i{

    padding: 15px 0;

    font-size: 1rem;

  }

}

</style>

</head>

<body>

  <div class="wrapper">

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

    <div class="play-board"></div>

    <div class="controls">

      <i data-key="ArrowLeft" class="fa-solid fa-arrow-left-long"></i>
```

```
<i data-key="ArrowUp" class="fa-solid fa-arrow-up-long"></i>
```

```
<i data-key="ArrowRight" class="fa-solid fa-arrow-right-long"></i>
```

```
<i data-key="ArrowDown" class="fa-solid fa-arrow-down-long"></i>
```

```
</div>
```

```
</div>
```

```
<script>
```

```
const playBoard = document.querySelector(".play-board");
```

```
const controls = document.querySelectorAll(".controls i");
```

```
const scoreDisplay = document.getElementById('score');
```

```
let gameOver = false;
```

```
let foodX, foodY;
```

```
let snakeX = 5, snakeY = 5;
```

```
let velocityX = 0, velocityY = 0;
```

```
let snakeBody = [];
```

```
let setIntervalId;
```

```
let score = 0;
```

```
const updateFoodPosition = () => {
```

```
    foodX = Math.floor(Math.random() * 30) + 1;
```

```
    foodY = Math.floor(Math.random() * 30) + 1;
```

```
};
```

```
const handleGameOver = () => {
```

```
  clearInterval(setIntervalId);
```

```
  alert("Game Over!");
```

```
  location.reload();
```

```
};
```

```
const changeDirection = e => {
```

```
  if (e.key === "ArrowUp" && velocityY !== 1) {
```

```
    velocityX = 0;
```

```
    velocityY = -1;
```

```
  } else if (e.key === "ArrowDown" && velocityY !== -1) {
```

```
    velocityX = 0;
```

```
    velocityY = 1;
```

```
  } else if (e.key === "ArrowLeft" && velocityX !== 1) {
```

```
    velocityX = -1;
```

```
    velocityY = 0;
```

```
  } else if (e.key === "ArrowRight" && velocityX !== -1) {
```

```
    velocityX = 1;
```

```
    velocityY = 0;
```

```
  }
```

```
};
```

```
controls.forEach(button => button.addEventListener("click", () =>
```



```
changeDirection({ key: button.dataset.key})));
```

```
const initGame = () => {
```

```
  if (gameOver) return handleGameOver();
```

```
  let html = `<div class="food" style="grid-area: ${foodY}/  
  ${foodX}"></div>`;
```

```
  if (snakeX === foodX && snakeY === foodY) {
```

```
    updateFoodPosition();
```

```
    snakeBody.push([foodY, foodX]);
```

```
    score += 5;
```

```
    scoreDisplay.innerText = 'Score: ' + score;
```

```
  }
```

```
  snakeX += velocityX;
```

```
  snakeY += velocityY;
```

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

```
    snakeBody[i] = snakeBody[i - 1];
```

```
  }
```

```
  snakeBody[0] = [snakeX, snakeY];
```

```
  if (snakeX <= 0 || snakeX > 30 || snakeY <= 0 || snakeY > 30) {
```

```
    return gameOver = true;
```

```

    }

    for (let i=0;i<snakeBody.length;i++){

        const className=i===0?'head':'body';

        html +=`<div class="${className}" style="grid-area: ${snakeBody[i][1]}
/${snakeBody[i][0]}">

            ${i===0?'<div class="eye left"></div><div class="eye
right"></div>':''}

            </div>`;

        if (i !==0&&snakeBody[0][1]=== snakeBody[i][1]&&snakeBody[0][0]===
snakeBody[i][0]){

            gameOver=true;

        }

    }

    playBoard.innerHTML=html;

};

updateFoodPosition();

setIntervalId= setInterval(initGame, 130);

document.addEventListener("keyup",changeDirection);

</script>

```

</body>

</html>

