Tic Tac Toe - Royal Edition HTML Code

```html
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

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

  <title>Tic Tac Toe - Royal Edition</title>

    <link  href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&display=swap"

rel="stylesheet">

  <style>

    body {

      margin: 0;

      font-family: 'Orbitron', sans-serif;

      background: linear-gradient(135deg, #0f0f0f, #1a1a1a);

      color: #fff;

      text-align: center;

      padding: 20px;

    }

    h1 {

      font-size: 3.2rem;

      color: gold;

      text-shadow: 0 0 10px gold;

      margin-bottom: 10px;

    }

    #scoreboard {
```

```css
  margin: 20px auto;

  font-size: 1.2rem;

  background: #111;

  border: 2px solid gold;

  border-radius: 10px;

  padding: 10px;

  width: fit-content;

  box-shadow: 0 0 20px gold;

}

.score-item {

  margin: 5px 15px;

  display: inline-block;

}

#game {

  display: grid;

  grid-template-columns: repeat(3, 110px);

  gap: 12px;

  justify-content: center;

  margin: 30px auto;

}

.cell {

  width: 110px;

  height: 110px;

  background: #222;

  border: 2px solid #666;

  border-radius: 15px;

  font-size: 2.8rem;
```

```css
  color: #fff;

  display: flex;

  justify-content: center;

  align-items: center;

  cursor: pointer;

  box-shadow: inset 0 0 10px #000;

  transition: all 0.3s ease;

}

.cell:hover {

  border-color: gold;

  box-shadow: 0 0 15px gold;

  transform: scale(1.05);

}

#status {

  font-size: 1.5rem;

  margin-top: 20px;

  color: cyan;

  text-shadow: 0 0 5px cyan;

}

.btns {

  margin-top: 25px;

}

button {

  margin: 5px;

  padding: 10px 20px;

  font-size: 1rem;

  border: none;
```

```css
      border-radius: 8px;

      background: gold;

      color: #000;

      font-weight: bold;

      cursor: pointer;

      box-shadow: 0 0 10px gold;

    }

    button:hover {

      background: #ffdf00;

      box-shadow: 0 0 15px #ff0;

    }

    .ai-toggle {

      background: #0cf;

      color: black;

      box-shadow: 0 0 10px #0cf;

    }

    .ai-toggle:hover {

      background: #0af;

    }

    .winner {

      color: #0f0;

      text-shadow: 0 0 10px lime;

    }

  </style>

</head>

<body>
```

```html
<h1>Tic Tac Toe - Royal Edition</h1>

<div id="scoreboard">

  <span class="score-item">X Wins: <span id="scoreX">0</span></span> |

  <span class="score-item">O Wins: <span id="scoreO">0</span></span> |

  <span class="score-item">Draws: <span id="scoreDraw">0</span></span>

</div>


<div id="game">

  <div class="cell" data-index="0"></div>

  <div class="cell" data-index="1"></div>

  <div class="cell" data-index="2"></div>

  <div class="cell" data-index="3"></div>

  <div class="cell" data-index="4"></div>

  <div class="cell" data-index="5"></div>

  <div class="cell" data-index="6"></div>

  <div class="cell" data-index="7"></div>

  <div class="cell" data-index="8"></div>

</div>


<div id="status">Player X's turn</div>


<div class="btns">

  <button onclick="restartGame()">Restart</button>

                  <button    class="ai-toggle"    onclick="toggleAI()">AI:    <span

id="aiStatus">OFF</span></button>

</div>
```

```html
<!-- Sounds -->

<audio id="clickSound"
src="https://cdn.pixabay.com/download/audio/2022/03/15/audio_ef4f208ea4.mp3"></audio>

<audio id="winSound"
src="https://cdn.pixabay.com/download/audio/2022/03/10/audio_a69846d2cb.mp3"></audio>

<audio id="drawSound"
src="https://cdn.pixabay.com/download/audio/2021/08/04/audio_52d8219631.mp3"></audio>


<script>
  const cells = document.querySelectorAll('.cell');

  const statusText = document.getElementById('status');

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

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

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

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

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

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

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


  let board = ['', '', '', '', '', '', '', '', ''];

  let currentPlayer = 'X';

  let gameOver = false;

  let isAI = false;


  const winCombos = [
    [0,1,2], [3,4,5], [6,7,8],
```

```javascript
  [0,3,6], [1,4,7], [2,5,8],

  [0,4,8], [2,4,6]

];



function checkWinner() {

  for (let combo of winCombos) {

    const [a,b,c] = combo;

    if (board[a] && board[a] === board[b] && board[b] === board[c]) {

      gameOver = true;

      statusText.textContent = `Player ${board[a]} wins!`;

      statusText.classList.add('winner');

      board[a] === 'X' ? scoreX.textContent++ : scoreO.textContent++;

      winSound.play();

      return;

    }

  }



  if (!board.includes('')) {

    statusText.textContent = "It's a draw!";

    scoreDraw.textContent++;

    drawSound.play();

    gameOver = true;

  }

}



function aiMove() {

  if (!isAI || currentPlayer !== 'O' || gameOver) return;
```

```
  const empty = board.map((v, i) => v === '' ? i : null).filter(v => v !== null);

  const move = empty[Math.floor(Math.random() * empty.length)];

  setTimeout(() => {

    makeMove(move);

  }, 500);

}


function makeMove(index) {

  if (board[index] || gameOver) return;

  board[index] = currentPlayer;

  cells[index].textContent = currentPlayer;

  clickSound.play();

  checkWinner();


  if (!gameOver) {

    currentPlayer = currentPlayer === 'X' ? 'O' : 'X';

    statusText.classList.remove('winner');

    statusText.textContent = `Player ${currentPlayer}'s turn`;

    aiMove();

  }

}


function handleClick(e) {

  const index = e.target.dataset.index;

  if (currentPlayer === 'X' || !isAI) makeMove(index);

}
```

```javascript
    function restartGame() {

      board = ['', '', '', '', '', '', '', '', ''];

      cells.forEach(cell => cell.textContent = '');

      currentPlayer = 'X';

      gameOver = false;

      statusText.classList.remove('winner');

      statusText.textContent = `Player X's turn`;

    }


    function toggleAI() {

      isAI = !isAI;

      aiStatus.textContent = isAI ? 'ON' : 'OFF';

      restartGame();

    }


    cells.forEach(cell => cell.addEventListener('click', handleClick));

  </script>


</body>

</html>
```